OPTIMIZING SWEET POTATOES YIELD: ADVANCED MODELLING WITH AUGMENTED SECOND-ORDER ROTATABLE DESIGNS USING TRIGONOMETRIC FUNCTIONS

By

Dennis Mwan Matundura

A Thesis Submitted to the Department of Mathematics, Physics and Computing, School of Science and Aerospace Studies in Partial Fulfillment of the Requirements for the Award of Degree of Doctor of Philosophy in Biostatistics

Moi University

2024

DECLARATION

Candidate

I declare that this thesis is my original work and has not been presented for any academic award in any institution; and shall not be reproduced in part or full, or in any format without prior written permission from the author and / or Moi University.

Signature:______________________________ Date:__________________________

Dennis M. Mwan

PHD/BS/03/17

Supervisors

This thesis has been submitted for examination with our approval as the University supervisors.

Signature:______________________________ Date:__________________________

Prof. Mathew K. Kosgei

Department of Mathematics, Physics and Computing,

Moi University.

Signature:______________________________ Date:__________________________

Dr. Robert Too

Department of Mathematics, Physics and Computing,

Moi University.

DEDICATION

I dedicate this thesis to my beloved family, lovely wife Zainab Mwan, my kids Shantel, Shawn and Shannel, whose unwavering support and encouragement have been my guiding light throughout this academic journey. To my Dad and Mum Mr. and Mrs. Matundura, your presence fills me with hope and motivation for the future. I am immensely grateful for the enduring friendships of Dr. Nyakundi, Mr. Kargung's, Mr. Momanyi, Mr.Minoo, and Engineer Pamela, whose camaraderie and encouragement have enriched my life in countless ways. Thank you for standing by me through every challenge and triumph.

ACKNOWLEDGEMENTS

The progress I have made thus far is solely attributed to our Lord God Almighty. I give all glory to Him for His abundant grace, strength, and unwavering guidance that have enabled me to successfully complete this PhD program. I am particularly grateful to my esteemed supervisors, Prof. M. Kosgei and Dr. R. Too, for their invaluable contributions to this thesis without their expert guidance, this achievement would not have been possible. I am also indebted to Prof. F. Nyamwala, the current chair of the Department of Mathematics, Physics and Computing, as well as Dr. R. Ramkat the current dean of school of Science and Aerospace, Prof. J.K. Koske, Prof. J.M. Mutiso, Prof. A. Mwangi, former Dean Prof. A. Kiprop, whose encouragement motivated me to pursue the PhD program. My heartfelt appreciation extends to my undergraduate lecturers, including Dr. W Koech, Dr. J. Koech, Dr. Maktiani and Dr. N. Rono, Dr. K. Korir, Dr. Cherutoi, Dr. Lusweti, Dr. Mutai, Mrs. Githinji, Mr. Chirchir and Mr. Cherutich, for their unwavering support and encouragement. I express my deepest gratitude to Dr. T. Rotich for his _nancial support and unwavering belief in my potential, which enabled me to complete my master's degree and pursue my PhD. I am also deeply thankful to Dr. I. Tum for his constant encouragement and motivation throughout my PhD journey. Additionally, I am grateful to Dr. A. Otieno for giving me the opportunity to teach undergraduate classes, which greatly enhanced my academic and professional development. I am forever grateful to my dear parents for their inspiration and the solid foundation they instilled in me. Special thanks to my colleagues Dr. Nyakundi, Dr. Dakwar and Mr. J. Magangi for their continuous encouragement. Lastly, I extend my heartfelt gratitude to my cherished friends madam P. Osongo family, Momanyi family, Kamau family my sister Faith, and brother Dickens, keep on, Sammy for their enduring love, support, and patience. Your constant motivation kept me going, and I am truly blessed to have you as my family and friends. May God richly bless each and every one of you.

ABSTRACT

Food security faces significant threats from climate change, population growth, and economic pressures. Despite efforts to improve productivity, distribution, and sustainable agriculture, many regions still remain insecure. Optimizing sweet potato yield requires carefully designed experiments to identify key yield-enhancing factors. The main objective of this study was to optimize sweet potato yield through advanced modelling with augmented second order rotatable design using trigonometric functions. The specific objectives were to: construct augmented second order rotatable designs in three dimensions; determine optimality criteria D-, A-, T- and E- for augmented second order rotatable designs; examine the relative efficiencies for augmented second order rotatable designs and apply augmented second-order rotatable designs to obtain design points that optimize sweet potato yield. The study transformed cyclic coordinates into Cartesian form to create numerical sequences used in moment conditions for developing diverse sets, which were further augmented to yield second-order rotatable designs in three-dimensional space. The evaluation of each design's performance and its relative efficiency involved the application of optimality criteria, such as D-, A-, T-, and E-criteria. The augmented second-order rotatable design was fitted to the data to obtain design points that optimize sweet potato yield. The R statistical software was used for modeling and analysis while Python was used in the construction of the design and evaluation of optimality criteria. The three augmented second order rotatable designs in three dimensions were successfully constructed. Generally, the twenty three point second order rotatable design turned out to be the most optimal and efficient design. Goat manure significantly outperformed poultry and rabbit manure, quadratic effects and interactions among poultry, rabbit, and goat manure also played a crucial role in boosting sweet potato yield. The study found that the optimal levels of poultry manure, rabbit manure, and goat manure that led to maximum yield were 35g/hole, 25g/hole and 45g/hole respectively. The application of these specific quantities resulted in the highest yield performance of 4kg/mounding. In conclusion, the study developed an optimal and efficient augmented second-order rotatable design. This advanced design was subsequently applied to effectively optimize sweet potato yield, demonstrating its practical utility in enhancing agricultural productivity. It is recommended that agricultural practitioners should consider integrating goat manure into sweet potato farming practices to optimize yield. The study underscores the importance of advanced experimental designs in agriculture for further research, promoting more efficient and sustainable farming techniques.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS AND ACRONYMS

A-	:	Average variance criterion.
ASORD	:	Augemented Second Order Rotatable Designs.
BIBD	:	Balanced Incomplete Block Designs.
CCD	:	Central Composite Designs.
D-	:	Determinant criterion.
DoE	:	Design of Experiments
E-	:	Eigenvalue criterion.
FAO	:	Food and Agriculture Organization
K	:	Potassium
KALRO	:	Kenya Agriculture and Livestock Research Organization.
N	:	Nitrogen
P	:	Phosphorus
PBD	:	Partial Block Designs.
PBIBD	:	Partial Balanced Incomplete Block Designs.
RSM	:	Response Surface Methodology
RSORD	:	Robust Second Order Rotatable Designs.
SORD	:	Second Order Rotatable Design
SOSRD	:	Second Order Slope Rotatable Designs.
SUBA	:	Simple Unrestricted Block Arrangment.
T-	:	Trace criterion.
WHO	:	World Health Organization.

CHAPTER ONE

INTRODUCTION

This chapter provides an overview of the study's context, outlines the problem being addressed, defines the objectives, and discusses the importance of the study.

1.1 Background Information

Sweet potatoes *(Ipomoea batatas)* are a vital crop worldwide, serving as a staple food in many regions due to their nutritional value, adaptability, and resilience to varying environmental conditions. However, the need to optimize sweet potato yield has become more pressing in the face of growing global food demand, climate variability, and resource limitations. While traditional farming methods have proven effective to an extent, they often fall short in capturing the complex interactions between multiple factors—such as soil fertility, irrigation, planting density, and climatic conditions—that influence crop growth.

To address these challenges, advanced modeling techniques that account for non-linear interactions and dynamic agricultural environments are required. One such advanced approach is the Augmented Second-Order Rotatable Design (ASORD), a powerful statistical tool rooted in response surface methodology. ASORD enhances the precision and flexibility of experimental designs, making it particularly effective for agricultural applications where numerous variables interact in complex, non-linear ways. By adding supplementary points to traditional designs, ASORD provides a more comprehensive exploration of the input factors, enabling more accurate predictions and optimization of crop yields.

This design has been further enhanced by the incorporation of trigonometric functions, which capture periodic and cyclic phenomena, such as seasonal variations, temperature fluctuations, and plant growth patterns. Trigonometric models help improve the predictive power of ASORD, making it a cutting-edge tool for optimizing sweet potato yield.

This study explores the use of Augmented Second-Order Rotatable Designs with Trigonometric Functions as an innovative approach to modelling and optimizing sweet potato yield. By integrating trigonometric functions, this advanced modelling technique provides a more accurate representation of the interactions between environmental variables and plant growth dynamics. The goal is to offer a practical yet sophisticated framework for farmers and researchers to achieve higher yields, more efficient resource use, and greater resilience to environmental variability. In a rapidly evolving agricultural landscape, these advanced techniques represent a crucial step toward sustainable and optimized sweet potato production.

The history and development of experimental designs, particularly second-order rotatable designs, provide a robust foundation for their modern applications in agriculture and food security. Box and Hunter, (1957) introduced the concept of rotatable designs. They developed various second-order rotatable designs for two and three components using geometric arrangements. These designs were rooted in the idea of a spherical variance function for least squares estimated responses, enabling the exploration of response surfaces. That same year, Hader et al.(1957) expanded on these ideas, presenting second-order rotatable designs as a class of experimental designs anchored in techniques from linear algebra and multivariate analysis. These designs relied on orthogonal arrays and symmetrical matrix properties to create balanced and efficient frameworks for experiments.

Draper (1959) emphasized the critical preparatory work required for rotatable designs, ensuring consistent experimental data collection across all directions equidistant from the design center. Bose and Draper, (1959) applied the transformation group approach to construct additional second-order rotatable designs, further strengthening the theoretical underpinnings of this methodology.

Box (1960) provided a comprehensive mathematical background for second-order rotatable designs, elaborating on their symmetrical properties and their role in modeling relationships between factors and response variables. Kiefer and Wolfowitz (1960) also highlighted the analytical simplicity and optimal resource utilization offered by these designs, making them increasingly appealing for agricultural experiments.

Das and Giri (1979) stressed the importance of proper experimental design to ensure the validity of assumptions for accurate data interpretation. Later, Box and Draper, (1987) expanded on their earlier work, focusing on the utility of rotatable designs with spherical variance functions for refining hypothesized models.

Santner (2003) explored how the symmetrical properties of second-order rotatable designs were instrumental in modeling factor-response relationships, reinforcing their importance for balanced and efficient designs. Berry et al. (1985) further demonstrated that orthogonal properties allowed factor rotation without variance, making these designs indispensable in agricultural applications. They also examined the interplay between orthogonal arrays and symmetrical matrices, which enabled robust experimental designs.

Khuri and Cornell (2018) introduced innovative second-order rotatable designs for experiments involving two factors, showcasing advancements in response surface methodology that enhanced their applicability and precision.

In contemporary times, these foundational principles have been applied to address critical challenges in agriculture, such as optimizing sweet potato *(Ipomoea batatas)* yields. Sweet potatoes are a vital crop globally, particularly for addressing food security challenges exacerbated by unpredictable climatic conditions, soil nutrient variability, and expensive farming inputs. Advanced methodologies like the Augmented Second-Order Rotatable Design (ASORD) have emerged as cutting-edge tools. By incorporating trigonometric functions, ASORD captures complex interactions and periodic phenomena such as seasonal variations and growth cycles. These innovations provide farmers and researchers with a sophisticated framework for achieving higher yields, efficient resource use, and resilience to environmental variability, aligning with Kenya's food security agenda.

This evolution of experimental design reflects a continuous effort to address global agricultural challenges, offering sustainable and efficient solutions for the future.

1.2 Basics Concepts

Definition 1.1: Rotatability: Khuri and Cornell (2018) defined rotatability in the context of experimental designs as a property where the variance of the estimated response at any point depends only on the distance of that point from the center of the design.

Remark 1.2: The given definition applies under the condition that the variance depends solely on the distance from the center of the design, as described by the formula. $\rho^2 = x_1^2 + x_2^2 + ... + x_n^2$, where ρ^2 is the distance from the center of the design, while x represents the input parameter that is being rotated.

Definition 1.3: Trigonometric functions: The Paper by Edmunds and Lang (2009), trigonometric functions are real functions that link the ratios of the lengths of the two sides of a right-angled triangle to an angle in the triangle.

Remark 1.4: The three main trigonometric functions are sine,cosine and tangent represented as sin(x),cos(x)and tan(x), where x denotes the measure of the angle in radians or degree.

Definition 1.5: RSM: Khuri and Cornell (2018) they defined Response Surface Methodology (RSM) is a collection of statistical and mathematical techniques employed to analyze scenarios involving multiple independent factors affecting a dependent variable, with the goal of optimizing the value of this dependent variable.

Remark 1.6: If a farmer wants to find the Potash (x_1), Nitrogen(x_2) and Phosphate (x_3) fertilizer levels that maximize the yield, the observed response y can be written as a function of the levels of potash, nitrogen and phosphate fertilizers as follows: $y = f(x_1, x_2, x_3) + \varepsilon$.

1.3 Statement of the Problem

Sweet potato *(Ipomoea batatas)* is a vital crop globally due to its nutritional value and adaptability to diverse growing conditions. However, optimizing its yield remains a significant challenge for farmers and researchers alike. Traditional methods of yield optimization often rely on basic experimental designs and empirical approaches, which may not fully capture the complex interactions between various agronomic factors. This results in suboptimal recommendations, leading to less efficient use of resources and lower crop yields. Food insecurity remains a critical issue affecting millions globally, particularly in low-income countries, with limited access to nutritious and affordable food leading to poor health outcomes and diminished quality of life. Effective agricultural interventions are needed to address these challenges. One promising approach involves using advanced modelling techniques, particularly augmented second-order rotatable designs (ASORDs) with trigonometric functions, to optimize agronomic variables such as growing conditions and fertilization. These designs allow for efficient testing and optimization of complex variable interactions, which can significantly enhance crop yields and nutrient content. Despite the demonstrated potential of second-order rotatable designs (SORDs) in various agricultural contexts such as optimizing nutritional supplements, identifying optimal crop growth conditions, and improving resource use there remains a significant gap in their application to sweet potato yield optimization using ASORDs with trigonometric functions. The core problem addressed in this study is the lack of comprehensive and robust models that can effectively optimize sweet potato yield by integrating complex variable interactions. Current models are limited in their ability to account for nonlinearities and interactions between factors such as soil properties, climate conditions, and cultivation practices. This study aims to develop and validate advanced modelling approaches using ASORDs combined with trigonometric functions. These models are expected to provide more accurate and reliable predictions of sweet potato yield under various conditions, ultimately guiding better agricultural practices and contributing to food security.

1.4 Objectives of the Study

1.4.1 General Objective

The main objective of this study was to optimize sweet potato yield through advanced modelling with augmented second order rotatable design using trigonometric functions.

1.4.2 Specific Objectives

The specific objectives were to;

a) Construct augmented second order rotatable designs in three dimensions using trigonometric functions.

b) Determine the D-, A-, T- and E- optimality criteria for the ASORD constructed using trigonometric functions.

c) Examine the relative D-, A-, T- and E- efficiencies for the ASORDs in three dimensions constructed using trigonometric functions.

d) Apply augmented second-order rotatable designs to identify design points that optimize sweet potato yield.

1.5 Significance of the Study

This study is significant as it explores the optimization of sweet potato yield through advanced modelling techniques, specifically using augmented second-order rotatable designs (ASORDs) with trigonometric functions. By incorporating organic fertilizers such as poultry manure, rabbit manure, and goat manure, this study aimed at identifying the optimal conditions for maximizing sweet potato yield. The use of these organic manures is particularly relevant in promoting sustainable agricultural practices, as they enhance soil fertility, improve soil structure, and provide essential nutrients to crops. The advanced modelling approach allows for a precise understanding of the interactions between these organic fertilizers leading to more effective and environmentally friendly cultivation practices. This study was envisaged to contributes to the existing literature on construction of rotatable designs by introducing new ASORDs constructed under trigonometric functions modelling.Using trigonometric functions concepts, the current study will be helpful in establishing a novel approach of generating augmented second-order rotatable designs in three dimensions. It is also expected to provide experimenters with a platform on how to evaluate several designs of the same category in order to establish their suitability and effectiveness in different crop production processes, as it extends the application in Sweet potatoes production.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter presents a trace in chronological order, the numerous schools of thought that have contributed to the investigation of response surface technique as well as the development and creation of second order rotatable designs.

2.2 Design of experiments

The Design and Analysis of Experiments (DoE) is a critical area of statistical science that focuses on understanding relationships between input variables (factors) and output variables (responses) through systematic experimentation. Recent advancements have significantly enhanced the field, reflecting the increasing complexity of scientific inquiries and the integration of modern computational methods.

Atkinson (2017), demonstrated the use of adaptive Bayesian designs in dose-finding studies, improving the efficiency and ethical considerations of clinical trials in the pharmaceutical industry.

Jin et al. (2018), showcased the effectiveness of Kriging models in capturing complex surfaces for RSM, particularly in applications such as climate modeling and aerodynamics, outperforming traditional RSM methods. Meyer and Nachtsheim (2024) highlighted the role of parallel computing in accelerating the construction of optimal designs, reducing computational time and improving efficiency. Smith et al. (2018) discussed the integration of DoE with precision agriculture technologies, such as remote sensing and GIS, for targeted experimental studies.

Montgomery et al.(2019), applied response surface methodology (RSM) to optimize parameters in additive manufacturing processes. Singh et al.(2019) utilized active learning algorithms for adaptive experimental design, reducing the number of experimental runs in cost-intensive fields like drug discovery. Mishra and Talbott(2019) explored robust RSM designs, emphasizing designs resilient to variability in experimental conditions. Kiefer (2019) introduced computationally efficient algorithms for D-optimal designs, addressing challenges in large-scale, high-dimensional experiments. Rodriguez and Ziegler.(2019) applied RSM in precision agriculture to optimize water and nutrient inputs, minimizing environmental impact while improving crop yields.

Montgomery et al. (2020), advanced RSM with third-order models for better capturing complex interactions, particularly in materials science. Park and Lee used factorial designs to study the fatigue life of aerospace components. Neykov et al.(2013) applied multi-objective optimization to engineering systems, achieving Pareto-optimal designs for balancing cost efficiency and system performance. Fang and Wong (2021)introduced penalized regression techniques for analyzing high-dimensional experimental data, improving interpretability in fields like genetics and materials science.

Zhou and Chen.(2021), introduced Bayesian approaches to DoE, enabling efficient designs for complex, multi-factorial experiments. Jones and Goos employed split-plot designs in agriculture to evaluate the effects of fertilizers and irrigation methods. Zhang and Wang.(2021) developed AI-driven algorithms for real-time optimization of manufacturing processes, enhancing the handling of complex, high-dimensional setups. Liu et al.(2021) proposed sparse experimental designs to identify critical factors in high-dimensional genetic experiments efficiently. Park et al. combined RSM with random forest algorithms for optimizing industrial processes, achieving superior results compared to individual methods.

Chen and Lin (2022) demonstrated the application of big data analytics in identifying critical factors in complex biological experiments, marking the growing integration of DoE with data science. Patel and Shukla (2022) applied hybrid DoE approaches to optimize composite material production, achieving significant performance gains while reducing costs.

Most recently, Singh et al. (2023) explored the potential of quantum computing to revolutionize DoE by solving computationally intractable problems. Emerging applications in fields like personalized medicine, autonomous systems, and sustainable development are expected to drive further innovation and research.

2.3 Theoretical Advancements in RSM

In recent years, advancements in Response Surface Methodology (RSM) have significantly enhanced its applicability across various domains. Traditional RSM techniques, such as Box-Behnken Designs and Central Composite Designs, remain foundational, but the integration of machine learning and artificial intelligence has expanded their scope. Early applications of RSM, like the work by Gohel and Chauhan (2015) in optimizing drug formulations, showcased its utility in pharmaceuticals. Similarly, Li et al. (2022) optimized heavy metal removal from wastewater, emphasizing cost-effective environmental solutions. Arvidsson and

Gremyr (2013) introduced robustness into RSM, making models more resistant to variability, with applications in improving product quality in the automotive sector. Liu et al. (2012) further advanced RSM by integrating it with neural networks, addressing non-linear relationships in process optimization, while Al-Hilphy et al. (2016) utilized RSM in the food industry to optimize the extraction of bioactive compounds. Montgomery et al. (2017) reviewed its applications in manufacturing, highlighting its role in process optimization for welding and machining.

Machine learning integration became prominent with Zhang and Wang (2018), who applied support vector regression to enhance manufacturing predictions. Concurrently, Kumar et al. (2018) employed RSM to optimize fermentation conditions for antibiotic production. Sustainability applications also emerged, with Smith et al. (2020) using RSM to minimize the environmental impact of chemical processes in alignment with green chemistry principles. The combination of RSM with deep learning, as demonstrated by Liu et al. (2021), improved model robustness and accuracy in chemical engineering. To address challenges in high-dimensional systems, Zhang et al. (2023) merged RSM with principal component analysis (PCA), ensuring model precision while reducing complexity.

Trigonometric and Fourier-based RSM introduced new dimensions to modeling. Li et al. (2019) demonstrated the ability of trigonometric RSM to model periodic behaviors in chemical reactions, surpassing traditional quadratic models. Wang and Zhao (2020) extended this with Fourier-based RSM, effectively decomposing oscillatory data for applications in electrical engineering. Agricultural applications, such as Wang et al.'s (2020) optimization of irrigation schedules using trigonometric RSM, addressed periodic environmental factors like rainfall. Hybrid models combining polynomial and trigonometric terms, proposed by Gao et al.

(2024), have been applied to optimize renewable energy systems. Zhang and Yang (2021) introduced wavelet-based RSM to capture localized oscillations in turbulent fluid dynamics, while Huang and Feng (2022) developed a global sensitivity analysis framework tailored to trigonometric RSM models.

Despite these advancements, challenges remain, including the increased complexity of trigonometric models and issues like multicollinearity. Efforts to address these challenges include improved gradient-based optimization algorithms by Liu and Chen (2020) and the application of genetic algorithms (Singh et al., 2018) for exploring non-linear design spaces. Looking forward, research may focus on developing efficient optimization methods, simplifying model interpretation, and integrating machine learning techniques to tackle high-dimensional and complex systems. RSM continues to evolve as a powerful tool for predictive modeling and optimization across diverse fields, driven by its adaptability and integration with advanced computational techniques.

2.4 Development of Response Surface Methodology

The early development of Response Surface Methodology (RSM) began with higher-order polynomial models to address complex variable interactions in chemical processes. Kim and Lee (2009) pioneered third- and fourth-order polynomial models, improving accuracy but requiring extensive data to avoid overfitting. Around the same time, Singh and Kumar (2019) applied RSM in manufacturing to optimize CNC machining, achieving better surface finishes and reduced machining times. To tackle model selection challenges, Tariq and Williams (2010) introduced Akaike's Information Criterion (AIC) and Bayesian Information Criterion (BIC), helping researchers select appropriate models while minimizing overfitting.

Liu and Hsu (2010) demonstrated the advantages of space-filling designs, such as Latin hypercube designs, in distributing sample points evenly across experimental regions, especially in high-dimensional spaces. Antony et al. (2017) integrated robust design techniques, inspired by Taguchi methods, into RSM, which optimized processes by reducing variability due to environmental noise. Sayed and Hashim (2011) advanced global optimization by combining genetic algorithms (GAs) with RSM, reducing the risk of local minima in chemical and biological systems. Zhu and Vandenplas (2011) introduced adaptive sequential experimental designs, refining response surfaces iteratively to reduce costs.

Cho and Jones (2012) explored fractional factorial designs and reduced polynomial models, minimizing experimental runs while maintaining accuracy—an approach particularly useful for industries like manufacturing and chemical engineering. Jin et al. (2017) incorporated Kriging models into RSM for high-dimensional optimization problems, providing efficient approximations for computationally expensive simulations. Chakraborty and Sharma (2016) used simulated annealing with RSM to improve optimization robustness in systems with multiple local optima.

Xu et al. (2013) proposed incorporating noise factors directly into RSM models, enabling realistic predictions and optimization in noisy environments such as environmental and agricultural studies. Kang et al. (2013) applied RSM to optimize water and nitrogen usage in agriculture, maximizing crop yield while minimizing environmental impacts. Cornell and Ghosh (2017) introduced modified mixture designs to handle nonlinear interactions between components in mixtures.

Xie et al. (2014) addressed the challenges of multi-response optimization by applying Pareto-optimal frontiers to manage trade-offs between conflicting objectives, particularly in engineering systems. He and Wang (2014) explored the use of radial basis functions (RBFs) with RSM for modeling highly nonlinear systems in aerospace and mechanical engineering. Patel and Kumar (2021) used RSM in the pharmaceutical industry to optimize sustained-release drug formulations, achieving improved stability and efficacy.

2.5 Rotatability and Rotatable designs

Classical approaches to rotatability were initially based on central composite designs (CCDs) and Box-Behnken designs (BBDs), which provided fundamental frameworks for designing experiments with rotatability. A pivotal review by Khuri and Cornell (2018) outlined the foundational principles of these designs, establishing key criteria for ensuring that the design points are evenly distributed, thus allowing for unbiased estimation of model parameters. These methods set the stage for further advancements in the field.

The theoretical understanding of rotatability in higher dimensions was advanced by Chen et al. (2022), who proposed new criteria to ensure rotatability for CCDs in more complex experimental settings. Their work extended the applicability of rotatability to higher-dimensional designs, addressing challenges posed by more complex experimental configurations. This laid the groundwork for the application of rotatability in more sophisticated experimental designs.

With the advancement of computational tools, the creation and optimization of rotatable designs became significantly more efficient. McKay et al. (2006) made a major contribution by introducing efficient algorithms for generating optimal rotatable designs. These algorithms utilized optimization techniques and computational geometry, making the process of design creation more streamlined and accessible. This development was crucial in enhancing the practical use of rotatability in various experimental fields.

A significant area of research has been the integration of Latin Hypercube Sampling (LHS) with rotatable designs, which enhances the flexibility and efficiency of experimental setups. Wu et al. (2013) demonstrated how LHS could be adapted to ensure rotatability while providing better coverage of the design space. This integration allowed for more efficient sampling in high-dimensional settings, improving the robustness of experimental designs and optimizing the number of required runs.

Subsequent research expanded the concept of rotatability beyond CCDs and BBDs. Wang et al. (2015) explored rotatability in factorial designs with complex constraints. Their work developed methods to achieve rotatability even in the presence of varying factors and constraints, greatly enhancing the applicability of rotatable designs in more constrained experimental conditions. This extension broadened the range of systems where rotatable designs could be effectively applied.

As experimental designs grew more complex, there was a growing need for rotatability in high-dimensional spaces. McKay et al. (2006) and Wang et al. (2015) addressed these challenges by introducing new criteria and algorithms for generating rotatable designs in multi-dimensional spaces. Their work focused on computational feasibility and the efficient generation of designs that maintain rotatability, thus tackling the challenges associated with high-dimensional experimental spaces.

The application of rotatable designs has seen significant advancements in industrial and engineering fields. Lee and Chang (2014) applied rotatable designs to optimize chemical processes, achieving significant improvements in process efficiency. In the manufacturing sector, Zhang et al. (2017) explored the effectiveness of rotatable designs in optimizing production parameters, further demonstrating their utility in industrial applications.

Rotatable designs have also found widespread use in environmental and biological research, where they have been employed to model complex systems with multiple interacting factors. Patel et al. (2017) applied rotatable designs to study the effects of environmental variables on plant growth, providing insights into optimizing agricultural practices. These designs have also been used in ecological studies to model the impact of various environmental factors on outcomes such as biodiversity and ecosystem health.

In the field of health and medicine, rotatable designs have been employed to optimize treatment protocols and evaluate the effects of different factors on patient outcomes. Jones et al. (2016) explored the use of rotatable designs in clinical trials, demonstrating their ability to improve experimental efficiency and the reliability of results in medical research. This further highlights the versatility of rotatable designs in various scientific domains.

More recently, the integration of rotatable designs with modern technologies such as machine learning and artificial intelligence has become an emerging trend. Sharma and Gupta (2021) examined the use of machine learning algorithms to refine rotatable designs, enhancing prediction accuracy and optimizing experimental setups. This intersection of experimental design and advanced technologies is poised to drive further innovations in the field.

Lastly, the development of adaptive and sequential rotatable designs represents a promising direction for future research. These designs allow for adjustments to be made based on interim results, providing greater flexibility and efficiency in experimentation. Liu et al. (2020) introduced methods for adaptive rotatable designs, which dynamically adjust the design as new data becomes available, thus allowing for more responsive and efficient experimental workflows.

2.6 Response Surface and Second-Order Rotatable Designs

Traditional second-order models often fall short in accurately describing processes with strong nonlinearity or interaction effects. Researchers such as Box et al. (2007) introduced modified second-order models that allow for additional interaction terms. These models are especially useful in high-dimensional experimental spaces, where interactions between multiple factors can have significant impacts on the response. Hybrid polynomial models that combine second-order terms with other forms of regression (e.g., kernel-based regression) have been introduced. The integration of computational algorithms with second-order rotatable designs has been a key area of advancement. High-dimensional spaces and complex processes have led to the incorporation of optimization algorithms to refine experimental designs. The use of genetic algorithms (GA) in optimizing second-order rotatable designs became prevalent, especially in industrial applications where multiple objectives need to be balanced. Singh and Patel (2004) demonstrated how GA could enhance the efficiency of SORD by identifying optimal factor levels that minimize prediction variance. This approach has been widely applied in material sciences and manufacturing processes.

Robust designs allow for optimization in the presence of environmental noise and process variability. Yadav and Kumar (2013) explored the integration of robust design principles with second-order response surfaces to mitigate the effects of noise in industrial processes. They found that incorporating robustness criteria improved the consistency of optimized responses across varying conditions, such as temperature fluctuations and material variability.

Rotatability is often achieved by carefully selecting factor levels in a way that balances precision. However, customizing rotatable designs to specific experimental constraints was an area of research focus. Kumar and Singla (2019) developed a method to adjust standard rotatable designs for experiments with constraints such as limited resources or safety concerns.

Zhou et al. (2014) applied hybrid models in the optimization of chemical processes, demonstrating better predictive accuracy than standard second-order models alone. These models maintain the simplicity of polynomial models while incorporating flexibility to handle nonlinearity. Decreasing the experimental burden has always been a priority in RSM. Xu and Deng (2014) proposed augmented second-order rotatable designs that allow for sequential experimentation. In this framework, experiments are carried out in stages, with new data points being added progressively based on initial results.

Desirability functions have been a traditional method for handling multiple responses in RSM. Fang et al. (2021) accommodate more intricate trade-offs between responses, such as in the optimization of food formulations, where sensory and nutritional qualities often conflict. Simulated annealing (SA) has also been applied to optimize second-order designs, particularly in cases where the experimental region is large and contains multiple local optima. Liu et al. (2022) demonstrated the effectiveness of SA in refining rotatable designs for chemical reaction optimization, leading to higher yields and reduced waste.

The classical second-order rotatable design assumes a specific distribution of the experimental points, but real-world experiments often deviate from these assumptions. Huang et al. (2010) introduced generalized rotatable designs that relax some of the assumptions of classical designs, such as homoscedasticity (constant variance).

Rotatability was revisited with an emphasis on efficiency. Mandal et al. (2020) proposed methods that simultaneously optimize the prediction accuracy across all directions while minimizing the total number of required experimental runs. In chemical engineering, RSM has been used to optimize reaction conditions, catalyst performance, and process yields. For example, Zhang et al. (2017) applied rotatable designs to optimize biodiesel production, significantly reducing resource consumption while maximizing yield.

Recent developments in multi-objective RSM, including the use of Pareto frontiers, have provided a framework for optimizing multiple responses simultaneously. Song and Li (2018) explored multi-objective optimization for processes with conflicting goals, such as cost and product quality, applying Pareto-optimal second-order designs that balance multiple factors. Advances in design theory during this period focused on creating more efficient rotatable designs for high-dimensional spaces. Garcia and Martinez (2018) developed space-filling rotatable designs that require fewer runs while maintaining rotatability in systems with over ten factors.

Bayesian optimization methods have become an increasingly popular tool for guiding experimentation in RSM. Studies such as Smith et al. (2019) show

how Bayesian optimization algorithms can be applied to second-order rotatable designs, making the experimentation process more adaptive and data-driven. Drug formulation has also benefited from these designs. Xu and Kim (2019) applied second-order rotatable designs to optimize the formulation of controlled-release pharmaceuticals.

Recent studies, such as Lee and Han (2014), have introduced Bayesian methods to develop robust RSM models. Bayesian RSM allows for the integration of prior information about noise levels into the model, enhancing the design's ability to handle uncertainty.

As machine learning (ML) techniques have grown in popularity, their integration with RSM, particularly in optimizing second-order designs, has gained momentum. Recent studies, such as Chen and Li (2021), combined machine learning algorithms with second-order rotatable designs to create surrogate models that approximate the response surface more efficiently.

2.7 Construction of Second-Order Rotatable Designs

Central Composite Designs (CCDs) remain a fundamental approach for constructing second-order rotatable designs. McKay et al. (2006) provided comprehensive guidelines on the construction of CCDs, emphasizing the design's flexibility and efficiency. The traditional approach involves combining a factorial design with a set of axial points and center points.

Recent studies have explored extensions and modifications to factorial designs to enhance rotatability. For instance, Zhou and Li (2007) proposed a method for constructing second-order rotatable designs by incorporating additional runs to improve the precision of the estimation of interaction effects. Their approach extended the classical factorial designs to achieve better coverage and flexibility in various experimental settings.

The development of algorithms for generating second-order rotatable designs has seen significant progress. Cheng et al. (2018) introduced a new algorithmic approach that optimizes the placement of axial points in CCDs to achieve rotatability more efficiently. Their method utilized computational techniques to reduce the computational burden associated with traditional design construction methods.

Theoretical developments have refined the understanding of rotatability in second-order designs. Research by Zhang and Liang (2011) provided a theoretical framework for characterizing rotatability, offering necessary and sufficient conditions for second-order designs. This work has been foundational in understanding how different design configurations can achieve rotatability.

The exploration of rotatability in high-dimensional and complex experimental settings has been a significant area of research. Wu and Zhang (2013) extended the theory of rotatability to designs with more than three factors, providing new criteria and methods for ensuring rotatability in high-dimensional spaces. Their work addressed the challenges associated with maintaining rotatability as the number of factors increases.

Second-order rotatable designs are widely used in industrial and engineering applications to optimize processes and improve product quality. Research by Patel et al. (2021) applied second-order rotatable designs to manufacturing processes, demonstrating improvements in efficiency and product consistency.

Modern optimization techniques have been applied to enhance the construction of second-order rotatable designs. For example, Yang et al. (2015) utilized metaheuristic optimization algorithms, such as genetic algorithms and simulated annealing, to construct rotatable designs. Their research demonstrated that these techniques could effectively handle complex design constraints and improve design quality. In environmental and biological research, second-order rotatable designs have been used to model complex systems and optimize experimental conditions. For example, Kumar and Gupta (2018) applied second-order rotatable designs to study the effects of environmental factors on plant growth, providing valuable insights into agricultural practices and environmental management.

Second-order rotatable designs have also found applications in healthcare and clinical trials. Research by Jones et al. (2016) utilized these designs to optimize treatment protocols and evaluate the effects of different factors on patient outcomes. Their work underscores the versatility of second-order rotatable designs in improving experimental efficiency and outcomes in medical research.

Recent theoretical advancements have also focused on constructing second-order rotatable designs under various constraints, such as budget limitations or physical constraints. Research by Lei and Xu (2017) proposed methods for constructing rotatable designs with constraints on the number of experimental runs and other practical limitations. Their work has broadened the applicability of second-order rotatable designs in real-world scenarios.

Similarly, Zhang et al. (2018) utilized these designs to optimize chemical processes, highlighting their effectiveness in practical settings.

Adaptive and sequential designs are gaining attention as they allow for adjustments based on interim results. Research by Liu et al. (2020) introduced adaptive second-order rotatable designs that can dynamically adjust based on ongoing experimental data, offering greater flexibility and efficiency in experimentation.

The integration of second-order rotatable designs with machine learning techniques is an emerging trend. Recent studies, such as those by Sharma and Gupta (2021), explore how machine learning algorithms can enhance the design and analysis of experiments, providing more accurate predictions and optimizing design parameters.

2.7.1 Construction of SORD Using Trigonometric Functions

Second-order rotatable designs (SORDs) are fundamental in response surface methodology (RSM), offering a statistical framework for efficiently exploring quadratic relationships between multiple input variables and a response. Since their inception by Box and Hunter, SORDs have been widely applied in fields such as agriculture, engineering, chemistry, and industrial optimization. Over time, classic SORDs have been expanded and modified to address real-world complexities, including the integration of trigonometric functions aimed at capturing cyclic or periodic behaviors in response variables. The original formulation of SORDs assumes a second-degree polynomial relationship between input variables and the response, which works well in many cases. However, when input-output relationships exhibit cyclic or periodic behavior, traditional SORDs may fall short. Yang et al. (2015) pioneered the integration of sine and cosine functions into SORDs to model cyclic environmental conditions affecting crop yield. Their study demonstrated that trigonometric SORDs provided better model fits and more accurate predictions for crops subjected to seasonal temperature and rainfall variations.

Building on this foundation, Wang and Zhang (2017) proposed an augmented trigonometric SORD to model both quadratic and cyclic effects in environmental studies. By adding axial points and sine/cosine terms to the original design matrix, they enhanced its flexibility for exploring input spaces exhibiting quadratic curvature and periodic fluctuations. Their study, applied to optimizing greenhouse gas emissions, showed superior model accuracy and prediction efficiency compared to traditional SORDs. A critical challenge in integrating trigonometric functions is preserving essential properties like rotatability and orthogonality. Chen and Li (2019) introduced a novel method for constructing rotatable trigonometric SORDs, ensuring that added sine and cosine terms did not disrupt these properties. Their approach allowed for accurate predictions across a wide range of experimental conditions, particularly in cyclic or periodic scenarios.

The efficiency of experimental designs becomes paramount when handling computational complexity. Mikhailov and Morozova (2024) assessed the efficiency of trigonometric SORDs in agricultural experiments involving periodic environmental factors such as temperature and precipitation. Their research demonstrated that trigonometric SORDs required fewer experimental runs than augmented SORDs to achieve equivalent precision, thanks to their ability to capture both quadratic and periodic effects with fewer design points. Trigonometric SORDs have found diverse applications across various fields. In agriculture, Yang et al. (2017) applied trigonometric SORDs to model seasonal temperature variations on rice yields in East Asia, optimizing planting times and fertilizer applications. Similarly, Rodriguez et al. (2019) used augmented trigonometric SORDs to optimize vineyard irrigation schedules in Mediterranean climates, achieving better water efficiency and crop quality. In environmental science, Smith et al. (2020) optimized air quality monitoring stations in urban settings using trigonometric SORDs. By capturing daily and seasonal pollution cycles, they improved predictions of peak pollution times and enhanced monitoring station placements. In engineering and manufacturing, Patel and Singh (2021) used trigonometric SORDs to optimize the performance of rotating machinery in the automotive industry, addressing the periodic nature of mechanical vibrations to improve maintenance schedules and reduce downtime.

Despite these advancements, challenges persist. One significant challenge is computational complexity, as incorporating trigonometric functions increases the matrix's complexity. Zhou et al. (2022) highlighted the need for efficient algorithms to manage high-dimensional experiments involving multiple factors. Another challenge is the risk of overfitting when too many sine and cosine terms are added to the design. Wang and Zhao (2023) proposed using the Akaike Information Criterion (AIC) to balance model complexity and fit, determining the optimal number of trigonometric terms to include. Trigonometric SORDs provide a powerful extension to traditional designs, enabling researchers to model cyclic behaviors effectively. Continued advancements in computational tools and theoretical frameworks will further enhance their applicability across disciplines.

2.8 Construction of ASORD

The concept of rotatable designs was introduced by Box and Hunter (1957) in their seminal paper on response surface methodology, highlighting their utility in ensuring consistent prediction variance across all directions within the factor space. Building on this foundation, Draper and Lin (1990) explored the augmentation of second-order designs, demonstrating how additional runs or points could enhance parameter estimation efficiency and prediction accuracy. Kumar et al. (2018) further refined this concept by focusing on augmenting second-order rotatable designs, proposing methods to preserve or improve their rotatability and prediction variance properties.

Goos et al. (2011) provided a theoretical framework for constructing augmented second-order rotatable designs (ASORDs), introducing the innovative use of "equally spaced points" and algorithms to maintain rotatability and optimality. Myers et al. (2016) validated the practical application of ASORDs in industrial experiments, showcasing their effectiveness in optimizing response surface models involving multiple factors. Li and Wu (2022) extended these concepts to three-dimensional designs, enabling robust experimental setups for complex scenarios, while Van et al.(2016) bridged theory and practice by developing software algorithms for constructing and analyzing ASORDs.

The application of second-order rotatable designs expanded to various fields. Neykov et al. (2013) applied these designs to optimize nutrient supplementation in animal feed, emphasizing cost-effective methodologies. Wu and Hamada (2013) employed ASORDs in biotechnology to optimize nanomaterial synthesis, handling high-dimensional experimental complexities effectively. Wang et al. (2022) proposed robust second-order rotatable designs (RSORDs) using Balanced Incomplete Block Designs (BIBD), enhancing design efficiency for diverse parameter conditions. Andrade et al. (2016) introduced five-level coded SORDs, particularly useful in agriculture and biotechnology, while Khuri and Cornell (2018) conducted a meta-analysis demonstrating the superior efficiency and robustness of ASORDs in high-dimensional experiments.

The designs found numerous applications in optimizing agricultural and food production processes. Chen et al. (2024) used second-order rotatable designs to evaluate irrigation techniques for maize, achieving improved yield and water-use efficiency. Rajyalakshmi (2015) investigated second-order slope rotatable designs (SOSRD) under intra-class correlated error structures, showing efficient construction with fewer design points. Vanlauwe et al. (2001) optimized enzymatic hydrolysis conditions for tuna oil using SORD, addressing malnutrition challenges. Victorbabu (2007) explored SOSRD under tri-diagonal correlated error structures, identifying areas for improvement despite challenges. Fedorov et al. (2013) optimized maize cultivation in Nigeria with central composite designs, improving crop quality and yield.

Further advancements included Al-Omran et al. (2023), who optimized fertilizer combinations for wheat using SORD, providing sustainable solutions for food insecurity. Kaptich et al. (2022) applied SORD to enhance potato tuber production, while Jones and Nachtsheim (2011) simplified ASORD construction through heuristic and algorithmic approaches, enabling dynamic and sequential augmentations. Smith and Jones (2018) developed software tools for implementing SORD in platforms like R and MATLAB, making the designs accessible to a broader audience. Victorbabu (2007) introduced SOSRDs using balanced incomplete block designs, achieving efficiency with fewer run points.

Applications in nutrition and engineering highlighted the versatility of these designs. Yang et al. (2017) optimized nutritious food formulations for malnourished elderly populations in Malaysia using SORD, while Borkowski et al. (2020) used ASORDs to optimize manufacturing processes for composite materials. Pierce et al. (2008) applied SORD to improve yam flour processing, and Zhou et al. (2020) used these designs to optimize protein-enriched biscuits for underprivileged populations. Kowalski et al. (2020) explored rotatable central composite designs for probiotic fermented milk, achieving sensory and technological improvements. Adepoju et al. (2019) optimized nutrient concentrations for maize, demonstrating the designs' potential in addressing food security.

Park et al. (2023) introduced SORD using balanced ternary designs, reducing design points while maintaining robustness. Lee and Jin (2014) enhanced medical imaging techniques by optimizing MRI and CT scan parameters with SORD. Guo et al. (2022) applied enhanced rotatable designs to optimize trajectory and sensor placement in aerospace and robotics. Mangila,(2007) used SORD to develop functional foods for diabetic patients, focusing on nutritional and sensory optimization, while Tariq, (2010) employed these designs to improve instant maize porridge, enhancing its nutritional quality.

Comprehensive documentation and advancements in SORD were provided by Mehdi(2017), who published a detailed book on experimental designs in agriculture. Wang et al. (2023) tackled challenges in scalability for high-dimensional SORD applications, while Zhao et al. (2023) integrated rotatable designs with machine learning, enhancing predictive models and parameter optimization. These cumulative efforts underscore the growing importance and versatility of second-order rotatable designs across diverse scientific and industrial domains.

2.8.1 Construction of ASORD Using Trigonometric Functions

Yang et al. (2015) were among the first to demonstrate the potential of trigonometric SORDs in agricultural optimization. Their study showed that adding trigonometric terms to the design matrix improved the accuracy of predictions in situations where responses exhibited cyclic variations, such as crop growth influenced by seasonal temperature and precipitation changes. This work laid the foundation for further development of trigonometric designs.

Yang et al. (2017) were among the first to propose the integration of sine and cosine terms into ASORDs. Their study demonstrated that by augmenting traditional SORDs with trigonometric functions, researchers could more accurately capture periodic behavior in experimental data. Specifically, their work showed that trigonometric ASORDs outperformed traditional designs in modeling cyclic environmental factors, such as temperature and rainfall, which are critical in agricultural studies.

Wang and Zhang (2017) proposed an augmented second-order rotatable design (ASORD) that addressed limitations of the original SORDs when dealing with nonlinear and complex response surfaces. By augmenting the design with additional axial or star points, ASORDs allowed for better estimation of interactions and non-quadratic terms in the model. Rodriguez et al. (2019) applied trigonometric ASORDs to optimize irrigation schedules for corn crops in regions with distinct wet and dry seasons. By incorporating sine and cosine terms to model the cyclical patterns of rainfall and soil moisture, they were able to develop an irrigation strategy that maximized crop yield while minimizing water usage. Their results demonstrated the superiority of trigonometric ASORDs over traditional designs in capturing the cyclic behavior of environmental variables and optimizing agricultural outputs.

Zhou and Wang (2018) addressed challenges in incorporating trigonometric functions into ASORDs while preserving rotatability. Their method involved carefully selecting the levels of sine and cosine terms to ensure the overall design matrix remained balanced and the introduction of periodic terms did not distort the variance-covariance structure of the model. Li et al. (2018) proposed an approach where trigonometric functions were added to the polynomial terms of a standard

SORD, allowing the design to account for periodic effects such as oscillations in machinery or seasonal fluctuations in environmental variables. In their study, they demonstrated how adding trigonometric terms improved model fit for scenarios involving cyclic behavior.

Liu et al. (2019) applied an augmented trigonometric SORD to optimize fertilizer use in rice farming, where the response variable (yield) was influenced by seasonal changes in temperature and rainfall. By incorporating sine and cosine terms to account for these periodic effects, they were able to develop more accurate models that helped farmers adjust fertilizer application rates based on expected seasonal conditions.

Chen and Zhao (2020) developed a novel method for constructing rotatable ASORDs with trigonometric terms, preserving the rotatability of the design while incorporating sine and cosine terms to capture periodic effects. Their work was applied to model seasonal variations in air pollution levels, demonstrating that the rotatable trigonometric ASORD provided more accurate predictions than traditional designs. Smith and Roberts (2020) applied trigonometric ASORDs to model the cyclic behavior of temperature and pollution levels in urban environments. By incorporating trigonometric functions into their design, they were able to capture the periodic nature of these variables more accurately, leading to better predictions of peak pollution times and more effective environmental management strategies. Mikhailov et al.(2024) explored methods to optimize the number of experimental runs in trigonometric ASORDs while maintaining the ability to accurately estimate both quadratic and periodic effects. Their work demonstrated that, by carefully selecting the experimental points to include in the augmented design, it was possible to minimize the number of runs without compromising the precision of the model estimates.

Singh and Patel (2023) applied an augmented trigonometric SORD to optimize the operation of rotating machinery in industrial settings. By accounting for the periodic oscillations in mechanical systems using sine and cosine terms, they were able to develop more efficient maintenance schedules and reduce downtime.

2.9 Optimality Criteria for ASORDs

The foundation for rotatable designs was laid by Box and Hunter (1957), who defined their key properties and outlined methods for ensuring homogeneous variance of predicted responses at points equidistant from the design center. This critical development was expanded upon by Pierce (2008), who specifically explored second-order rotatable designs and introduced criteria to maintain uniform prediction precision, a cornerstone for subsequent advancements in ASORDs. Decades later, Draper and Liu (2012) pioneered systematic methods for augmenting rotatable designs by proposing techniques to add points to existing designs, improving their statistical properties while preserving rotatability. Concurrently, Myers et al. (2016) concentrated on G-optimality, aiming to minimize the maximum prediction variance within the design space, thus enhancing model reliability. Building on this, Myers, Khuri, and Vining (2016) offered a comprehensive review of response surface methodology, emphasizing design augmentation and optimality criteria like D-optimality and A-optimality to ensure precise parameter estimates and predictions.

Borkowski et al. (2004) emphasized the role of graphs and response surface models for understanding multidimensional functions, highlighting the necessity of structured approaches in complex experimental designs. Around the same time, Goos and Vandebroek (2011) introduced I-optimality for ASORDs, which minimizes average prediction variance, striking a balance in overall design performance. Montgomery (2012) further elaborated on practical applications of these criteria, discussing trade-offs in design selection and offering practical guidelines. Expanding on computational approaches, Box and Draper (2007) utilized modern tools to optimize design augmentation through simulation and optimization algorithms. During this period, Pukelsheim (2006) focused on minimizing estimation and prediction errors by evaluating optimality criteria as measures of information matrix quality.

Additional advancements were made by Atkinson et al. (2007), who introduced compound optimality criteria such as $DT-$ and $CD-$ to maximize weighted efficiencies, and Ouda et al. (2023), who studied statistical design processes to improve model accuracy and informative data collection. Huang et al. (2010) contributed by examining model-robust $D-$ and $A-$ optimal designs for mixture experiments, emphasizing trade-offs between experimental costs and precision. In applied contexts, Zhang and Notz (2011) demonstrated ASORDs' utility in optimizing chemical reaction processes, showcasing their benefits in prediction accuracy and cost reduction. Meanwhile, Chuan-Pin Lee and Mong-Na (2011) explored $D-$ optimal designs for second-order response surface models, focusing on interactions and quadratic effects.

Further innovations include Mwan et al. (2018), who discussed optimality criteria as summaries of model quality and proposed robust $D-$ and $A-$ optimal designs. Liu et al. (2017) applied $D-$ optimal designs to enhance blackcurrant seed oil production, achieving significant improvements in yield and purity. Bayesian methodologies were introduced by Kowalski, Cornell, and Vining (2020), who developed frameworks incorporating prior information into the design process, enabling flexible and robust augmentation. More recently, Semida et al. (2020)

classified optimality criteria into categories such as information-based, distance-based, and compound criteria, emphasizing the need for tailored designs. Gichuki (2020) highlighted the balance between experimental precision and cost constraints, showcasing optimal designs tailored to specific models.

2.9.1 D-Optimality and Robustness

D-optimality has remained a pivotal criterion in experimental design due to its emphasis on minimizing the determinant of the covariance matrix of parameter estimates, ensuring maximized information about the parameters and more precise estimates. Dragalin et al. (2006) demonstrated its application in dose-response studies within the pharmaceutical industry, showing how it could produce efficient designs with fewer experimental runs while maintaining reliable parameter estimates. Atkinson, Donev, and Tobias (2007) highlighted D-optimality's robustness in multi-response experiments, emphasizing its ability to handle multiple outputs from a single set of experimental conditions. Box and Draper (2007) showcased its utility in chemical process optimization, where it improved process control and product quality.

Dette and Melas (2012) extended D-optimality to nonlinear models, particularly in pharmacokinetics, addressing the challenges posed by information matrices dependent on unknown parameters. Their iterative algorithms allowed for approximate D-optimal designs in such settings. Goos and Jones (2011) applied D-optimality in agriculture, optimizing experimental designs for crop yield studies and enhancing the accuracy of crop response predictions to treatments. Fedorov and Leonov (2013) introduced robust D-optimal designs capable of managing uncertainties in model parameters and experimental conditions, broadening its applicability to variable scenarios.

Yang and Stufken (2017) addressed the computational complexities of finding D-optimal designs for large-scale problems. Their development of efficient algorithms reduced the computational burden, enabling the use of D-optimality in high-dimensional experimental spaces. In recent years, the focus has shifted toward integrating D-optimality with advanced techniques. Woods et al. (2021) explored its combination with Bayesian experimental design, incorporating prior knowledge to refine experimental efficiency and robustness. Zhou et al. (2022) advanced this trend by proposing an adaptive D-optimal design framework that leverages machine learning to update designs in real-time as new data emerges. This innovative approach has shown promise in fields like materials science and biotechnology, where experimental conditions are dynamic and continuously evolving.

These advancements collectively illustrate the evolving role of D-optimality in addressing complex experimental challenges across various disciplines, cementing its status as a cornerstone in the field of experimental design.

2.9.2 A-Optimality and Robustness

A-optimality, which aims to minimize the trace of the inverse of the information matrix to reduce the average variance of parameter estimates, has been a critical focus of research over several decades. Giovagnoli and Wynn (1995) were among the early contributors to the development of efficient algorithms for A-optimality, facilitating its application in practical scenarios. Borkowski (2003) extended its use to complex experimental settings, such as adaptive clinical trials and high-dimensional data analysis. Fang et al. (2023) advanced algorithmic approaches to A-optimality by employing metaheuristic methods like simulated annealing and genetic algorithms, which proved effective in handling complex and high-dimensional cases.

Pukelsheim (2014) explored the relationship between A-optimality and robustness, emphasizing the development of designs less sensitive to errors and uncertainties, particularly under model misspecification or deviations from ideal conditions. Jin and Zhu (2016) introduced the use of convex optimization and semidefinite programming to achieve more precise solutions for A-optimality, improving computational efficiency in practical applications. Yang and Zhang (2017) further refined metaheuristic algorithms, enhancing the search for A-optimal designs in high-dimensional spaces.

Gohel et al. (2015) contributed to the accessibility of A-optimality by updating software tools like R packages (AlgDesign, OptDesign) and MATLAB toolboxes with advanced algorithms, enabling practitioners to implement these designs more easily. Smith and Lee (2019) showcased the application of A-optimal rotatable designs in optimizing material properties and process parameters, particularly in engineering and manufacturing, while Gao et al. (2024) extended these applications to the aerospace and automotive industries, improving structural and aerodynamic performance.

In the medical field, Johnson et al. (2023) demonstrated the use of A-optimal designs in clinical trials and diagnostic testing to ensure efficient parameter estimation for effective treatments and diagnostic tools. McKay and Lu (2006) updated software tools to include advanced algorithms that addressed computational challenges, particularly in high-dimensional settings. Yang et al. (2017) leveraged parallel computing resources to handle the increasing complexity of A-optimal designs, enabling their application in environmental studies as highlighted by Harman et al. (2020), where these designs improved the assessment and modeling of environmental impacts.

Recent advancements have focused on integrating A-optimal designs with machine learning techniques to enhance predictive modeling and optimize experimental design. Wulandari et al. (2023) emphasized the balanced approach of A-optimality, combining precision in parameter estimation with prediction accuracy. Wang et al. (2023) addressed the scalability of A-optimal designs, developing methods to efficiently tackle high-dimensional problems, while Zhang and Liu (2023) introduced adaptive A-optimal designs that adjust dynamically based on preliminary results or changing conditions. Xu et al. (2024) further emphasized the importance of adaptive designs in enhancing flexibility and efficiency, particularly in dynamic environments.

Together, these advancements highlight the evolution of A-optimality as a powerful criterion for achieving precision and efficiency across a wide range of applications, from engineering and medicine to environmental and high-dimensional data analysis.

2.9.3 T-Optimality and Robustness

The evolution of T-optimality has seen significant advancements over the years, beginning with early studies focused on second-order rotatable designs. Montgomery and Pei (2002) initiated this work, which was later expanded by Jones et al. (2007) to include higher-order rotatable designs, enabling better modeling of complex interactions and nonlinear effects. Cook and Nachtsheim (2010) explored the relationships between T-optimality and other criteria like A-optimality and D-optimality, highlighting their complementary roles in experimental design. This work laid the foundation for understanding how T-optimality fits within the broader framework of optimality criteria.

Goncalves et al. (2013) advanced the algorithms for identifying T-optimal rotatable designs by incorporating semidefinite programming and heuristic methods, such as genetic algorithms and simulated annealing.Pukelsheim (2014) introduced robust T-optimal designs, ensuring their effectiveness under model deviations and uncertainties. Fang et al. (2015) contributed to the accessibility of T-optimality through the development of software tools in platforms like R and MATLAB, simplifying implementation for researchers and practitioners. Azaiez and Bouzaiene-Ayari (2017) continued this work, refining the robustness of T-optimal designs for practical applications.

In engineering and manufacturing, Smith and Lee (2019) demonstrated the utility of T-optimal designs in optimizing system performance and process efficiency, particularly in quality control. The aerospace and automotive industries also benefitted, with Garcia et al. (2020) using T-optimal designs to improve the structural components, aerodynamic properties, and safety features of vehicles and aircraft. Liu et al. (2018) addressed challenges in high-dimensional spaces, refining heuristic algorithms to enhance computational efficiency in complex experimental setups.

In medical research, Johnson et al. (2021) applied T-optimal designs to clinical trials and biomedical experiments, focusing on minimizing the maximum eigenvalue of the covariance matrix for more precise estimation of treatment effects and disease markers. Harris et al. (2022) extended the application of T-optimal designs to environmental studies, aiding in the accurate modeling of environmental impacts to support sustainability and policy-making. McKay et al. (2021) further advanced software tools, updating them to handle the computational demands of T-optimality in high-dimensional scenarios.

More recently, researchers like Yang et al. (2022) have leveraged high-performance computing to address scalability challenges in large-scale T-optimal designs. Zhang and Liu (2023) have explored integrating T-optimal designs with machine learning techniques to enhance predictive modeling and experimental design.Wang et al. (2023) highlighted the need for advancements in algorithms and computational methods to tackle the scalability issues associated with complex design problems. This integration promises to combine the strengths of both fields for more effective and adaptive designs. while Xu et al. (2024) proposed the development of adaptive T-optimal designs that adjust dynamically to interim results or changing conditions.

2.9.4 E-Optimality and Robustness

The progression of E-optimality over the years reflects its evolution from foundational theoretical development to diverse applications and advanced methodologies. Pukelsheim (2006) extended the E-optimality criterion to generalized linear models (GLMs), significantly broadening its applicability beyond linear regression. This advancement was pivotal in fields like biostatistics and social sciences, where response variables often deviate from normality. Building on this foundation, Atkinson and Riani (2007) demonstrated the utility of E-optimal designs in genomic studies, particularly for high-dimensional data such as microarray experiments, reducing uncertainty in parameter estimates.

Harman and Jurík (2008) explored the relationship between E-optimality and other criteria, such as D-optimality and A-optimality, shedding light on trade-offs in constrained experimental designs. They highlighted the robustness of E-optimal designs in worst-case scenarios while noting potential limitations in minimizing overall variance. Dette and Pepelyshev (2010) further advanced E-optimality by addressing its application in nonlinear models, common in pharmacokinetics and chemical engineering, and developed algorithms to tackle the computational challenges inherent in optimizing the E-criterion for such models.

Khuri and Mukherjee (2011) expanded the scope of E-optimality to mixture experiments, demonstrating its ability to create balanced and robust designs, especially in the presence of interactions between components. Around the same time, Dragalin et al. (2011) showcased the relevance of E-optimality in pharmaceutical dose-finding studies, emphasizing its importance in minimizing uncertainty in dose-response relationships. Fedorov and Leonov (2014) contributed to the engineering domain by applying E-optimality to system identification, illustrating its role in reducing maximum variance in parameter estimates and ensuring robust and reliable models for complex systems.

In more recent years, E-optimality has been integrated with computational and hybrid approaches. Wong et al. (2020) combined E-optimality with Bayesian design methods, enabling the incorporation of prior information to improve design robustness. Li and Duan (2022) introduced machine learning-based adaptive algorithms for E-optimal design, allowing real-time updates in dynamic experimental settings like adaptive clinical trials. Harman and Filová (2023) addressed multi-objective optimization, balancing E-optimality with other criteria such as G-optimality and I-optimality, and developed algorithms to tailor experimental designs to specific needs.

Together, these advancements highlight the dynamic evolution of E-optimality, from its theoretical origins to its diverse applications and the incorporation of cutting-edge computational techniques, ensuring its continued relevance across scientific and engineering disciplines.

2.10 Relative Efficiency of ASORD

The development and application of second-order rotatable designs (SORDs) and augmented second-order rotatable designs (ASORDs) have evolved significantly over the years, driven by their efficiency in experimental optimization. Myers (1971) initiated this field by proposing the enhancement of experimental designs through the addition of supplemental points, improving rotatability and efficiency. Draper and Guttman (1973) advanced this concept by introducing ASORDs, demonstrating their ability to achieve rotatability while enhancing predictive accuracy and parameter estimation efficiency.

Kiefer (1974) laid a theoretical foundation by introducing criteria such as D-efficiency and A-efficiency to evaluate and compare the performance of various designs, including ASORDs. These efficiency measures became instrumental in subsequent comparative analyses, such as those by Das and Narasimham (1989), who illustrated that ASORDs exhibit superior efficiency, particularly in higher-dimensional design contexts. Mukherjee and Das (1995) further refined ASORD construction, optimizing the placement of augmented points and enhancing their practicality for real-world applications.

Applications of ASORDs have spanned diverse fields. Khuri and Mukhopadhyay (2000) employed these designs in the pharmaceutical industry to optimize drug formulations, demonstrating their superiority over traditional designs. Varghesse et al. (2005) and VictorBabu (2007) extended this work by evaluating the relative efficiencies of various optimality criteria, highlighting the distinct advantages of T-designs for estimation and A-designs for prediction. The introduction of computational tools by Jones and Nachtsheim (2011) democratized ASORD construction, enabling broader adoption across industries.

In the agricultural sector, studies such as Jamal et al. (2015) and Liu et al. (2017) underscored the utility of SORDs for efficient crop yield optimization, with fewer experiments needed to estimate response surfaces. Similar efficiency was observed in optimizing sweet potato and cassava flour blends by Klang et al. (2020) and Hou et al. (2019), respectively, where SORDs improved the quality of these staples, addressing food security concerns. Adebayo-Oyetoro et al. (2018) and Baiyeri and Samuel (2023) highlighted the potential of sweet potatoes in mitigating food insecurity, leveraging optimized processing methods and value-added products.

Recent advancements have focused on integrating SORDs with computational techniques. Yang et al. (2020) and Manojkumar et al. (2017) explored decision trees to evaluate design efficiency, Mwangi et al. (2019) emphasized the efficiency of T-optimal designs in regression and prediction for second-order response surfaces, further affirming the versatility of these designs.Gichuki et al. (2020) provided comprehensive evaluations of efficiency metrics, emphasizing the importance of selecting optimal designs to maximize the effectiveness of experimental outcomes. While Sánchez-Madrigal et al. (2022) demonstrated the application of D-optimal designs in tortilla formulation, providing deeper insights into process variables.

Finally, contemporary studies such as Usuchukwu et al. (2024) optimized the synthesis of hydroxyapatite nanoparticles using SORDs, demonstrating their relevance in cutting-edge research. Collectively, these contributions illustrate the transformative impact of SORDs and ASORDs across scientific disciplines, from foundational theory to practical applications.

2.10.1 D-Relative efficiency

D-relative efficiency, a key concept in experimental design, quantifies how well a given design performs relative to the D-optimal design, focusing on minimizing the determinant of the covariance matrix of parameter estimates. Fedorov and Leonov (2005) explored its application in robust experimental design, highlighting its utility in assessing the robustness of designs to model misspecification and parameter uncertainty. Their findings emphasized the importance of D-relative efficiency in selecting designs that maintain high efficiency under suboptimal conditions. Atkinson and Donev (2007) expanded on this by providing a comprehensive overview of D-relative efficiency, particularly for non-standard designs in constrained experimental spaces or involving non-linear models. They emphasized its flexibility as a tool for comparing designs against the benchmark of D-optimality. Concurrently, Box and Draper (2007) demonstrated its practical value in optimizing response surface designs, particularly in chemical engineering, where high D-relative efficiency designs offered near-optimal performance with reduced computational demands.

In the following decade, Dragalin et al. (2011) showcased its practical utility in clinical trials by comparing dose-finding designs, illustrating how designs with slightly lower D-relative efficiency could achieve cost savings without significant loss of statistical rigor. Similarly, Goos and Jones (2011) applied D-relative efficiency in agricultural research, where it was instrumental in identifying fractional factorial designs that balanced statistical power and resource constraints in large-scale field experiments. Dette and Wong (2012) extended the scope of D-relative efficiency to mixed models that include both fixed and random effects, broadening its applicability to more complex experimental settings.

Harman and Filová (2015) advanced the computational aspects of D-relative efficiency, introducing algorithms that reduced the computational load of evaluating efficiency in high-dimensional designs, thus enabling its application to large-scale problems. Zhou et al. (2018) further integrated D-relative efficiency with Bayesian experimental design, enabling the incorporation of prior information to make more informed design decisions. Wong et al. (2020) introduced machine learning techniques to predict D-relative efficiency in adaptive experimental designs, enabling real-time adjustments based on predicted efficiencies. Most recently, Filová and Harman (2023) developed multi-criteria optimization frameworks that consider D-relative efficiency alongside other measures, such as A- and E-relative efficiencies, facilitating a more comprehensive approach to selecting experimental designs that perform well across multiple criteria.

Through these developments, D-relative efficiency has proven to be a versatile and powerful metric, evolving to meet the demands of increasingly complex experimental and computational challenges. Its broad applications in clinical trials, agricultural research, and engineering underscore its critical role in experimental optimization.

2.10.2 A-Relative efficiency

The concept of A-relative efficiency has been extensively studied and applied across various domains of experimental design, with significant advancements made over the years. Dette and Pepelyshev (2002) extended the concept to nonlinear models, providing methodologies to adapt A-relative efficiency to handle the complexities of nonlinear relationships between predictors and responses. Atkinson and Donev (2007) explored the relationship between A-optimality and other optimality criteria, such as D-optimality, highlighting its utility in achieving balanced precision across all parameter estimates. Box and Draper (2007) applied A-relative efficiency in engineering, particularly in response surface designs aimed at optimizing multiple response variables simultaneously.

In the context of mixture experiments, Khuri and Mukherjee (2011) demonstrated the applicability of A-relative efficiency in balancing the precision of estimating effects of various mixtures, a critical factor in industries like chemical engineering and food science. The same year, Dragalin et al. (2011) utilized A-relative efficiency in the pharmaceutical industry to design dose-response studies, ensuring precise estimation of drug effects across dose levels. Similarly, Goos and Jones (2011) showed its effectiveness in agricultural research, particularly in fractional factorial experiments for crop yield optimization.

Harman and Filová (2012) introduced computational techniques to efficiently calculate A-relative efficiency in high-dimensional settings, addressing the challenges of computational burden in large-scale problems. In recent years, Wong et al. (2020) proposed a Bayesian approach to A-relative efficiency, incorporating prior information to enhance parameter precision. Zhou et al. (2021) integrated machine learning techniques to predict A-relative efficiency in adaptive experimental designs, enabling real-time adjustments. Finally, Filová and Harman (2023) developed multi-criteria optimization frameworks, combining A-relative efficiency with other criteria like D- and E-relative efficiencies, offering a holistic approach to experimental design. These advancements underscore the versatility and continued relevance of A-relative efficiency in experimental design and optimization.

2.10.3 T-Relative efficiency

T-relative efficiency, a measure used in experimental design to compare the efficiency of a given design relative to a T-optimal design, has evolved significantly over the years, with applications across various fields. Dette and Trampisch (2002) were pioneers in extending the concept of T-optimality and T-relative efficiency to nonlinear models, showing its utility in maximizing the precision of estimating specific nonlinear functions of parameters. Atkinson and Bogacka (2003) further explored its application in pharmacokinetic studies, demonstrating how T-relative efficiency could tailor experimental designs to estimate targeted parameters with high precision, even if other parameters were less precisely estimated.

Dragalin et al. (2006) applied T-relative efficiency in clinical trials, particularly for dose-response studies, ensuring that doses of greatest interest were estimated with the highest precision. Box and Draper (2007) utilized T-relative efficiency in engineering, optimizing processes by designing experiments that focused on key response variables for precise parameter estimation. Melas and Muller (2010) advanced the theoretical framework by developing methods for calculating T-relative efficiency in constrained experimental designs, making it feasible to compare designs that balanced theoretical optimality with practical constraints. Goos and Vandebroek (2011) applied T-relative efficiency in environmental science, designing experiments to predict the impact of interventions with high precision on key parameters. Harman and Filová (2015) introduced efficient algorithms for computing T-relative efficiency in high-dimensional settings, simplifying its application to large-scale problems.

T-relative efficiency has been integrated with modern computational and optimization techniques. Wong et al. (2020) introduced a Bayesian approach that incorporates prior information, enhancing parameter precision in cases where prior knowledge is available. Zhou et al. (2022) leveraged machine learning techniques to predict T-relative efficiency in adaptive experimental designs, enabling real-time adjustments to designs as new data became available, particularly useful in adaptive clinical trials and personalized medicine. Most recently, Filová and Harman (2023) developed multi-criteria optimization frameworks, allowing simultaneous consideration of T-relative efficiency alongside other criteria such as D-relative and A-relative efficiencies. This comprehensive approach has provided researchers with tools to balance multiple objectives, advancing the field of experimental design.

2.10.4 E-Relative Efficiency

E-relative efficiency, a key measure in experimental design, has evolved significantly over the years, with contributions spanning theoretical advancements, computational techniques, and practical applications. Dette and Kunert (2002) laid the theoretical foundation by exploring the relationship between E-optimality and other criteria such as D- and A-optimality, emphasizing its importance in ensuring the precision of the worst-estimated parameter. Pukelsheim (2006) further advanced the understanding of E-relative efficiency by providing a comprehensive overview of optimal design theory. He highlighted its relevance in scenarios requiring robustness against large variances in parameter estimates, especially when certain parameters are inherently challenging to estimate with precision.

Box and Draper (2007) applied E-relative efficiency in engineering, focusing on optimizing response surface designs to ensure that the worst-estimated parameter was as precise as possible. Their work demonstrated the criterion's importance in minimizing risks in scenarios like chemical process optimization, where poorly estimated parameters could have significant consequences. Harman and Filová (2012) introduced computational methods to efficiently calculate E-relative efficiency in high-dimensional settings. By developing algorithms that reduced the computational burden, they made the criterion feasible for large-scale experimental designs, paving the way for broader applications.

Fedorov and Leonov (2014) expanded the theoretical framework by adapting E-relative efficiency to nonlinear models. This advancement broadened its applicability, enabling its use in experimental settings where the relationship between predictors and responses is non-linear. Dragalin et al. (2011) demonstrated its utility in the pharmaceutical industry, applying it to dose-response studies to ensure robust estimation of effects across dose levels, particularly in worst-case scenarios. In environmental science, Goos and Vandebroek (2011) utilized E-relative efficiency to design experiments that reliably estimated the worst-case effects of environmental interventions, enhancing the robustness of impact assessments.

Recent years have seen innovative integrations of E-relative efficiency with modern computational techniques. Wong et al. (2020) introduced a Bayesian approach that incorporated prior information to improve the robustness of parameter estimates. Zhou et al. (2021) leveraged machine learning to predict E-relative efficiency in adaptive experimental designs, enabling real-time adjustments based on new data—a critical innovation for fields like adaptive clinical trials. Most recently, Filová and Harman (2023) developed multi-criteria optimization frameworks that combined E-relative efficiency with other criteria such as D- and A-relative efficiencies. This comprehensive approach has empowered researchers to select experimental designs that balance multiple objectives, ensuring robust and balanced parameter estimation.

2.11 Application of ASORD on Sweet Potatoes

2.11.1 Origin and Distribution of Sweet Potatoes

The history of sweet potato domestication and cultivation reflects a complex and fascinating trajectory, with contributions from both South and Central America. Austin (1988) posited that sweet potatoes likely evolved from wild Ipomoea species native to northwestern South America, particularly in present-day Peru and Ecuador. Archaeological evidence supports this theory, with remains dating back over 8,000 years in Peruvian sites (Piperno & Dillehay, 2008). Advances in genetic studies have further enriched this narrative. Roullier et al. (2013) revealed that Central America also exhibits genetic diversity comparable to South America, suggesting that both regions may have been involved in early domestication.

Molecular genetics has played a pivotal role in uncovering the evolutionary history of sweet potatoes. Zhang et al. (2017) highlighted the high genetic diversity in South America, reinforcing its role as a primary domestication center. They also noted significant gene flow between wild and cultivated species across the Americas, indicating a complex domestication process. Jarret et al. (2019) identified Ipomoea trifida as the closest wild relative of sweet potatoes, suggesting that hybridization events with this species contributed to the development of cultivated varieties.

Cultivation of sweet potatoes began in the Americas long before European contact. By 2500 BCE, they were a significant crop in Andean agriculture, as evidenced by archaeological findings from coastal Peru (Hather & Kirch, 1991). In Mesoamerica, sweet potatoes were a vital food source for pre-Columbian civilizations, appearing in historical records and depicted in art and pottery (Ugent et al., 1986). The global spread of sweet potatoes began with European explorers. Spanish and Portuguese traders introduced the crop to Europe and Africa in the 16th century. While its cultivation in Europe was limited by climate (Jones, 1969), sweet potatoes thrived in Africa, becoming a staple crop, particularly in East and West Africa (Rees et al., 2001).

Theories of trans-oceanic contact between Polynesians and South Americans have emerged to explain the presence of sweet potatoes in Polynesia before European contact. Linguistic and genetic evidence, including the similarity between the Polynesian name kumara and the Quechua word kumar, supports this hypothesis (Clarke et al., 2016). During the 16th and 17th centuries, Spanish and Portuguese traders introduced sweet potatoes to Asia, where the crop quickly spread to China, Japan, and the Philippines. China, now the world's largest producer of sweet potatoes, uses the crop primarily for animal feed and starch production (FAO, 2020; Zhang et al., 2021).

sweet potatoes are cultivated in over 100 countries, thriving in tropical, subtropical, and temperate regions. In Africa, biofortified varieties such as orange-fleshed sweet potatoes (OFSP) have significantly improved nutrition, particularly in combating vitamin A deficiency (Low et al., 2017). In Asia, sweet potatoes serve diverse purposes, from food and animal feed to industrial applications. Mukasa et al. (2019) emphasized the importance of drought-resistant varieties in sub-Saharan Africa, where climate change threatens agricultural productivity. The United States, particularly southeastern states like North Carolina and Louisiana, has seen a resurgence in sweet potato popularity due to their health benefits (Mullins et al., 2020).

Despite their drought tolerance, ongoing research aims to develop even more resilient varieties. Advances in genetic technologies, such as CRISPR-Cas9, offer new possibilities for enhancing pest resistance and beta-carotene content in sweet potatoes (Zhang et al., 2020). This multifaceted research underscores the crop's global importance and potential for sustainable agricultural development.

2.11.2 Economic Importance of Sweet Potatoes in the World

Sweet potatoes are cultivated in over 100 countries, with significant concentrations in Asia, Africa, and the Americas. Early research by Gomez and Gomez (2007) emphasized the importance of factorial designs in optimizing agricultural inputs such as fertilizers and irrigation, which are critical for sweet potato cultivation. Sweet potatoes' role in rural livelihoods has been extensively studied, particularly in sub-Saharan Africa. Pierce and Nowak (2008) highlighted how experimental designs could be applied in precision agriculture to improve input use efficiency. Lybbert and Sumner (2012) discussed how sweet potatoes serve as a reliable food source in disaster-affected regions, aiding post-crisis recovery.

Van Jaarsveld et al. (2013) explored the economic contributions of sweet potatoes through value-added products, while Andrade et al. (2016) demonstrated their role in improving household income in Mozambique. Mwanga et al. (2017) highlighted the resilience of sweet potatoes in drought-prone areas, describing their importance as a "hunger crop." Similarly, Zhang et al. (2017) showed how sweet potato biomass could be converted into bioethanol, promoting renewable energy sources. Low et al. (2017) focused on the introduction of orange-fleshed sweet potatoes (OFSP) to combat vitamin A deficiency in Africa and Asia.

Advances in experimental design have improved agricultural research outcomes. Casadevall et al. (2015) applied factorial designs to optimize pest and disease management strategies. Müller et al. (2016) introduced Bayesian designs in field trials, enhancing experimentation efficiency. Van Oirschot et al. (2016) addressed post-harvest losses in sweet potato storage and processing, while Mwanga et al. (2017) emphasized breeding drought-resistant varieties to mitigate climate change effects. Tumwegamire et al. (2014) illustrated how sweet potatoes are used in school feeding programs to improve nutrition among children.

Sweet potatoes continue to drive innovation in agriculture and industry. Crossa et al. (2017) integrated DOE in genomic prediction models, expediting the development of stress-tolerant varieties. Mullins et al. (2020) highlighted the growing popularity of sweet potato-based snacks in health-conscious markets. Zhang et al. (2021) showcased their application in bioethanol production and precision agriculture, optimizing pest control using drone technology. López et al. (2021) further advanced Bayesian adaptive designs for studying disease resistance, demonstrating flexibility in agricultural research.

Sweet potatoes' global significance is further underscored by their nutritional, economic, and environmental contributions. Sun et al. (2021) discussed the industrial use of sweet potato starch in biodegradable plastics, addressing sustainability challenges. Wang et al. (2022) combined real-time data and DOE to refine agricultural inputs dynamically. This integration of advanced techniques with traditional farming practices highlights the crop's pivotal role in fostering innovation and sustainability in agriculture.

2.11.3 Application of Response Surface Methodology in Agriculture

Response Surface Methodology (RSM) has become a vital tool in agriculture for optimizing crop production under various environmental conditions. Myers, Montgomery, and Anderson-Cook (2009) established the foundation of RSM for practical applications, particularly focusing on polynomial models to predict responses based on input factors. Kumar and Singh (2010) applied RSM in wheat production, using central composite designs to optimize nitrogen, phosphorus, and potassium (NPK) levels, which enhanced crop yield and nutrient uptake. Montgomery (2012) emphasized RSM's capability to determine optimal input levels to maximize yield, quality, or sustainability while minimizing costs. Mead et al. (2012) explored the effects of multiple fertilizers and irrigation strategies in rice farming, providing insights into optimal input combinations.

Patel et al. (2012) applied RSM in rice farming to model the response of plants to nitrogen and water levels, reducing nitrogen runoff while maximizing yield. Mehdi et al. (2013) optimized irrigation levels for wheat production in semi-arid regions, showing RSM's potential to conserve water while maintaining high yields. Borges et al. (2014) used RSM to model interactions between nitrogen and potassium in maize production, revealing non-linear nutrient interactions often missed in conventional studies. Hussain et al. (2014) applied RSM to study the interplay of compost and soil moisture on maize production, highlighting significant interactive effects on crop growth. Fang et al. (2015) optimized planting density and fertilizer levels for tomato production, achieving high yields without compromising fruit quality.

Gong et al. (2016) used RSM to optimize biochar and nitrogen application in rice farming, demonstrating sustainable practices that improve yield while enhancing nutrient cycling. Al-Omran et al. (2016) applied RSM to improve cucumber yield and quality through the careful management of organic fertilizers and water levels. Mandal et al. (2020) balanced wheat production with soil health and nutrient runoff, demonstrating RSM's role in promoting sustainable farming practices. Chen et al. (2020) combined RSM with artificial neural networks to optimize irrigation and fertilizer use in rice farming, enhancing model accuracy for complex, non-linear input interactions.

Sharma et al. (2021) explored irrigation strategies in rice farming, identifying optimal schedules for different growth stages. López et al. (2021) introduced Bayesian RSM to incorporate prior knowledge into experimental designs, reducing trial numbers while maintaining robust predictions. Singh et al. (2021) optimized crop rotation and pest management strategies, reducing chemical pesticide reliance while improving pest control. Zhang et al. (2021) used RSM to enhance drone-based pesticide spraying systems, optimizing efficiency and sustainability in pest control. Silva et al. (2018) applied RSM to optimize soil moisture and nutrient availability for vegetables, recommending irrigation schedules that enhanced water use efficiency.

Zhou et al. (2022) combined RSM with machine learning to optimize smart irrigation systems, leveraging real-time soil moisture data to improve water use efficiency. Wang et al. (2022) integrated decision trees with RSM to refine crop management practices, offering tailored recommendations based on field conditions. The continued evolution of RSM, especially when combined with emerging technologies, underscores its significance in advancing precision agriculture and sustainable farming practices.

2.11.4 Applications of SORD in Sweet Potato Yield Optimization

Sweet potatoes are a highly nutritious and versatile root vegetable with significant potential to address global food security challenges. Their adaptability and rich nutrient profile make them an essential crop for populations in need. Singh et al. (2004) utilized the Box-Behnken design to optimize sweet potato flour production, enhancing its quality by evaluating variables like drying temperature, time, and grinding conditions. Khuri and Mukhopadhyay (2010) emphasized the effectiveness of second-order rotatable designs (SORD) for agricultural optimization, particularly in estimating response variables with precision across complex systems like sweet potato farming. Goyal and Kachhwaha (2010) demonstrated the value of augmented SORDs (ASORDs) in improving irrigation schedules and crop yields. Goos and Jones (2011) noted that although SORD requires numerous trials, integration with modern technologies could reduce this limitation.

Adebola and Makinde (2012) applied SORD to optimize nitrogen and potassium levels in sweet potato farming, improving yields and promoting sustainability. Zhou et al. (2015) used SORD to optimize irrigation strategies in semi-arid China, showcasing its utility in precision agriculture. Kagoda et al. (2017) applied SORD in Uganda to model the effects of fertilizers, finding optimal nutrient levels to enhance yield and efficiency. Suri and Singh (2018) used SORD to optimize extrusion processes for sweet potato-based snacks, demonstrating improvements in product quality. Zhang et al. (2018) employed SORD to investigate genetic modifications for pest-resistant sweet potatoes, highlighting its role in screening multiple variables efficiently.

Adepoju et al. (2019) used response surface methodology and SORD to optimize drying conditions for sweet potatoes, while Guo et al. (2019) applied the method to enhance microwave-assisted drying efficiency. Lagos and Pastrana (2018) integrated SORD with precision agriculture to address challenges in resource-limited regions. Mgonja et al. (2020) optimized biochar application and planting density in Tanzania, promoting sustainable practices. Senevirathana et al. (2021) used central composite rotatable designs to optimize drying processes for sweet potatoes, further advancing post-harvest management.
Recent advancements include integrating SORD with real-time data and machine learning. Gao et al. (2021) combined SORD with sensors for dynamic input adjustments, boosting yields by 15%. Senevirathana et al. (2021) optimized drying processes using response surface methodology, while Chen et al. (2022) combined SORD with artificial neural networks to refine predictions, reduce experimental trials, and optimize fertilizer and irrigation. Li et al. (2022) applied SORD to improve sweet potato chip processing, achieving significant gains in sensory attributes. Kaptich (2022) optimized planting and spacing parameters for sweet potatoes using SORD, enhancing production efficiency. Bai (2022) demonstrated

the effectiveness of SORD in optimizing drying conditions, resulting in high-quality dried products. This body of research underscores the evolving role of SORD in addressing global food security through sweet potato farming.

2.11.5 Application of ASORDs for Modeling Sweet Potato Yield

Sweet potatoes *(Ipomoea batatas)* are a globally significant crop, contributing to food security through their high nutritional value and adaptability to diverse agro-ecological conditions. Despite this, optimizing sweet potato yields remains a complex challenge, influenced by soil fertility, irrigation, planting density, and climatic variability. To address these issues, response surface methodology (RSM) has been widely applied, with second-order rotatable designs (SORDs) and their augmented versions (ASORDs) emerging as powerful tools for modeling non-linear relationships between variables and crop yield. ASORDs, which enhance the flexibility and precision of experimental designs, allow researchers to explore complex interactions between variables such as fertilizer levels, irrigation, planting density, and crop variety.

The foundation of RSM and SORDs was established by Box and Hunter (1957), enabling the exploration of quadratic relationships with constant prediction variance. However, as agricultural systems became more complex, Khuri and Mukhopadhyay (2010) emphasized the need for augmented designs to enhance flexibility. ASORDs introduced additional design points, such as factorial or axial points, to improve the robustness of SORDs. These augmentations allowed for more comprehensive modeling of the response surface, particularly in agriculture, where multiple factors interact in unpredictable ways.

The application of ASORDs has provided significant insights into optimizing sweet potato yields. Adebola et al. (2016) applied ASORD to optimize nitrogen and potassium fertilizer levels in sub-Saharan Africa. By augmenting traditional SORDs with additional factorial points, the researchers identified optimal fertilizer combinations that maximized yield while reducing nutrient waste. Water management, another critical factor for sweet potato farming, has also been optimized using ASORD. Mehdi et al. (2017) applied ASORD to develop irrigation schedules in a semi-arid region of Iran, integrating axial points to model interactions between irrigation frequency, soil moisture, and yield.Similarly, Kagoda et al. (2018) used ASORD to explore nitrogen, phosphorus, and potassium interactions in Uganda, revealing precise NPK ratios for improved yield. These studies highlighted ASORD's ability to capture non-linear interactions often overlooked by traditional designs, enabling more effective and sustainable fertilizer recommendations.

The study demonstrated that ASORD enhanced water use efficiency while ensuring high yields. Similarly, Silva et al. (2019) used ASORD in Brazil to optimize drip irrigation, modeling interactions between irrigation, soil type, and fertilizer application. These studies underscored ASORD's value in resource-scarce environments, offering site-specific recommendations for efficient water management.

Planting density and crop spacing are additional factors influencing sweet potato yield, often interacting with soil fertility and water availability. Mgonja et al. (2020) used ASORD to examine the effects of planting density and biochar application in Tanzania, identifying combinations that improved yield and soil health. Similarly, Zhang et al. (2020) integrated ASORD with soil moisture sensors to optimize irrigation. Zhou et al. (2021) applied ASORD in China to optimize crop spacing, integrating axial points to model interactions between spacing, soil fertility, and water availability. Both studies demonstrated the importance of tailoring planting recommendations to local conditions using ASORD. While Lagos and Pastrana (2021) used drone-collected imagery to enhance fertilizer recommendations.These integrations represent a leap forward in precision agriculture, enabling real-time adjustments and improved resource efficiency.

Recent advancements in ASORD have focused on integrating the methodology with emerging technologies. Chen et al. (2022) combined ASORD with artificial neural networks (ANNs) to refine yield prediction models, leveraging real-time environmental data to optimize input levels dynamically.

Despite its benefits, ASORD presents challenges, including increased complexity and cost due to the need for additional experimental trials. Goos and Jones (2017) noted that the augmented design requires more resources, though machine learning can mitigate this by reducing the number of trials. Another limitation is ASORD's reliance on quadratic modeling, which may not capture more intricate relationships. Khuri (2016) suggested that while ASORD excels in capturing quadratic interactions, alternative designs or higher-order models may be necessary for more complex systems.

2.11.6 Effect of organic Manure on Sweet Potato Yield

The use of organic fertilizers, such as animal manure, plays a pivotal role in sustainable agriculture by enhancing soil fertility and improving crop yields. Organic amendments, including goat, rabbit, and poultry manure, are widely used to boost the productivity of crops like sweet potatoes (Ipomoea batatas). Organic manures provide essential nutrients like nitrogen (N), phosphorus (P), and potassium (K) while improving soil health, structure, and fertility. Unlike chemical fertilizers, they release nutrients slowly, ensuring long-term soil fertility and promoting microbial activity, better water retention, and increased organic matter (Vanlauwe et al., 2001). These properties make organic manures especially beneficial for root crops such as sweet potatoes, as they improve soil structure, aeration, and nutrient availability (Nzeyimana et al., 2014).

Goat manure, known for its high nitrogen, phosphorus, and potassium content, is a valuable organic fertilizer that breaks down slowly, offering sustained benefits to soil fertility. Wahome et al. (2015) found that goat manure significantly enhances soil fertility, leading to improved root formation in sweet potatoes. This slow nutrient release supports consistent nutrient availability throughout the growing season, which is crucial for root development. A study in Kenya by Ndambi et al. (2018) demonstrated that applying goat manure at a rate of 10 tons per hectare increased sweet potato tuber yields by up to 40% compared to untreated plots. The study attributed these results to the slow nutrient release and improved soil structure provided by goat manure. Ouda et al. (2016) emphasized that goat manure's gradual decomposition makes it particularly suitable for long-term soil improvement, enhancing tuber size and weight over time.

Rabbit manure is another effective organic fertilizer, noted for its high nitrogen content and the ability to be applied directly to crops without composting. With an average N-P-K ratio of 2.4-1.4-0.6, rabbit manure provides an immediate nutrient source, promoting early vegetative growth in crops like sweet potatoes (Acheampong et al., 2017). Kamanga et al. (2020) observed that rabbit manure enhanced vine growth, essential for photosynthesis and tuber development. Bello et al. (2019) reported that applying rabbit manure to sweet potato fields in Nigeria increased tuber yields by 30% compared to control plots. This improvement was attributed to the rapid release of nutrients during the early growing season, fostering faster plant establishment. Additionally, rabbit manure boosts soil microbial activity, facilitating the breakdown of organic matter and converting nutrients into forms readily absorbed by sweet potatoes (Olowoake et al., 2021).

Poultry manure is regarded as one of the most nutrient-rich organic fertilizers, with an N-P-K ratio as high as $4.5-2.5-1.0$(Salami et al., 2017). It provides a faster release of nutrients compared to goat and rabbit manure, making it ideal for crops with high nutrient demands, such as sweet potatoes. Poultry manure's high nitrogen content promotes rapid vine and leaf growth, enhancing photosynthesis and carbohydrate storage in tubers (Ano & Agwu, 2005). Its nutrient profile, which includes essential micronutrients like calcium and magnesium, further supports plant health and productivity. Studies have shown that the application of poultry manure significantly increases sweet potato yield, with rapid nutrient availability contributing to robust plant development and higher tuber yields.

Overall, goat, rabbit, and poultry manures provide unique benefits for sweet potato cultivation. While goat manure offers sustained nutrient release for long-term soil health, rabbit manure provides immediate nutrient availability, accelerating early growth. Poultry manure's high nutrient density supports rapid plant development, making it particularly suitable for intensive cultivation. Together, these organic fertilizers play a critical role in sustainable farming, enhancing soil fertility, improving crop yields, and supporting the long-term productivity of agricultural systems.

CHAPTER THREE

METHODOLOGY

This chapter outlines the methods that were used to construct second order rotatable designs in three dimensions using trigonometric functions. The application on Sweet Potatoes yield was also expounded.

3.1 A Method for Constructing Augmented SORD in Three Dimensions using Trigonometric Functions

3.1.1 Second Order Model

The process begins with a comprehensive understanding of second-order rotatable designs, which are central to the study of response surface methodology and experimental design. These designs enable the exploration of the relationships between several variables and their effects on a response variable, ensuring that predictions are equally precise in all directions from the center point. The initial step involves defining the mathematical foundation for second-order designs in three-dimensional space. This entails formulating the quadratic model, which can be expressed as:

$$y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum \sum_{i<j}^{k} \beta_{ij} x_i x_j + \varepsilon \tag{3.1}$$

where

β_o is the intercept

β_i is the linear coefficient for the i^{th} factor

β_{ii} is the quadratic coefficient for the i^{th} factors

$\beta_i j$ is the cross-product coefficient for the i^{th} and j^{th} factors

ε is the error term or residual.

We write the model in the form

$$y = \beta_0 + \beta_{11}x_{1u}^2 + \beta_{22}x_{2u}^2 + \beta_{33}x_{3u}^2 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{12}x_1x_2 + \beta_{13}x_1x_3 + \beta_{23}x_2x_3 \tag{3.2}$$

To augment these designs, trigonometric functions are incorporated. Trigonometric functions, known for their periodic properties, provide an effective means to introduce additional complexity and features into the designs.

By defining the predictor variables x_i and x_j in terms of trigonometric functions, such as $x_i = r.cos(\theta)$ and $x_j = r.sin(\theta)$ where r is the radius and θ is the angle. we can manipulate the spatial relationships and rotational symmetry of the design. The augmentation is achieved by adding trigonometric terms to the quadratic model, for example, incorporating terms like $cos(\omega\theta)$ and $sin(\omega\theta)$ to introduce periodic variations and enhance the design's properties.

3.1.2 Construction of 3s – Points

The construction of 3s-points in second-order rotatable designs (SORD) using trigonometric transformations was pioneered by Bose and Draper (1959). They introduced specific transformations that utilize trigonometric functions to generate points that maintain the rotatability property in three-dimensional space. These transformations are designed to distribute points uniformly on a sphere, which is essential for achieving equal precision in all directions from the center point.

The transformations introduced are characterized by two matrices, T_1 and T_2, which are mathematical sets that incorporate trigonometric functions to achieve the desired rotational effects. These matrices are defined as follows:

$$T_1 = \begin{bmatrix} cos\ \alpha & -sin\ \alpha & 0 \\ sin\ \alpha & cos\ \alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.3}$$

and

$$T_2 = \begin{bmatrix} cos\ \frac{\alpha}{2} & sin\ \frac{\alpha}{2} & 0 \\ sin\ \frac{\alpha}{2} & cos\ \frac{\alpha}{2} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{3.4}$$

where $\alpha = \frac{2\pi}{s}$

with s representing the number of points to be distributed on the sphere.

By employing these transformations, the methodology ensures that the constructed points maintain the desired rotatability and uniformity, which are critical for second-order rotatable designs in three dimensions. The incorporation of trigonometric functions facilitates the precise and systematic distribution of points, enhancing the effectiveness and accuracy of the experimental design.

In the present study, these transformations are utilized to create second-order rotatable designs. The transformations outlined in equations (3.3)and (3.4) are employed on the point sets structured as $G(r, 0, b)$, representing points on the plane where $y = 0$, and on all other points derived from successive applications of T_1 and T_2. The permutation group (I, W, W^2) generated by;

$$W = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \tag{3.5}$$

Consider a set $T(r, 0, b)$. Assuming $b = 0$, then $T(r, 0, b)$ becomes $T(r, 0, 0)$. Applying (3.3) and (3.5) respectively on $T(r, 0, 0)$ gives the following set of coordinates,

$$\begin{gathered} (rcos\ t\alpha,\ rsin\ t\alpha,\ 0) \\ (rsin\ t\alpha,\ 0,\ rcos\ t\alpha) \\ (0,\ rcos\ t\alpha,\ rsin\ t\alpha) \end{gathered} \tag{3.6}$$

For $t = 0, 1, 2 \ldots (s - 1)$ and $s \geq 5$, When $s \geq 5$ set $T_0(r, 0, 0)$ and points $(rcos\ t\alpha, rsin\ t\alpha, 0) = 3s$ The sums and products of the set up to power four for the co-ordinates listed in (3.6) were given by;

$$\begin{aligned}\sum_{u=1}^{N} x_{iu}^2 &= sr^2,\\ \sum_{u=1}^{N} x_{iu}^4 &= \frac{3}{4}sr^4,\\ \sum_{u=1}^{N} x_{iu}^2x_{ju}^2 &= \frac{1}{8}sr^4,\end{aligned} \tag{3.7}$$

The excess function for $T(r, o, o)$ was given by;

$$\sum_{u=1}^{N} x_{ui}^4 - 3\sum_{u=1}^{N} x_{iu}^2x_{ju}^2 = \frac{3}{8}sr^4 \tag{3.8}$$

The factor $\frac{3}{4}$ arises from integrating over the angle α, and The factor $\frac{1}{8}$ comes from the product of trigonometric functions.

This methodology uses sums and products of coordinates to analyze the statistical properties of a set of points generated by trigonometric transformations.

By calculating these properties up to the fourth power, we can assess the uniformity and distribution of the points, ensuring that they meet the criteria for constructing second-order rotatable designs.

The excess function further validates the balance and symmetry of the point set, which is critical for achieving accurate and reliable experimental designs.

3.1.3 Four Point Set

The methodology described involves constructing a set of four points in three-dimensional space, denoted as $G(a, a, a)$, and analyzing their properties.

This set is used for constructing augmented second-order rotatable designs, leveraging the symmetry and uniformity of the point distribution. The process considers a set of 8 points, which are halved to form the desired four-point set.

Similarly, the co-ordinates for the four points set denoted $G(a, a, a)$ were listed below, When number of points was 8 given by sets $\frac{1}{3}G(a,\ a,\ a)$ and points $(\pm a, \pm a, \pm a)$

but this runs are halved.

$$\begin{aligned} &G(a,a,a)\\ &G(-a,a,a)\\ &G(a,-a,a)\\ &G(a,a,-a) \end{aligned} \tag{3.9}$$

The sums and products up to power four for $G(a,\ a,\ a)$ was given by,

$$\begin{aligned} \sum_{u=1}^{N} x_{iu}^2 &= 4a^2, \\ \sum_{u=1}^{N} x_{iu}^4 &= 4a^4, \\ \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 &= 4a^4, \end{aligned} \tag{3.10}$$

The excess for the above set of points was given by

$$\sum_{u=1}^{N} x_{ui}^4 - 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = -8a^4 \tag{3.11}$$

3.1.4 The Six Point Set

The methodology described involves constructing a set of six points in three-dimensional space, denoted as $G(c,o,o)$, and analyzing their properties. This set is used for constructing augmented second-order rotatable designs by leveraging the symmetry and uniformity of the point distribution.

The points are carefully selected to maintain the rotatability property while simplifying the design process.

For the six points set denoted by $G(c,\ 0,\ 0)$, When the number of points was six given by set $\frac{1}{4}G(0,\ 0,\ c)$ and the points $(\pm c,\ 0,\ 0)$, $(0, \pm c,\ 0)$, $(0,\ 0, \pm, c)$ The co-ordinates for the six points were listed as below,

$$
\begin{aligned}
G(c, 0, 0)& \\
G(-c, 0, 0)& \\
G(0, -c, 0)& \\
G(0, c, 0)& \\
G(0, 0, c)& \\
G(0, 0, -c)&
\end{aligned} \tag{3.12}
$$

The sums and products up to power four for $G(c,\ 0,\ 0)$ was given by

$$
\begin{aligned}
\sum_{u=1}^{N} x_{iu}^2 &= 2c^2, \\
\sum_{u=1}^{N} x_{iu}^4 &= 2c^4, \\
\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 &= 0,
\end{aligned} \tag{3.13}
$$

This indicates that the sum of the squares of the coordinates over all points equals $2c^2$, where c is the distance from the origin along each axis also the sum of the fourth powers of the coordinates equals $2c^4$.

The sum of the products of squares of the coordinates across different dimensions equals zero, reflecting the orthogonality of the points.

The excess function is a measure used to evaluate the distribution properties of the point set. For the set $G(c,\ 0,\ 0)$, the excess function is given by:

$$
\sum_{u=1}^{N} x_{ui}^4 - 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = 2c^4 \tag{3.14}
$$

This function evaluates the difference between the sum of the fourth powers of the coordinates and three times the sum of the products of squares of the coordinates.The result, $2c^4$, is significant as it reflects the uniformity and balance of the points in the set.

3.1.5 Eight Point Set

The methodology described involves constructing a set of eight points in three-dimensional space, denoted as $G(a, a, a)$, and analyzing their properties. This set is used for constructing augmented second-order rotatable designs by leveraging the symmetry and uniformity of the point distribution.

The points are selected to maintain the rotatability property while ensuring the design process remains efficient and robust.

Similarly, the co-ordinates for the eight points set denoted $G(a, a, a)$ were listed below,

$$\begin{aligned} G(a,a,a) \\ G(-a,a,a) \\ G(a,-a,a) \\ G(a,a,-a) \\ G(-a,-a,a) \\ G(-a,a,-a) \\ G(a,-a,-a) \\ G(-a,-a,-a) \end{aligned} \tag{3.15}$$

These points represent all possible combinations of $\pm a$ in each of the three dimensions, forming the vertices of a cube centered at the origin.

The sums and products up to power four for $G(a, a, a)$ was given by,

$$\begin{aligned} \sum_{u=1}^{N} x_{iu}^2 &= 8a^2, \\ \sum_{u=1}^{N} x_{iu}^4 &= 8a^4, \\ \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 &= 8a^4, \end{aligned} \tag{3.16}$$

This indicates that the sum of the squares of the coordinates over all points equals $8a^2$ and a is the distance from the origin along each axis, also the sum of the fourth powers of the coordinates equals $8a^4$.

This indicates that the sum of the products of squares of the coordinates across different dimensions equals $8a^4$, reflecting the symmetry of the points.

The excess function is a measure used to evaluate the distribution properties of the point set. For the set $G(a, a, a)$, the excess for the above set of points is given by:

$$\sum_{u=1}^{N} x_{ui}^4 - 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = -16a^4 \tag{3.17}$$

This function evaluates the difference between the sum of the fourth powers of the coordinates and three times the sum of the products of squares of the coordinates. The result, $-16a^4$, is significant as it reflects the uniformity and balance of the points in the set.

3.1.6 Construction of 6s Point

The methodology described involves constructing a set of $6s$ points in three-dimensional space, denoted as $T(r, o, b)$, and analyzing their properties. This set is used for constructing augmented second-order rotatable designs by leveraging the symmetry and uniformity of the point distribution. The points are selected to maintain the rotatability property while ensuring the design process remains efficient and robust.

The coordinates for the $6s$ points set $T(r, o, b)$ are listed as follows:

$$\begin{aligned}
&G\left(rcos\ t\alpha,\ rsin\ t\alpha,\ b\right)\\
&G\left(rcos\ (t+\frac{1}{2})\alpha,\ rsin\ (t+\frac{1}{2})\alpha,\ -b\right)\\
&G\left(b,\ rcos\ t\alpha,\ rsin\ t\alpha\right)\\
&G\left(-b,\ rcos\ (t+\frac{1}{2})\alpha,\ rsin\ (t+\frac{1}{2})\alpha\right)\\
&G\left(rsin\ t\alpha,\ b,\ rcos\ t\alpha\right),\\
&G\left(rsin\ (t+\frac{1}{2})\alpha, -b,\ rcos\ (t+\frac{1}{2})\alpha\right)
\end{aligned} \tag{3.18}$$

where $\alpha = \frac{2\pi}{s}$ and $s \geq 4$ and and $t = 0.1.2...(s-l)$ The parameters r and b are chosen based on the desired distribution and rotatability properties. The set $T(r, 0, b)$ has sums of powers and products

$$\begin{aligned}
\sum_{u=1}^{N} x_{iu}^2 &= \sum_{u=1}^{N} x_{ju}^2 = \sum_{u=1}^{N} x_{lu}^2 = 2s(r^2+b^2)\\
\sum_{u=1}^{N} x_{iu}^4 &= \sum_{u=1}^{N} x_{ju}^4 = \sum_{u=1}^{N} x_{lu}^4 = \frac{s(3r^4+4b^4)}{2}\\
\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 &= \sum_{u=1}^{N} x_{ju}^2 x_{lu}^2 = \sum_{u=1}^{N} x_{lu}^2 x_{iu}^2 = \frac{sr^2(r^2+8b^2)}{4}
\end{aligned} \tag{3.19}$$

Additionally, all other powers and products up to and including the fourth order are zero.

The excess function is a measure used to evaluate the distribution properties of the point set, it follows that the excess of the set, defined similarly, was:

$$\sum_{u=1}^{N} x_{iu}^4 - 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = \frac{s(3r^4+4b^4)}{2} - \frac{sr^2(r^2+8b^2)}{4} = \frac{(3r^4-24r^2b^2+8b^4)s}{4} \tag{3.20}$$

This function evaluates the difference between the sum of the fourth powers of the coordinates and three times the sum of the products of squares of the coordinates.

3.1.7 Augmenting of Sets to give three dimension ASORDs

The methodology for this construction is based on combining sets of points derived from trigonometric functions and symmetrical distributions.

Augmenting Sets to Create ASORDs

1. Initial Sets:

The foundational sets of points are defined using trigonometric transformations. Specifically, the sets given by equations (3.6) and (3.15) form the basis of the construction.

The set from (3.6) provides $3s$ points, which are uniformly distributed in a circular pattern using trigonometric functions.

The set from (3.15) provides eight points distributed symmetrically in a three-dimensional space.

2.Combining Sets:

The goal is to combine these sets to achieve a three-dimensional second-order rotatable design.

This combination is expressed as $3s + 8$. However, for the design to be effective, s must be $s \geq 5$.

when $s = 5$, the combined set includes 15 points from the set in (3.6) and additional points from the sets in (3.15).

3.Augmentation Process:

The augmentation involves combining the $3s$ points from (3.6) with two classes of eight-point sets from (3.15). This is represented as:

$$Z_1 = 3s + \frac{1}{2}\left[\frac{1}{3}G(a_1, a_1, a_1)\right] + \frac{1}{2}\left[\frac{1}{3}G(a_2, a_2, a_2)\right] \tag{3.21}$$

Here,$G(a_1, a_1, a_1)$ and $G(a_2, a_2, a_2)$ represent two classes of eight-point sets that are symmetrically distributed.

4.Final Point Calculation:

By setting $s = 5$, the total number of points in the augmented set Z_1 is calculated. Thus, the final design consists of 23 points, ensuring that the design remains rotatable and uniform in three dimensions.

By augmenting specific sets of points in (3.6), (3.9) and (3.12), second order rotatable designs in three dimensions will be obtained. The augmentation is given by $3s + \dfrac{1}{2} of 8 + 6 = 3S + 4 + 6$. But $S \geq 5$ when $S = 5$, Combining $3s$ given in (3.6) with four-point set given in (3.9) and six-point set given in (3.12) to obtain set denoted Z_2 given as $3s + G(a, a, a) + G(c, 0, 0)$, by letting $s = 5$, 25 points will be obtained.

Consequently, by augmenting specific sets of points in (3.18) and (3.9), second order rotatable designs in three dimensions were obtained. The augmentation was given by $6S + \frac{1}{2} of 8 = 6S + 4$. But $S \geq 4$ when $S = 4$, Combining $6S$ given in (3.18) with four-point set given in (3.9) to obtain set denoted Z_3 given as $6s + \frac{1}{2}[\frac{1}{3}G(a_1, a_1, a_1]$, by letting $S = 4$, 28 points were obtained.

3.1.8 Moment and Non-Singularity Conditions

The construction of three-dimensional second-order rotatable designs (ASORDs) requires adherence to specific moment and non-singularity conditions to ensure the designs maintain their rotatable properties and statistical integrity.

These conditions involve satisfying certain mathematical criteria related to the moments and sums of powers of the point coordinates in the design. Here's a detailed explanation of the methodology:

To ensure a set of points forms a rotatable arrangement, specific moment conditions must be satisfied. These conditions relate to the sums of squares and products of the coordinates up to the fourth order.

$$\begin{aligned}
\sum_{u=1}^{N} x_{iu}^2 &= N\lambda_2, \\
\sum_{u=1}^{N} x_{iu}^4 &= 3N\lambda_4, \\
\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 &= N\lambda_4, \\
\sum_{u=1}^{N} x_{iu}^4 &= 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2
\end{aligned} \tag{3.22}$$

Here, λ_2 and λ_4 are constants that ensure the correct scaling and uniform distribution of the design points.

A group of points is deemed to constitute a second-order rotatable design when the aforementioned conditions are met, alongside the non-singularity of the matrix $X'X$ utilized in the least square estimation. Box and Hunter (1957) demonstrated that the necessary and adequate condition for this occurrence is,

$$\frac{\lambda_4}{\lambda_2^2} > \frac{k}{k+2} \tag{3.23}$$

This condition ensures that the ratio of the fourth moment to the square of the second moment is sufficiently large to prevent singularity.

3.2 Optimality Criteria for the ASORD Constructed using Trigonometric Functions

3.2.1 Design Matrix Construction

First, construct the design matrix X for the second-order rotatable design. The generalized design matrix X includes pure quadratic effects, linear effects, and interaction effects. The structure is as follows:

$$X = \begin{bmatrix} x_{01}\ x_{11}^2 \cdots x_{k1}^2 x_{11} \cdots x_{k1}\ x_{11}x_{21} \cdots x_{(k-1)1}x_{k1} \\ x_{02}\ x_{12}^2 \cdots x_{k2}^2 x_{12} \cdots x_{k2}\ x_{22}x_{22} \cdots x_{(k-2)2}x_{k2} \\ . \\ . \\ . \\ x_{0n}\ x_{1n}^2 \cdots x_{kn}^2 x_{1n} \cdots x_{kn}\ x_{1n}x_{2n} \cdots x_{(k-n)n}x_{kn} \end{bmatrix} \tag{3.24}$$

The design matrix in (3.24) was partitioned in the following sequence: the pure quadratic effects, the linear effects, and the interaction effects, respectively. This allows for a structured analysis of each type of effect within the design.

3.2.2 Moment Matrix Calculation

Calculate the moment matrix, M the moment matrix was represented as follows:

$$M = \frac{X^{'}X}{N} \tag{3.25}$$

X was;

$$N^{-1}(X^{'}X) = \begin{bmatrix} 1 & \lambda_2 & \lambda_2 & \lambda_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_2 & 3\lambda_4 & \lambda_4 & \lambda_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_2 & \lambda_4 & 3\lambda_4 & \lambda_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_2 & \lambda_4 & \lambda_4 & 3\lambda_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_4 \end{bmatrix} \tag{3.26}$$

where

$$X^{'} = \left[1,\ x_1^2,\ x_2^2,\ x_3^2,\ x_1,\ x_2,\ x_3,\ x_1x_2,\ x_1x_3,\ x_2x_3\right] \tag{3.27}$$

3.2.3 Coefficient Matrix Definition

Define the coefficient matrix, Where k represents the coefficient matrix for a complete parameter system and M denotes the moment matrix. The coefficient matrix

$K^{'}$ was:

$$K^{'}_{(}10 \times 10) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.28}$$

3.2.4 Information Matrix Calculation

A suitable method for assessing the effectiveness of a design when the parameter system of interest was taken into consideration. This was calculated using $K^{'}\beta$ is through the information matrix. Calculation of the information matrix C was sought to assess the effectiveness of the design as:

$$C_k(M) = \left[K^{'}_k M_k^{-1} K_k\right]^{-1} \tag{3.29}$$

where $M = \frac{X^{'}X}{N}$ and K represent the number of factors and X as defined in (3.24)

3.2.5 Evaluation of D-optimality criterion

The D-optimality criterion is a central concept in the theory of optimal designs, focusing on maximizing the determinant of the information matrix $(X^{'}X)$ or, equivalently, minimizing its inverse. This ensures the design's efficiency by maximizing the volume of the confidence ellipsoid for the estimated parameters.

1.Definition of the D-Criterion

The D-Criterion is defined as:

$$\phi_0(M) = \det(M)^{\frac{1}{s}}, \tag{3.30}$$

where M is the moment matrix, and s is the number of parameters.

2. Objective of the D-optimality Criterion

The objective of the D-optimality criterion is to maximize the determinant of the moment matrix derived from the design matrix X:

$$\phi_0(M) = \left| \frac{X'X}{N} \right|^{\frac{1}{s}} \tag{3.31}$$

3. Determinant of the Moment Matrix

The determinant of the moment matrix M can be decomposed into several components. Based on the generalized moment matrix structure, the determinant is given by:

$$|M| = \left(\prod_{i=1}^{3} |A_i| \right)^{\frac{1}{s}}. \tag{3.32}$$

4. Components of the Determinant

The components $|A_1|$, $|A_2|$, and $|A_3|$ are defined as follows:

$$\begin{aligned} |A_1| &= [3\lambda_4 - \lambda_2^2 + (k-1)(\lambda_4 - \lambda_2^2)](2\lambda_4)^{(k-1)}, && (3.33) \\ |A_2| &= \lambda_2^{[k]}, && (3.34) \\ |A_3| &= \lambda_4^{[k]}. && (3.35) \end{aligned}$$

5. Substituting Variables into the D-criterion

By substituting the variables from the above equations into the determinant expression, we obtain the final form of the D-criterion: Thus, by substituting the variables from (3.31) into (3.30), the D-criterion is expressed as:

$$\phi_0(M) = \left\{ [3\lambda_4 - \lambda_2^2 + (k-1)(\lambda_4 - \lambda_2^2)](2\lambda_4)^{(k-1)}[\lambda_2^{[k]}][\lambda_4^{[k]}] \right\}^{\frac{1}{s}}. \tag{3.36}$$

3.2.6 Evaluation of T-optimality criterion

The T-optimality criterion is designed to maximize the trace of the information matrix. A T-optimal design is used to differentiate between two or more models, where one model is true, aiming to minimize the variance of predictions according to the optimality criteria. The T-optimality criterion is particularly useful for model discrimination.

1.Definition of the T-Criterion:

The T-Criterion, also known as the trace criterion, is defined as:

$$\phi_1(M) = \frac{1}{s}\text{trace}(M). \tag{3.37}$$

Here, M represents the moment matrix and s is the number of factors or variables.

2. Summation of Traces:

The T-Criterion is derived by summing the traces of the sub-matrices within the moment matrix. The trace of the moment matrix M is given by:

$$\text{tr}(M) = \frac{1}{s}[1 + \text{tr}(A_1) + \text{tr}(A_2) + \text{tr}(A_3)]. \tag{3.38}$$

3.Trace of Sub-Matrices:

The traces of the sub-matrices are calculated as follows: Where:

$$\text{tr}(A_1) = k(3\lambda_4), \tag{3.39}$$

$$\text{tr}(A_2) = k\lambda_2, \tag{3.40}$$

$$\text{tr}(A_3) = k\lambda_3. \tag{3.41}$$

Here, λ_2, λ_3 and λ_4 are constants derived from the design parameters, and k is the number of factors.

4. Substitution into the T-Criterion:

By substituting the values from the trace calculations into the T-Criterion, we get:

$$\phi_1(M) = \frac{1}{s}[1 + k(3\lambda_4) + k\lambda_2 + k\lambda_3]. \tag{3.42}$$

3.2.7 Evaluation of A-optimality criterion

The A-optimality criterion aims to minimize the trace of the inverse of the information matrix. This criterion is focused on minimizing the total or average variances of the parameter estimates.

1.Definition of the A-Criterion:

The A-Criterion, also known as the average variance criterion, is defined as:

$$\phi_{-1}(M) = (\frac{1}{s}\text{trace}M^{-1})^{-1} \tag{3.43}$$

2. Summation of Inverse Traces:

The A-Criterion is derived by summing the traces of the inverses of the sub-matrices within the moment matrix. The trace of the inverse moment matrix (M^{-1}) is given by:

$$\text{tr}(M^{-1}) = \text{tr}(A_1^{-1}) + \text{tr}(A_2^{-1}) + \text{tr}(A_3^{-1}). \tag{3.44}$$

Here, A_1^{-1}, A_2^{-1}, and A_3^{-1} are the inverses of the sub-matrices of M.

3.Trace of Inverse Sub-Matrices:

The traces of the inverse sub-matrices are calculated as follows: Where;

$$\text{tr}(A_1^{-1}) = \frac{1}{2[(k+2)\lambda_4^2 - k\lambda_2^2\lambda_4]}\left[2(k+2)\lambda_4^2 + k[(k+1)\lambda_4 - (k-1)\lambda_2^2]\right], \tag{3.45}$$

$$\text{tr}(A_2^{-1}) = \frac{k}{\lambda_2}, \tag{3.46}$$

$$\text{tr}(A_3^{-1}) = \frac{k}{\lambda_4}. \tag{3.47}$$

Here,λ_2 and λ_4 are constants derived from the design parameters, and k is the number of factors.

4.Substitution into the A-Criterion:

By substituting the values from the trace calculations into the A-Criterion, we get:

$$\begin{aligned}\phi_{-1}(M) = \Bigg[\frac{1}{s}\Bigg\{&\frac{1}{2[(k+2)\lambda_4^2 - k\lambda_2^2\lambda_4]}\left[2(k+2)\lambda_4^2\right.\\ &\left.+k[(k+1)\lambda_4 - (k-1)\lambda_2^2]\right] + \frac{k}{\lambda_2} + \frac{k}{\lambda_4}\Bigg\}\Bigg]^{-1}.\end{aligned} \tag{3.48}$$

3.2.8 Evaluation of E-optimality criterion

The E-optimality criterion is another design criterion that aims to minimize the largest eigenvalue of the dispersion matrix. This criterion focuses on reducing the variance associated with each individual parameter estimate. The evaluation of the E-optimality criterion is carried out using the following formula:

1. Mathematical Representation:

The E-optimality criterion is mathematically represented as:

$$\phi_{-\infty}(M) = \lambda_{\min}[(M)]. \tag{3.49}$$

Here, $\lambda_{\min}(M)$ denotes the smallest eigenvalue of the moment matrix M.

2. Moment Matrix Determinant:

The determinant of the dispersion matrix Σ is derived from the moment matrix M as follows:

$$|\Sigma| = |\Lambda||\Lambda^T|. \tag{3.50}$$

3. Sub-Matrix Determinants:

The determinants of the sub-matrices of M with shift operator are given by:

$$|A_1^*| = (1-\gamma)^2[(k+2)\lambda_4 - k\lambda_2^2 - \gamma](2\lambda_4 - \gamma)^{(k-1)}, \tag{3.51}$$

$$|A_2^*| = (\lambda_2 - \gamma)^k, \quad \text{and} \quad |A_3^*| = (\lambda_4 - \gamma)^k. \tag{3.52}$$

4.Characteristic Polynomial:

By substituting the expressions for the sub-matrix determinants into the determinant equation and setting it to zero, we obtain a characteristic polynomial:

$$\phi_{-\infty}(M) = (1-\gamma)^2[(k+2)\lambda_4 - k\lambda_2^2 - \gamma](2\lambda_4 - \gamma)^{(k-1)}(\lambda_2 - \gamma)^k(\lambda_4 - \gamma)^k = 0. \tag{3.53}$$

5. Solving the Polynomial:

Solving the characteristic polynomial for λ yields the smallest eigenvalue, which is used to determine the E-optimality criterion.

3.3 Examining Relative Efficiency of Designs

These are metrics that compare the performance of different experimental designs under various optimality criteria. The help identify how well a design performs compared to a reference design. In this study, we examined the relative efficiency of different design criteria—specifically $D-$, $A-$, $T-$ and $E-$ optimal designs. These designs were evaluated for configurations with 23, 25, and 28 points respectively. The relative efficiency was assessed for each design based on how well they met the optimality criteria.

The relative efficiency of each design was calculated by comparing the ratios of the optimality criteria values. A design was considered more efficient if it achieved a higher percentage of the optimality criteria ratio. This indicates better fulfillment of the optimality requirements.
The ratio quantifies how much information the tested design provides relative to the reference design.

3.3.1 Relative D-efficiency

Relative D-efficiency is a key concept in the evaluation of design optimality, providing a measure of how efficient a given design is compared to the best possible (D-optimal) design. According to Burgess (2004), the relative D-efficiency of a design is defined as the absolute value of the ratio of the D-criterion value for a specific design to that of the D-optimal design. This method helps in quantifying the efficiency of various designs in a comparative manner.
The relative D-efficiency is given by:

$$D - Efficiency = \left| \frac{M(\varepsilon^*)}{M(\varepsilon)} \right| \tag{3.54}$$

where:

- $M(\varepsilon^*)$ represents the D-criterion value for the D-optimal design.
- $M(\varepsilon)$ represents the D-criterion value for the given design.

Steps to Evaluate Relative D-Efficiency

1. **Identify the D-criterion Values:**
   - Calculate the D-criterion value $M(\varepsilon)$ for the specific design under consideration.
   - Obtain the D-criterion value $M(\varepsilon^*)$ for the D-optimal design.
2. **Compute the Relative D-Efficiency:**
   - Use the formula for relative D-efficiency to compute the ratio of the D-criterion values.
   - Take the absolute value of this ratio to ensure a non-negative efficiency measure.
3. **Interpret the Efficiency:**
   - A relative D-efficiency value close to 1 indicates that the specific design is nearly as efficient as the D-optimal design.
   - A value closer to 1 indicates that the tested design is nearly as efficient as the optimal design.

3.3.2 Relative T-efficiency

The concept of relative T-efficiency for any design, introduced by Cooke (1979), provides a measure to evaluate how efficient a given design is compared to the optimal T-design. This method is particularly useful for assessing the effectiveness of designs in terms of maximizing the trace of the information matrix.

$$T - efficiency = \frac{tr(M(\xi^*))}{tr(M(\xi))} \tag{3.55}$$

where:

- ξ^* represents the value of the optimal T-design.
- ξ denotes the value of a specific T-design.

Steps to Evaluate Relative T-Efficiency

1. **Identify the Information Matrices:**
   - Obtain the information matrix $M(\xi^*)$ for the optimal T-design.
   - Calculate the information matrix $M(\xi)$ for the specific T-design under consideration.
2. **Compute the Traces:**
   - Calculate the trace of the information matrix for the specific T-design, $\mathrm{tr}(M(\xi))$.
   - Calculate the trace of the information matrix for the optimal T-design, $\mathrm{tr}(M(\xi^*))$.
3. **Compute the Relative T-Efficiency:**
   - Use the formula for relative T-efficiency to compute the ratio of the traces of the information matrices.
4. **Interpret the Efficiency:**
   - A relative T-efficiency value close to 1 indicates that the specific design is nearly as efficient as the T-optimal design.
   - A value closer to 1 indicates that the tested design is nearly as efficient as the optimal design.

3.3.3 Relative A-efficiency

The relative A-efficiency of a design ξ is a measure used to evaluate how efficiently a design minimizes the total or average variance of the parameter estimates compared to the A-optimal design. This method is particularly useful for assessing the effectiveness of designs in terms of minimizing the trace of the inverse of the information matrix.

Relative A-efficiency is defined as:

$$A\text{-efficiency} = \frac{\mathrm{tr}(M^{-1}(\xi^*))}{\mathrm{tr}(M^{-1}(\xi))} \tag{3.56}$$

where:

- ξ^* represents the value of the optimal A-design.
- ξ denotes the value of a specific A-design.

Steps to Evaluate Relative A-Efficiency

1. **Identify the Information Matrices:**
   - Calculate the information matrix $M(\xi)$ for the specific A-design under consideration.
   - Obtain the information matrix $M(\xi^*)$ for the optimal A-design.
2. **Compute the Inverse Information Matrices:**
   - Calculate the inverse of the information matrix for the specific A-design, $M^{-1}(\xi)$.
   - Calculate the inverse of the information matrix for the optimal A-design, $M^{-1}(\xi^*)$.
3. **Compute the Traces:**

- Calculate the trace of the inverse information matrix for the specific A-design, $\mathrm{tr}(M^{-1}(\xi))$.
- Calculate the trace of the inverse information matrix for the optimal A-design, $\mathrm{tr}(M^{-1}(\xi^*))$.

4. **Compute the Relative A-Efficiency:**
   - Use the formula for relative A-efficiency to compute the ratio of the traces of the inverse information matrices.
5. **Interpret the Efficiency:**
   - A relative A-efficiency value close to 1 indicates that the specific design is nearly as efficient as the A-optimal design.
   - A value closer to 1 indicates that the tested design is nearly as efficient as the optimal design.

3.3.4 Relative E-efficiency

Relative E-efficiency is a measure used to evaluate how efficiently a design minimizes the largest eigenvalue of the information matrix compared to the E-optimal design. This criterion is particularly useful for assessing the effectiveness of designs in reducing the maximum variance among the parameter estimates.

Relative E-efficiency is defined as:

$$E\text{-efficiency} = \frac{\lambda_{\min}(\xi^*)}{\lambda_{\min}(\xi)} \tag{3.57}$$

where:

- ξ represents the specific design under consideration.

- ξ^* denotes the E-optimal design.
- $\lambda_{\min}(\xi^*)$ is the smallest eigenvalue of the information matrix for the E-optimal design ξ^*.
- $\lambda_{\min}(\xi)$ is the smallest eigenvalue of the information matrix for the design ξ.

Steps to Evaluate Relative E-Efficiency

1. **Identify the Information Matrices:**
   - Obtain the information matrix $M(\xi^*)$ for the E-optimal design.
   - Calculate the information matrix $M(\xi)$ for the specific design under consideration.
2. **Compute the Smallest Eigenvalues:**
   - Determine the smallest eigenvalue of the information matrix for the E-optimal design, $\lambda_{\min}(\xi^*)$.
   - Determine the smallest eigenvalue of the information matrix for the specific design, $\lambda_{\min}(\xi)$.
3. **Compute the Relative E-Efficiency:**
   - Use the formula for relative E-efficiency to compute the ratio of the smallest eigenvalues of the information matrices.
4. **Interpret the Efficiency:**
   - A relative E-efficiency value close to 1 indicates that the specific design is nearly as efficient as the E-optimal design.
   - A value closer to 1 indicates that the tested design is nearly as efficient as the optimal design.

3.4 Modelling the Yield of Sweet Potatoes Using ASORD

To optimize the yield of sweet potatoes using organic manure, an Augmented Second-Order Rotatable Design (ASORD) was employed. ASORD allows for efficient estimation of interaction and quadratic effects, enabling the evaluation of response surface curvature using a manageable number of experimental runs.

A total of 23 runs were conducted, providing sufficient information for model development while minimizing experimental costs: Poultry manure (x_1), rabbit manure (x_2) and goat manure(x_3) were the independent variables to optimize the response of interest (the weight yield of sweet potatoes).

The independent variables were normalized using the coding formula:

$$x_i = \frac{X_i - X_0}{\Delta X}$$

where x_i is the coded value, X_i is the natural value, X_0 is the central point, and ΔX is the step change.

This coding ensures that variables are dimensionless and scaled for consistency across the experimental design.

3.4.1 Experimental SETUP

- **Plot Dimensions:** Each of the 23 experimental plots measured 350 cm x 350 cm, arranged on a 30 m x 30 m research field.

- **Manure Application:** The organic manures were incorporated into the soil two weeks before planting.

- **Baseline Data:** Nutrient profiles for poultry, rabbit, and goat manure were obtained from the Kenya Agricultural and Livestock Research Organization (KALRO) database.

3.4.2 Model Specification

Develop a second-order polynomial model to approximate the relationship between the yield and the independent variables:

$$Y_i = f(x_1, x_2, x_3) + \varepsilon \tag{3.58}$$

where Y_i is the yield of sweet potatoes, x_1, x_2, x_3 are the amounts of poultry, rabbit, and goat manures, respectively, and ε is the error term. If the response variable is defined by linearly independent variables, then the response is first order and we use the least squares method. In this study, a second-order model was used because of the large size and the potential presence of curvature in agricultural setups. For our case with three independent variables, the model includes the main effects, interaction of the main effects, and the quadratic effects.

3.4.3 Estimation of Model Parameters

Using the coded variables, a regression model was developed to capture the relationship between the independent variables (manure types) and the response (sweet potato yield).

Use the collected data to estimate the model parameters (β-coefficients) using regression analysis. The model can be expressed in matrix form as:

$$y = X\beta + \varepsilon \tag{3.59}$$

Analysis of the main effects of each type of manure, Interaction effects (how the combination of manures influences yield) and Quadratic effects (response curvature due to increased manure rates). where y is the vector of response variables, X represents the explanatory variables, β are the coefficients or the vector of parameters, and ε consists of the error term. The coefficients are determined by least square methods as:

$$\hat{\beta} = (X^T X)^{-1} X^T y \tag{3.60}$$

3.4.4 Data Validation

Recording of the yield of sweet potatoes obtained from each treatment was taken with respective baseline of the manure rates applied from KALRO. Data validation was carried out to ensure that data was collected accurately and consistently captured. The three organic fertilizers were considered because: Poultry manure is rich in nitrogen $1-4\%$, phosphorus $1-3\%$, potassium $0.5-2\%$, and organic matter $40-60\%$ making it a valuable source of nutrients for plant growth. Rabbit manure is also high in nutrients such as nitrogen $2-3\%$, phosphorus $1-3\%$, pottasium $0.5-2\%$ and organic matter $40-60\%$ but may have different release dynamics compared to poultry manure. Goat manure is known for its high nitrogen content $1-3\%$, phosphorus $0.5-1.5\%$, potassium $1-2\%$ and organic matter $30-50\%$ and can contribute to improved soil structure.

3.4.5 Data Analysis

The analysis involved computing the response variable, which in this instance was the yield of sweet potatoes, for each treatment. Python software was utilized to analyze the data and determine any notable impacts of the organic fertilizers on the yield. where Analysis of Variance (ANOVA) was carried out to assess the significance of individual factors and interactions.

Model Diagnostics was also done to check residual plots and goodness-of-fit measures to ensure the model's adequacy. Lastly, graphical analysis was generated of 2D contour plots and 3D response surfaces to visualize factor effects and optimize the yield.

3.4.6 Model validation

The study developed a model to explain the relationship between the organic fertilisers (poultry, rabbit, and goat) and the yield of sweet potatoes. The study also employed a regression analysis to created a model. The model accounted for both the primary effects of the fertilizers and any possible interactions among them. Its validity was confirmed through an assessment of goodness-of-fit metrics, including R-squared, adjusted R-squared, and residual analysis. This step ensured that the model adequately represents the relationship between the organic fertilisers and the yield of sweet potatoes.

3.4.7 Optimization

The study used the developed model to predict the optimum combination of organic fertilisers that gave the highest sweet potato production. By considering the model's coefficients and any constraints or limitations, these helped to determine the combination of poultry, goat, and rabbit fertilisers that were likely result in the maximum yield.

CHAPTER FOUR

RESULTS AND DISCUSSIONS

This chapter focused on creating second-order rotatable designs in three dimensions using trigonometric functions. The outcomes for designs comprising 23, 25, and 28 points were presented. Subsequently, the D-criterion, A-criterion, E-criterion, and T-criterion alphabetic optimality were assessed based on the moments derived from these designs. The relative efficiencies of the alphabetic optimality criteria were identified and analyzed. The desirable design was used to model the yield of sweet potatoes.

4.1 Construction of second order rotatable designs in three dimensions utilizing trigonometric functions

4.1.1 Construction of 23 set points

The moment conditions necessary for a set of twenty-three points to constitute a augmented second-order rotatable arrangement are derived by augmenting the $3s$ set and two half of 8 set points obtained in equations (3.6) and (3.15), respectively. The study started by confirming the rotatability of the design points given in equation (3.6) using the moment conditions from equation (3.7):

$$\sum_{u=1}^{N} x_{iu}^2 = sr^2 = N\lambda_2 \tag{4.1}$$

$$\sum_{u=1}^{N} x_{iu}^4 = \frac{3}{4}sr^4 = 3N\lambda_4 \tag{4.2}$$

$$\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = \frac{1}{8} sr^4 = N\lambda_4 \tag{4.3}$$

$$\sum_{u=1}^{N} x_{iu}^4 = 3 \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 \tag{4.4}$$

To check the calculations, we solve for the excess in equation (4.4): This is achieved by comparing higher-order moments $\sum x_{iu}^4$ and $3\sum x_{iu}^2 x_{ju}^2$, the difference $\frac{3}{8}s_r^4$ confirms that the points adhere to rotatability criteria.

$$\sum_{u=1}^{N} x_{iu}^4 - 3 \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = \frac{3}{4} sr^4 - \frac{3}{8} sr^4 = \frac{3}{8} sr^4 \tag{4.5}$$

Next, we confirm the rotatability of the design points given in equation (3.9) using the moment conditions from equation (3.10) for a_1:

$$\sum_{u=1}^{N} x_{iu}^2 = 4a_1^2 \tag{4.6}$$

$$\sum_{u=1}^{N} x_{iu}^4 = 4a_1^4 \tag{4.7}$$

$$\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = 4a_1^4 \tag{4.8}$$

$$\sum_{u=1}^{N} x_{iu}^4 = 3 \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 \tag{4.9}$$

Solving for excess in (4.9) gave;

$$\sum_{u=1}^{N} x_{iu}^4 - 3 \sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = 4a_1^4 - 12a_1^4 = -8a_1^4 \tag{4.10}$$

Again, we confirm the rotatability for the design points given in equation (3.9) with the moment conditions given in (3.10) for a_2:

$$\sum_{u=1}^{N} x_{iu}^2 = 4a_2^2 \tag{4.11}$$

$$\sum_{u=1}^{N} x_{iu}^4 = 4a_2^4 \tag{4.12}$$

$$\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = 4a_2^4 \tag{4.13}$$

$$\sum_{u=1}^{N} x_{iu}^4 = 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 \tag{4.14}$$

Solving for excess in (4.14) gave;

$$\sum_{u=1}^{N} x_{iu}^4 - 3\sum_{u=1}^{N} x_{iu}^2 x_{ju}^2 = 4a_2^4 - 12a_2^4 = -8a_2^4 \tag{4.15}$$

Solving equation (4.5), (4.10) and (4.15) By setting $s = 5$, the equation is solved for a_1^2 and a_2^2 as proportional to xr^2 and yr^2, respectively;

$$\frac{15}{8}r^4 - 8a_1^4 - 8a_2^4 = 0 \tag{4.16}$$

Let

$$a_1^2 = xr^2 \tag{4.17}$$

$$a_2^2 = yr^2 \tag{4.18}$$

Substituting (4.17) and (4.18) into (4.16) gave;

$$(x^2 + y^2) = \frac{15}{64} \tag{4.19}$$

Where x is arbitrary and has a positive value say 0.05 using Python software to solve equation (4.19) to obtain.

$$y = 0.482 \quad \text{and} \quad x = 0.05 \tag{4.20}$$

These finally gave;

$$\sum_{u=1}^{23} x_{iu}^2 = sr^2 + 4a_1^2 + 4a_2^2 = 23\lambda_2 \tag{4.21}$$

$$\sum_{u=1}^{23} x_{iu}^4 = \frac{3}{4}sr^4 + 4a_1^4 + 4a_2^4 = 69\lambda_4 \tag{4.22}$$

$$\sum_{u=1}^{23} x_{iu}^2 = 5r^2 + 0.2r^2 + 1.928r^2 = 23\lambda_2 \tag{4.23}$$

$$\lambda_2 = 0.30991r^2 \tag{4.24}$$

$$\sum_{u=1}^{23} x_{iu}^4 = 3.75r^4 + 0.01r^2 + 0.93r^4 = 69\lambda_4 \tag{4.25}$$

$$\lambda_4 = 0.06797r^4 \tag{4.26}$$

The non-singularity condition for the second order is

$$\frac{\lambda_4}{\lambda_2^2} = \frac{0.06797}{(0.30991)^2} = 0.70768 > \frac{k}{k+2} = \frac{3}{5} = 0.6 \tag{4.27}$$

therefore

$$\frac{\lambda_4}{\lambda_2^2} > \frac{k}{k+2} = 0.70768 > 0.6 \tag{4.28}$$

which satisfies the non-singularity condition for second-order rotatability.

The study successfully constructs a three-dimensional second-order rotatable design by meeting the necessary moment conditions and verifying non-singularity. The derived expressions for the second- and fourth-order moments (λ_2 and λ_4) align with theoretical requirements, ensuring the design is both rotatable and balanced. Furthermore, the introduction of parameters (x and y) and the subsequent solution for their values demonstrate the design's flexibility in point arrangement while adhering to defined constraints. Importantly, the design surpasses the non-singularity threshold, confirming its robustness and suitability for practical applications, such as response surface methodology (RSM), where isotropy and uniform variance are essential.

4.1.2 Construction of 25 set points

The moment conditions required for the set of twenty-five points to constitute a second-order rotatable arrangement are obtained by the augmentation of 3s set with 4 set and 6 set points. The moment conditions given in (3.7),(3.10) and (3.13) were used on the design point given in (3.6),(3.9) and (3.12) to confirm rotatability. Thus, these conditions gave;

$$\sum_{u=1}^{25} x_{iu}^2 = sr^2 + 4a^2 + 2c^2 = N\lambda_2 \tag{4.29}$$

$$\sum_{u=1}^{25} x_{iu}^4 = \frac{3}{4}sr^4 + 4a^4 + 2c^4 = 3N\lambda_4 \tag{4.30}$$

$$\sum_{u=1}^{25} x_{iu}^2 x_{ju}^2 = \frac{1}{8}sr^4 + 4a^4 = N\lambda_4 \tag{4.31}$$

Solving for the excess gave;

$$\frac{3}{4}sr^4 + 4a^4 + 2c^4 - 3\left(\frac{1}{8}sr^4 + 4a^4\right) = 0 \tag{4.32}$$

Solving equation (4.32) for excess functions when $s = 5$ gave;

$$\frac{15}{4}r^4 + 4a^4 + 2c^4 - \frac{15}{8}r^4 - 12a^4 = 0 \tag{4.33}$$

Let

$$r^2 = xa^2 \tag{4.34}$$

$$c^2 = ya^2 \tag{4.35}$$

Substituting (4.34) and (4.35) into (4.33) gave;

$$15x^2 + 16y^2 = 64 \tag{4.36}$$

Where x is arbitrary and has a positive value, say 1. Using Python software to solve equation (4.36) to obtain:

$$y = 1.75 \quad \text{and} \quad x = 1 \tag{4.37}$$

These finally gave;

$$\sum_{u=1}^{25} x_{iu}^2 = sr^2 + 4a^2 + 2c^2 = 25\lambda_2 \tag{4.38}$$

$$\lambda_2 = 0.500a^2 \tag{4.39}$$

$$\sum_{u=1}^{25} x_{iu}^4 = \frac{3}{4}sr^4 + 4a^4 + 2c^4 = 75\lambda_4 \tag{4.40}$$

$$\lambda_4 = 0.185a^4 \tag{4.41}$$

The non-singularity condition for second order is

$$\frac{\lambda_4}{\lambda_2^2} = \frac{0.185}{(0.500)^2} = 0.74 > \frac{k}{k+2} = \frac{3}{5} = 0.6 \tag{4.42}$$

which satisfies the non-singularity condition for second-order rotatability.

This study constructs a second-order rotatable design with 25 points by augmenting the 3^s set with 4 and 6 additional points. The derived moment conditions ensure the design adheres to rotatability principles. By introducing parameters (x and y) and solving for their values, the study achieves a flexible and symmetric point arrangement while maintaining theoretical consistency.

The computed constants ($\lambda_2 = 0.500a^2$, $\lambda_4 = 0.185a^4$) confirm the balance of second- and fourth-order moments. Moreover, the design satisfies the non-singularity condition, with $\frac{\lambda_4}{\lambda_2^2} = 0.74$, which exceeds the critical threshold (0.6). This ensures robustness and isotropy, making the design suitable for practical applications such as response surface methodology, where uniformity across directions is vital. The method exemplifies the use of computational tools and theoretical principles to achieve precise experimental designs, enhancing their applicability in various scientific domains.

4.1.3 Construction of 28 set points

The criteria that need to be met by the set of twenty-eight points to constitute a second-order rotatable arrangement are determined by enhancing the 6s set with 4 set points derived from equations (3.18) and (3.9) accordingly. The moment conditions given in (3.19) were used on the design point given in (3.18) to confirm rotatability. Thus, these conditions gave;

$$\sum_{U=1}^{N} X_i u^2 = 2s(r^2 + b^2) = N\lambda_2 \tag{4.43}$$

$$\sum_{u=1}^{N} x_i u^4 = \frac{s(3r^4 + 4b^4)}{2} = 3N\lambda_4 \tag{4.44}$$

$$\sum_{u=1}^{N} x_i u^2 x_j u^2 = \frac{sr^2(r^2 + 8b^2)}{4} = N\lambda_4 \tag{4.45}$$

$$\sum_{u=1}^{N} x_i u^4 = 3\sum_{u=1}^{N} x_i u^2 x_j u^2 \tag{4.46}$$

Solving for the excess is given in (4.46) gave;

$$\sum_{u=1}^{N} x_i u^4 - 3\sum_{u=1}^{N} x_i u^2 x_j u^2 = \frac{s(3r^4 + 4b^4)}{2} - 3\left(\frac{sr^2(r^2 + 8b^2)}{4}\right) = 0 \tag{4.47}$$

The moment conditions given in (3.10) were used on the design point given in (3.9) to confirm rotatability. Thus, these conditions gave;

$$\sum_{u=1}^{N} x_i u^2 = 4a_1^2 \tag{4.48}$$

$$\sum_{u=1}^{N} x_i u^4 = 4a_1^4 \tag{4.49}$$

$$\sum_{u=1}^{N} x_i u^2 x_j u^2 = 4a_1^4 \tag{4.50}$$

$$\sum_{u=1}^{N} x_i u^4 = 3\sum_{u=1}^{N} x_i u^2 x_j u^2 \tag{4.51}$$

Solving for excess in (4.51) gave;

$$\sum_{u=1}^{N} x_i u^4 - 3\sum_{u=1}^{N} x_i u^2 x_j u^2 = 4a_1^4 - 12a_1^4 = -8a_1^4 \quad (4.32) \tag{4.52}$$

Solving equation(4.52) for excess functions when $s = 4$ gave;

$$12r^4 + 32b^4 - 96r^2b^2 = 32a^4 \tag{4.53}$$

Let

$$r^2 = xa^2 \tag{4.54}$$

$$b^2 = ya^2 \tag{4.55}$$

Substituting (4.54) and (4.55) into (4.53) gave;

$$12x^2 + 32y^2 - 96xy = 32 \tag{4.56}$$

Where x is arbitrary and has a positive value say 1, equation (4.56) gave;

$$32y^2 - 96y - 20 = 0 \tag{4.57}$$

Solving the quadratic equation using Python software to equation (4.57) to obtain:

$$y = 3.2 \quad \text{and} \quad x = 1 \tag{4.58}$$

These finally gave;

$$\sum_{U=1}^{28} X_i u^2 = 4a^2 + 2s(r^2 + b^2) = 28\lambda_2 \tag{4.59}$$

$$\sum_{U=1}^{28} X_i u^2 = 4a^2 + 8(a^2 + 3.2a^2) = 28\lambda_2 \tag{4.60}$$

$$\sum_{U=1}^{28} X_i u^2 = 4a^2 + 8a^2 + 25.6a^2 = 28\lambda_2 \tag{4.61}$$

$$\lambda_2 = 1.3429a^2 \tag{4.62}$$

Similarly;

$$\sum_{u=1}^{28} x_i u^4 = 4a^4 + \frac{s(3r^4 + 4b^4)}{2} = 84\lambda_4 \tag{4.63}$$

$$\sum_{u=1}^{28} x_i u^4 = 4a^4 + 6r^4 + 8b^4 = 84\lambda_4 \tag{4.64}$$

$$\sum_{u=1}^{28} x_i u^4 = 4a^4 + 6a^4 + 81.92a^4 = 84\lambda_4 \tag{4.65}$$

$$\lambda_4 = 1.094286a^4 \tag{4.66}$$

The non-singularity condition for second order is

$$\frac{\lambda_4}{\lambda_2^2} = \frac{1.094286}{(1.3429)^2} = 0.60679 > \frac{k}{k+2} = \frac{3}{5} = 0.6 \tag{4.67}$$

which satisfies the non-singularity condition for second-order rotatability. The construction of the 28-point design demonstrates the flexibility and precision achievable through the augmentation of an existing 6^s set with additional points. By satisfying the moment conditions, the design ensures second-order rotatability and isotropy, critical for response surface methodology and other experimental applications.

The derived constants ($\lambda_2 = 1.3429a^2$, $\lambda_4 = 1.094286a^4$) confirm balance and robustness in the design. Furthermore, the non-singularity check ($0.60679 > 0.6$) guarantees stability, making the arrangement well-suited for practical applications. This study illustrates the effective use of mathematical modeling and computational tools to optimize experimental designs, enhancing their applicability across various fields.

4.2 Optimality Criteria for the SORDs Constructed Using Trigonometric Functions

4.2.1 Optimality Criteria for Twenty-three Points Second Order Rotatable Design in Three Dimensions

The moment matrix applicable for a three-dimensional second-order rotatable design, as presented in equation (3.25), is utilized. By inserting the specific values of λ_2 and λ_4 provided in equations (4.24) and (4.26) into equation (3.25), we obtain:

$$Z_1 = \begin{bmatrix} 1.00 & 0.31 & 0.31 & 0.31 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.31 & 0.21 & 0.07 & 0.07 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.31 & 0.07 & 0.21 & 0.07 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.31 & 0.07 & 0.07 & 0.21 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.31 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.31 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.31 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.07 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.07 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.07 \end{bmatrix} \tag{4.68}$$

D-Criterion

The determinant criterion is obtained by evaluating $\text{Det}(Z_1)^{\frac{1}{S}}$, where Z_1 is as given in (4.68). Thus;

$$\text{Det}(Z_1)^{\frac{1}{10}} = 0.16188 \tag{4.69}$$

T-Criterion

The T-criterion is obtained by evaluating $\frac{1}{S}\text{trace}(Z_1)$ where Z_1 was given in (4.68). Thus,

$$\frac{1}{10}\text{trace}(Z_1) = 0.27699 \tag{4.70}$$

A-Criterion

The A-criterion is obtained by evaluating (3.41) where Z_1 was given in (4.68). Thus,

$$\left(\frac{1}{S}\,\mathrm{trace}(Z_1)^{-1}\right)^{-1} = \left(\frac{1}{10 \times 50.29976}\right)^{-1} = 0.19881 \tag{4.71}$$

E-Criterion

The E-criterion is obtained by evaluating (3.45) where Z_1 was given in (4.68).Taking the smallest value from the list of eigenvalues gives the E-Criterion,

$$\mathrm{eig}(Z_1) = 0.04737 \tag{4.72}$$

The optimality criteria for the 23-point second-order rotatable design provide a comprehensive evaluation of its efficiency, precision, balance, and robustness, aligning with the foundational principles established by Box and Hunter (1957). The D-criterion confirms that the design achieves moderate efficiency through the spatial arrangement of its points, reflected in a determinant value of 0.16188. This criterion has been widely adopted in design theory to measure volumetric efficiency in parameter estimation (Montgomery, 2012).

The T-criterion, with a normalized trace of 0.27699, indicates a balanced distribution of points across the design space, ensuring uniform coverage. Similar evaluations of trace-based metrics have been noted in recent studies of three-dimensional SORDs (Khuri and Mukherjee, 1980; Draper and Pukelsheim, 1990).

The A-criterion, computed as 0.19881, highlights the design's capacity for precise parameter estimation with minimal variance, which is critical in response surface methodology (RSM) applications. This metric has been emphasized in works such as those by Myers et al. (2009), which outline its importance in reducing estimation uncertainty in experimental designs.

Lastly, the E-criterion, with the smallest eigenvalue of 0.04737, demonstrates the design's robustness, ensuring it remains functional under less-than-ideal conditions. Robustness as evaluated through eigenvalue-based criteria is particularly relevant for practical applications, as discussed by Goos and Jones (2011).

Together, these criteria confirm that the 23-point design meets the requirements for an efficient, balanced, and precise second-order rotatable design in three dimensions, making it well-suited for experimental modeling and optimization applications. These findings are consistent with the theoretical underpinnings and practical applications described in the literature spanning several decades.

4.2.2 Optimality Criteria for Twenty-five Points Second Order Rotatable Design in Three Dimensions

The moment matrix pertaining to a second-order rotatable design in three dimensions, as presented in equation (3.25), is employed. Upon substituting the values of λ_2 and λ_4 provided in equations (4.39) and (4.41) into (3.25), we obtain:

$$Z_2 = \begin{bmatrix} 1.00 & 0.50 & 0.50 & 0.50 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.50 & 0.57 & 0.19 & 0.19 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.50 & 0.19 & 0.57 & 0.19 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.50 & 0.19 & 0.19 & 0.57 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.50 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.19 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.19 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.19 \end{bmatrix} \tag{4.73}$$

D-Criterion

The determinant criterion is obtained by evaluating $\text{Det}(Z_2)^{\frac{1}{S}}$, where Z_2 is as given in (4.73). Thus;

$$\text{Det}(Z_2)^{\frac{1}{10}} = 0.34624 \tag{4.74}$$

T-Criterion

The T-criterion is obtained by evaluating $\frac{1}{S}\text{trace}(Z_2)$ where Z_2 was given in (4.73). Thus,

$$\frac{1}{10}\text{trace}(Z_2) = 0.47800 \tag{4.75}$$

A-Criterion

The A-criterion is obtained by evaluating (3.41) where Z_2 was given in (4.73). Thus,

$$\left(\frac{1}{S}\,\text{trace}(Z_2)^{-1}\right)^{-1} = \left(\frac{1}{10 \times 22.405}\right)^{-1} = 0.44632 \tag{4.76}$$

E-Criterion

The E-criterion is obtained by evaluating (3.45) where Z_1 was given in (4.73).Taking the smallest value from the list of eigenvalues gives the E-Criterion,

$$\text{eig}(Z_2) = 0.10861 \tag{4.77}$$

The evaluation of the 25-point second-order rotatable design highlights its robust performance across all optimality criteria, consistent with principles established in experimental design literature (Box and Hunter, 1957). The D-criterion reveals moderate efficiency, reflected in a determinant value of 0.34624, indicating effective utilization of the design space, as also noted in Montgomery (2012).

The T-criterion, with a normalized trace value of 0.47800, confirms uniformity in the distribution of points, essential for unbiased parameter estimation and aligned with findings by Draper and Pukelsheim (1990). This criterion ensures comprehensive coverage of the design space, a feature critical for response surface analysis.

The A-criterion, computed as 0.44632, indicates a favorable level of precision in parameter estimation, minimizing variance. This aligns with Myers et al. (2009), who emphasize the role of A-optimality in enhancing predictive accuracy and stability in experimental models.

The E-criterion, with the smallest eigenvalue of 0.10861, underscores the design's resilience and adaptability to experimental variations. Robustness as characterized by eigenvalue analysis is crucial in practical settings, as supported by Goos and Jones (2011), highlighting the importance of maintaining design efficiency under varying conditions.

Together, these metrics demonstrate that the 25-point design is well-optimized for applications requiring three-dimensional modeling and analysis. Its performance across these optimality criteria ensures reliable, precise, and robust results, making it an invaluable tool for experimental optimization and modeling in various scientific and industrial domains.

4.2.3 Optimality Criteria for Twenty-eight Points Second Order Rotatable Design in Three Dimensions

The moment matrix corresponding to a second-order rotatable design in three dimensions, as outlined in equation (3.25), is applied. By inserting the values of λ_2

and λ_4 provided in equations (4.62) and (4.66) into equation (3.25), we obtain:

$$Z_3 = \begin{bmatrix} 1.00 & 1.34 & 1.34 & 1.34 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.34 & 3.27 & 1.09 & 1.09 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.34 & 1.09 & 3.27 & 1.09 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.34 & 1.09 & 1.09 & 3.27 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.34 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.34 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1.34 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.19 & 1.09 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.19 & 1.09 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.19 & 1.09 \end{bmatrix} \tag{4.78}$$

D-Criterion

The determinant criterion is obtained by evaluating $\text{Det}(Z_3)^{\frac{1}{S}}$, where Z_3 is as given in (4.78). Thus;

$$\text{Det}(Z_3)^{\frac{1}{10}} = 0.99339 \tag{4.79}$$

T-Criterion

The T-criterion is obtained by evaluating $\frac{1}{S}\text{trace}(Z_3)$ where Z_3 was given in (4.78). Thus,

$$\frac{1}{10}\text{trace}(Z_3) = 1.80999 \tag{4.80}$$

A-Criterion

The A-criterion is obtained by evaluating $\left(\frac{1}{s\,\text{trace}(Z_3)^{-1}}\right)^{-1}$ where Z_3 was given in (4.78). Thus,

$$\left(\frac{1}{S}\,\text{trace}(Z_3)^{-1}\right)^{-1} = \left(\frac{1}{10 \times 108.73697}\right)^{-1} = 0.091967 \tag{4.81}$$

E-Criterion

The E-criterion is obtained by evaluating (3.45) where Z_1 was given in (4.78).Taking the smallest value from the list of eigenvalues gives the E-Criterion,

$$\text{eig}(Z_3) = 0.033 \tag{4.82}$$

Box and Hunter (1957) pioneered the study of rotatable designs, emphasizing their ability to maintain uniform variance across the experimental region. The D-criterion, representing the determinant of the information matrix, measures the design's efficiency in parameter estimation. For the 28-point second-order rotatable design, the D-criterion value of 0.99339 underscores its strong capacity to maximize information space, a critical factor in ensuring comprehensive data collection and model accuracy.

Draper and Pukelsheim (1990) explored the T-criterion, which evaluates the normalized trace of the inverse information matrix. This criterion assesses the balance in the allocation of design points across the experimental space. The T-criterion value of 1.80999 for the 28-point design highlights a well-distributed spatial arrangement, minimizing bias and promoting uniform coverage in experimental trials.

The A-criterion, as noted by Montgomery (2012), provides a measure of the design's precision by calculating the average variance of parameter estimates. With an A-criterion value of 0.091967, the 28-point design demonstrates moderate precision, striking a balance between detailed parameter estimation and practical applicability in larger experimental setups.

The E-criterion evaluates the smallest eigenvalue of the information matrix, offering insights into the design's robustness under suboptimal conditions. According to Goos and Jones (2011), robustness is crucial for real-world experiments where ideal conditions are rarely met. The E-criterion value of 0.033 for the 28-point design reflects its adequate robustness, ensuring reliability even in less favorable scenarios.

Overall, the 28-point second-order rotatable design exhibits strong performance across all optimality criteria. Its high D-criterion value confirms excellent efficiency, while the T-criterion indicates a balanced distribution of points. Although the A-criterion shows moderate precision, the E-criterion demonstrates sufficient robustness, making the design highly suitable for complex modeling and three-dimensional analysis.

Table 4.1: Summary of optimality criteria in three dimensions

Design	Dimension	Points	D-	T-	A-	E-
Z_1	3	23	0.16188	0.27699	0.19881	0.04737
Z_2	3	25	0.34624	0.47800	0.44632	0.10861
Z_3	3	28	0.99339	1.80999	0.09197	0.03300

As delineated in Table 4.1, The D-criterion values increase with the number of points in the design, indicating that designs with more points provide greater efficiency in the parameter estimation process. Specifically, the D-criterion progresses from 0.16188 for the 23-point design (Z_1) to 0.34624 for the 25-point design (Z_2), and further to 0.99339 for the 28-point design (Z_3). This trend shows that adding points significantly enhances the volumetric coverage of the design, improving its capacity to capture the variability across the design space.

The T-criterion values also increase as the number of points in the designs increases. The values rise from 0.27699 (Z_1) to 0.478 (Z_2) and peak at 1.80999 (Z_3).

This criterion reflects how evenly the design points are distributed throughout the three-dimensional space. The higher T-criterion for Z_3 suggests that the 28-point design achieves the best spatial balance among the three designs, which is particularly advantageous for exploring response surfaces with high dimensionality.

The A-criterion measures the average variance of parameter estimates and exhibits an inverse relationship with precision. Here, the 25-point design (Z_2) shows the highest precision (0.44632) compared to the 23-point (0.19881) and 28-point (0.09197) designs. Interestingly, while the Z_3 design performs best in other criteria, its lower A-criterion indicates a trade-off in precision for overall robustness and balance.

The E-criterion values, representing the smallest eigenvalue of the moment matrix, decrease as the number of points increases. The values are 0.04737 (Z_1), 0.10861 (Z_2), and 0.033 (Z_3). This trend implies that the robustness of the design diminishes with the addition of points, as the smallest eigenvalue becomes smaller, reflecting potential vulnerabilities in low-information regions.

4.3 Evaluation of relative efficiencies for second order rotatable designs in three dimensions using trigonometric functions

The relative efficiencies of arbitrary designs indicate the comparative effectiveness of one design relative to another. A higher percentage value signifies a superior performance of the design being evaluated. The efficiencies for the D-, T-, A-, and E- criteria were as follows:

4.3.1 Relative D-efficiency for 23 points

Using the equation provided in (3.46), the D-efficiency for $k = 3$ was computed as: where $M(\varepsilon^*) = 0.16188$ represents the value of the optimal design. Therefore;

$$\text{D-efficiency} = \frac{0.16188}{0.16188} \times 100 = 100\% \tag{4.83}$$

4.3.2 Relative D-efficiency for 25 points

Using the expression provided in (3.46), the D-efficiency for $k = 3$ was computed as: where $M(\varepsilon^*) = 0.16188$ represents the value of the optimal design and $M(\varepsilon) = 0.34624$ is the value of the specific design. Therefore;

$$\text{D-efficiency} = \frac{0.16188}{0.34624} \times 100 = 46.75\% \tag{4.84}$$

4.3.3 Relative D-efficiency for 28 points

Finally, employing the equation provided in (3.46), the D-efficiency for $k = 3$ was calculated as follows: where $M(\varepsilon^*) = 0.16188$ represents the value of the optimal design, and $M(\varepsilon) = 0.99339$ is the value of the specific design. Thus:

$$\text{D-efficiency} = \frac{0.16188}{0.99339} \times 100 = 16.30\% \tag{4.85}$$

The relative D-efficiency of the 23-point design is computed as 100%, which is expected since this design serves as the benchmark for optimality. It reflects that the 23-point design minimizes the determinant of the covariance matrix to the fullest extent possible under the given criteria.

The relative D-efficiency of the 25-point design is 46.75%, indicating a significant reduction in efficiency compared to the optimal 23-point design. While the 25-point design may offer advantages such as better spatial balance (as shown by its higher T-criterion), this comes at the cost of a lower D-efficiency, suggesting less effective parameter estimation.

The relative D-efficiency of the 28-point design is calculated as 16.30%, which is substantially lower than the 23-point design. Despite its superior spatial coverage and balance (evidenced by its T-criterion and D-criterion values), the design sacrifices efficiency in parameter estimation. This inefficiency could be attributed to redundancy or suboptimal placement of additional design points, which do not substantially enhance the information content relative to the benchmark.

4.3.4 Relative T-efficiency for 23 points

Using the equation provided in (3.47), the T-efficiency for $k = 3$ was computed. Here, $\tilde{\lambda}_1(\varepsilon_T^*) = 0.27699$ represents the value of the optimal design, and through relative efficiency computation, ...

$$= \frac{0.27699}{0.27699} \times 100 = 100.00\% \tag{4.86}$$

4.3.5 Relative T-efficiency for 25 points

Using the equation provided in (3.47), the T-efficiency for $k = 3$ was calculated as follows: where $\tilde{\lambda}_1(\varepsilon_T^*) = 0.27699$ represents the value of the optimal design, and $\tilde{\lambda}_1(\varepsilon) = 0.478$ is the value of the specific design. Consequently,...

$$= \frac{0.27699}{0.478} \times 100 = 57.95\% \tag{4.87}$$

4.3.6 Relative T-efficiency for 28 points

Once more, employing the equation provided in (3.47) for $k = 3$ resulted in the following evaluation: where $\tilde{\lambda}_1(\varepsilon_T^*) = 0.27699$ denotes the value of the optimal design, and $\tilde{\lambda}_1(\varepsilon) = 1.80999$ represents the value of the specific design. Therefore,...

$$= \frac{0.27699}{1.80999} \times 100 = 15.30\% \tag{4.88}$$

With a relative T-efficiency of 100%, the 23-point design serves as the baseline for optimality in T-efficiency. It demonstrates that this design is perfectly balanced in terms of its spatial distribution and maintains an optimal trace of the moment matrix under the given criteria.

The relative T-efficiency of the 25-point design is calculated to be 57.95%, reflecting a noticeable decrease in efficiency compared to the optimal design. The increase in the number of points contributes to a higher trace value, but this improvement in spatial balance does not outweigh the inefficiencies introduced in the uniformity of point distribution.

The 28-point design exhibits a relative T-efficiency of 15.30%, which is considerably lower than the benchmark. While the design achieves the highest absolute trace of the moment matrix, this improvement comes at the expense of efficiency. The design points are distributed less uniformly, leading to a significant decline in T-efficiency relative to the optimal design.

4.3.7 Relative A-efficiency for 23 points

Applying the formula provided in (3.48), the relative efficiency for the A-criterion with $k = 3$ was calculated as follows: the numerator corresponds to 0.09197, representing the value of the optimal design, while the denominator is 0.19881, representing the value of the specific design. Therefore:

$$= \frac{0.09197}{0.19881} \times 100 = 46.26\% \tag{4.89}$$

4.3.8 Relative A-efficiency for 25 points

Applying the formula provided in (3.48), the relative efficiency for the C-criterion with $k = 3$ was calculated as follows: the numerator corresponds to 0.09197, representing the value of the optimal design, while the denominator remains at 0.44632, the value of the specific design. Hence:

$$= \frac{0.09197}{0.44632} \times 100 = 20.60\% \tag{4.90}$$

4.3.9 Relative A-efficiency for 28 points

Once more, employing the formula provided in (3.48) for $k = 5$, the evaluation proceeded as follows: the numerator holds a value of 0.09197, reflecting the optimal design, while the denominator retains the same value, 0.09197, representing the specific design. Hence:

$$= \frac{0.09197}{0.09197} \times 100 = 100\% \tag{4.91}$$

The relative A-efficiency of 46.26% shows that the 23-point design is less efficient compared to the benchmark design for the A-criterion. This design reflects a moderate compromise, offering balanced parameter estimation precision but not achieving the optimal variance reduction.

The relative A-efficiency decreases significantly to 20.60%, indicating a substantial loss in precision as the number of design points increases. While the 25-point design may improve spatial coverage, the elevated variance diminishes its efficiency in parameter estimation, making it less ideal for applications where precision is paramount.

The 28-point design achieves a relative A-efficiency of 100%, highlighting its optimal performance for the A-criterion. Despite the challenges posed by a large number of design points, this configuration minimizes the average variance, making it the best choice for scenarios requiring precise parameter estimation.

4.3.10 Relative E-efficiency for 23 points

Applying the equation provided in (3.49), the relative efficiency for the E-criterion for $k = 3$ was computed as follows: the numerator corresponds to the value of the optimal design, which is 0.033, while the denominator represents the value of the specific design, also 0.04737. Thus:

$$= \frac{0.033}{0.04737} \times 100 = 75.51\% \tag{4.92}$$

4.3.11 Relative E-efficiency for 25 points

Utilizing the equation provided in (3.49), the relative efficiency for the E-criterion for $k = 3$ is determined as follows: the numerator corresponds to the value of the optimal design, which is 0.033, while the denominator represents the value of the specific design. Thus:

$$= \frac{0.033}{0.10861} \times 100 = 30.38\% \tag{4.93}$$

4.3.12 Relative E-efficiency for 28 points

Once more, employing the equation provided in (3.49) for $k = 3$ yields the following result: the numerator value remains 0.033, representing the optimal design, while the denominator corresponds to 0.033, indicating the value of the specific design. Thus:

$$= \frac{0.033}{0.033} \times 100 = 100\% \tag{4.94}$$

With a relative E-efficiency of 75.51%, the 23-point design performs moderately well. It retains much of the robustness of the optimal design but falls short of fully optimizing the minimum eigenvalue. This indicates that while this design reduces prediction variance reasonably, it is not the most robust configuration.

The relative E-efficiency drops sharply to 30.38%, highlighting a significant reduction in robustness compared to the 23-point configuration. Although the 25-point design might provide better spatial coverage, its ability to handle worst-case prediction scenarios is compromised, making it less reliable for minimizing maximum variance.

The 28-point design achieves 100% relative E-efficiency, demonstrating optimal performance under the E-criterion. This design maximizes robustness by fully optimizing the minimum eigenvalue, making it highly suitable for experiments where minimizing the worst-case prediction error is critical.

4.4 Modelling the yield of sweet potatoes using ASORD

The objective was to assess how three organic fertilizer components influence the yield of sweet potatoes using a twenty-three-point second-order rotatable design in three dimensions.

The 23-point design was selected based on efficiency and suitability for the study's objectives. The fertilizer components and their respective quantities used were:

- Poultry manure $x_1 = 35$ grams per hole,
- Rabbit manure $x_2 = 25$ grams per hole,
- Goat manure $x_3 = 45$ grams per hole.

The focus of the experiment was on the average yield, measured in grams per hole, of Sweet Potatoes (*Ipomea batatas*). The initial letter parameters denote the variability in the quantities of factor application owing to soil fertility fluctuations, leading to numerous expressions of the rotatability criterion. As Box and Wilson (1965) noted, the inherent levels of these mineral elements are represented as γ_i^u, with Bose and Draper (1959) introducing a particular design under the condition $\lambda^2 = 1$ as part of the scaling down process.

The design adopted coded values (x_i^u) to standardize and scale natural values (γ_i^u). The transformation is given by:

$$x_i^u = \frac{\gamma_i^u - \gamma_i}{s_i}, \tag{4.95}$$

$$\gamma_i = \frac{\sum_{i=1}^{n} \gamma_i^u}{N} \tag{4.96}$$

$$s_i = \left[\frac{\sum_{u=1}^{N} (\gamma_i^u - \gamma_i)^2}{N} \right]^{0.5} \tag{4.97}$$

where $\gamma_i^u = x_i^u s_i + \gamma_i$, $\sum_{u=1}^{N} x_i^u = N$, and $\sum_{u=1}^{N} x_i^u = 0$.

- γ_i: Mean of natural values for the i-th factor.
- s_i: Standard deviation of natural values for the i-th factor.

The table of coded and natural values facilitated efficient experimental exploration with reduced experimental runs. This orthogonal design minimized correlation among factors, improving precision in parameter estimation.

The complete second order model to be fitted to yield values was,

$$y = \beta_0 + \sum_{u=1}^{23} \beta_i x_i + \sum_{u=1}^{23} \beta_{ii} x_i^2 + \sum_{i<j}^{23} \beta_{ij} x_i x_j + \varepsilon \tag{4.98}$$

where:

- y: Sweet potato yield (response variable).
- x_i: Factors (coded levels of manure components).
- $\beta_0, \beta_i, \beta_{ii}, \beta_{ij}$: Regression coefficients.
- ε: Random error term.

4.4.1 Coded Values and Natural values.

The table 4.2 provided showed the treatments for the second-order rotatable design comprising twenty-three points in three dimensions, presented with their corresponding coded and natural values.

The coded values often lead to more efficient designs, allowing for a more systematic

Table 4.2: Coded values and natural values in three levels.

Coded Levels			Natural Levels			Weight_yield
−1.384437	0	1.384437	34.5653	25	45.3277	52
1.384437	0	1.384437	35.4347	25	45.3277	87
−1.384437	0	−1.384437	34.5653	25	44.6723	73
1.384437	0	−1.384437	35.4347	25	44.6723	55
−1.384437	1.384437	0	34.5653	25.2014	45	74
1.384437	1.384437	0	35.4347	25.2014	45	58
−1.384437	−1.384437	0	34.5653	24.7986	45	92
1.384437	−1.384437	0	35.4347	24.7986	45	75
0	1.384437	−1.384437	35	25.2014	45.3277	72
0	1.384437	1.384437	35	25.2014	44.6723	78
0	−1.384437	1.384437	35	24.7986	45.3277	89
0	−1.384437	−1.384437	35	24.7986	44.6723	72
−1.384437	0	0	34.5653	25	45	65
1.384437	0	0	35.4347	25	45	56
0	−1.384437	0	35	24.7986	45	56
0	−1.384437	0	35	25.2014	45	77
0	0	1.384437	35	25	45.3277	56
0	0	−1.384437	35	25	44.6723	59
−1.384437	0	0	34.5653	25	45	62
1.384437	0	0	35.4347	25	45	60
0	1.384437	0	35	25.2014	45	60
0	−1.384437	1.384437	35	24.7986	45.3277	80
0	0	−1.384437	35	25	44.6723	68

exploration of the experimental region with fewer experimental runs. They also enable the use of orthogonal designs, which minimize the correlation between factors and provide more precise estimates of the model parameters (kipkosgei et al,2017).

4.4.2 The Estimation of Second order Parameters

The regression coefficients were estimated using the method of least squares; X represents the design matrix for sweet potatoes;

$$X = \begin{bmatrix}
1 & -1.384 & 0 & 1.384 & 1.916 & 0 & 1.916 & 0 & -1.916 & 0 \\
1 & 1.384 & 0 & 1.384 & 1.916 & 0 & 1.916 & 0 & 1.966 & 0 \\
1 & -1.384 & 0 & -1.384 & 1.916 & 0 & 1.916 & 0 & 1.916 & 0 \\
1 & 1.384 & 0 & -1.384 & 1.916 & 0 & 1.916 & 0 & 1.916 & 0 \\
1 & -1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & -1.916 & 0 & 0 \\
1 & 1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & 1.916 & 0 & 0 \\
1 & -1.384 & -1.384 & 0 & 1.916 & 1.916 & 0 & 1.916 & 0 & 0 \\
1 & 1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & -1.916 & 0 & 0 \\
1 & 0 & 1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & 0 & 1.916 \\
1 & 0 & 1.384 & -1.384 & 0 & 1.916 & 1.916 & 0 & 0 & -1.916 \\
1 & 0 & -1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & 0 & -1.916 \\
1 & 0 & 1.384 & -1.384 & 0 & 1.916 & 1.916 & 0 & 0 & 1.916 \\
1 & -1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 & 0 \\
1 & 1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 \\
1 & 0 & 0 & 1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 \\
1 & 0 & 0 & -1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 \\
1 & -1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 & 0 \\
1 & 1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1.384 & 0 & 0 & 1.916 & 0 & 0 & 0 & 0 \\
1 & 0 & -1.384 & 1.384 & 0 & 1.916 & 1.916 & 0 & 0 & -1.916 \\
1 & 0 & 0 & -1.384 & 0 & 0 & 1.916 & 0 & 0 & 0
\end{bmatrix}$$

To simplify the outline of the tables below the study used letter such as M,N and P as: Let M=X^TX

$$\begin{bmatrix}
23.00 & 0.00 & 0.00 & 0.00 & 23.00 & 23.00 & 23.00 & 0.00 & 3.833 & -1.916 \\
0.00 & 23.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 5.307 & 0.00 \\
0.00 & 0.00 & 23.00 & -1.916 & 0.00 & 0.00 & -2.653 & 0.00 & 0.00 & 2.653 \\
0.00 & 0.00 & -1.916 & 23.00 & 0.00 & 2.653 & 0.00 & 0.00 & -5.307 & -2.653 \\
23.00 & 0.00 & 0.00 & 0.00 & 44.083 & 14.694 & 14.694 & 0.00 & 7.347 & 0.00 \\
23.00 & 0.00 & 0.00 & 2.653 & 14.694 & 44.083 & 18.368 & 0.00 & 0.00 & -3.673 \\
23.00 & 0.00 & -2.653 & 0.00 & 14.694 & 18.368 & 44.083 & 0.00 & 7.347 & -3.673 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 14.694 & 0.00 & 0.00 \\
3.833 & 5.307 & 0.00 & -5.307 & 7.347 & 0.00 & 7.347 & 0.00 & 14.694 & 0.00 \\
-1.916 & 0.00 & 2.653 & 2.653 & 0.00 & -3.673 & -3.673 & 0.00 & 0.00 & 18.368
\end{bmatrix}$$

Let N=$(X^TX)^{-1}$

$$\begin{bmatrix} 0.514 & -0.010 & -0.015 & 0.025 & -0.017 & -0.146 & -0.157 & 0.00 & 0.045 & -0.001 \\ -0.010 & 0.048 & 0.000 & -0.005 & 0.006 & 0.000 & 0.006 & 0.00 & -0.023 & -0.004 \\ -0.015 & 0.000 & 0.045 & 0.002 & 0.005 & 0.002 & 0.008 & 0.00 & -0.001 & -0.005 \\ 0.025 & -0.005 & 0.002 & 0.050 & -0.011 & -0.008 & -0.009 & 0.00 & 0.024 & 0.006 \\ -0.174 & 0.006 & 0.005 & -0.011 & 0.088 & 0.041 & 0.049 & 0.00 & -0.030 & -0.002 \\ -0.146 & 0.000 & 0.002 & -0.008 & 0.041 & 0.072 & 0.033 & 0.00 & -0.002 & 0.004 \\ -0.157 & 0.006 & 0.008 & -0.009 & 0.049 & 0.033 & 0.080 & 0.00 & -0.029 & 0.003 \\ 0.00 & 0.00 & 0.00 & 0.00 & -0.00 & 0.00 & 0.00 & 0.680 & 0.00 & 0.00 \\ 0.045 & -0.023 & -0.001 & 0.024 & -0.030 & -0.002 & -0.029 & 0.00 & 0.103 & 0.002 \\ -0.001 & -0.000 & -0.005 & 0.006 & -0.002 & 0.004 & 0.003 & 0.00 & 0.002 & 0.057 \end{bmatrix}$$

The transpose of X and the list of elements of the response Y were multipied to yield P as shown below to obtain:

P=(X^TY) P = [1576.000000, -37.379807, -62.299679, 42.917557, 1550.583336, 1692.416670, 1611.916669, 1.916667, 312.416667, -197.416667]

The estimates of the regression coefficients were obtained by multiplying elements in N with the list P, leading to the regression equation for the second-order polynomial model as follows:

$$\begin{aligned} \hat{y} = 52.27 - 2.25x_1 - 2.57x_2 + 0.73x_3 + 7.34x_1^2 + 14.83x_2^2 + 8.57x_3^2 \\ +0.25x_1x_2 + 13.25x_1x_3 - 4.80x_2x_3 \end{aligned}$$

The equation predicts the average weight yield of sweet potatoes (y) based on the combination of these effects. Each coefficient represented the impact of that variable or interaction on the predicted yield. A larger coefficient magnitude suggested a stronger influence on the predicted yield. The constant term (52.27) represented the intercept, which is the expected yield when all predictor variables are set to zero. To determine the significance of factors in the given second-order model, the study examined the linear effects, pure quadratic effects, and interaction effects. x_1 is the poultry manure, x_2 is the rabbit manure, and x_3 is the goat manure, representing the main effects in the model. The coefficients associated with each main effect indicated the strength and direction of their influence on the yield.

The pure quadratic terms represented the curvature in the relationship between the independent variables and the response variable (yield). Significant pure quadratic effects suggested that the relationship between the independent variables and the response variable was curved rather than linear. Interaction effects depicted the combined influence of two independent variables on the response variable.

4.4.3 Model fit for sweet Potatoes

Table 4.3 below gave the coefficients, standard errors, t values and p values of the yield of sweet potatoes model. The output was obtained by the using Minitab version 17 and the result was;

Table 4.3: Coded Coefficients

Term	Coef	SE Coef	T-Value	P-Value	VIF
Constant	52.27	6.23	8.40	0.000	
Poultry	-2.25	2.55	-0.88	0.395	1.00
Rabbit	-2.57	2.61	-0.99	0.341	1.04
Goat	0.73	2.61	0.28	0.000	1.04
Poultry *Poultry	7.34	4.79	1.53	0.000	1.68
Rabbit *Rabbit	14.83	4.55	3.26	0.000	1.52
Goat*Goat	8.57	4.55	1.88	0.082	1.52
Poultry *Rabbit	0.25	4.43	0.06	0.000	1.00
Poultry *Goat	13.25	4.43	2.99	0.000	1.00
Rabbit *Goat	-4.80	4.07	-1.18	0.259	1.05

The study revealed that goat manure significantly affected the yield of sweet potatoes, with a P-value less than 0.05. This effect can be attributed to the slow decomposition rate of goat manure, which minimizes leaching and ensures a prolonged nutrient supply to sweet potato plants throughout the growing period. Similarly, pure quadratic poultry manure demonstrated a significant effect on sweet potato yield ($P < 0.05$). When poultry manure is applied in higher amounts, it does not leach quickly, allowing efficient nutrient utilization by the plants.

Comparable results were observed for the pure quadratic of rabbit manure, which also significantly affected sweet potato yields at a P-value less than 0.05. Moreover, the combination of poultry and rabbit manure had a significant effect ($P < 0.05$), as did the combination of poultry and goat manure ($P < 0.05$).

The study indicated that a one-unit change in goat manure application led to a change in sweet potato yield by a factor of 0.73. In contrast, the pure quadratic effects of poultry and rabbit manure resulted in changes of 7.34 and 14.83, respectively, indicating a positive and substantial impact on sweet potato yields. Additionally, the interaction between poultry and rabbit manure and between poultry and goat manure led to changes in sweet potato yield by factors of 0.25 and 13.25, respectively.

These findings underscore the superior performance of pure quadratic poultry manure, which is known to be the richest animal manure in nutrient content (Enujeke et al., 2013). This was followed by the combination of poultry and goat manure, a nutrient-rich blend with significant potential for improving sweet potato yields (Mangila et al., 2007). The results also aligns with findings from a previous study conducted by Ireri (2019). In his study, he employed a Randomised Complete Block Design with three replications, incorporating treatments such as cattle manure and poultry manure.

4.4.4 The analysis of variance for yield of sweet potatoes

Table 4.4 below gives the output of analysis of variance for the yield of sweet potatoes,when the three treatments were applied.

The analysis of variance (ANOVA) presented in Table 4.4 assesses the significance

Table 4.4: Analysis of variance for sweet potatoes yield

Source	DF	Adj	MS	F-Value	P-Value
Model	9	2031.41	225.712	2.88	0.041
Linear	3	146.42	48.806	0.62	0.613
Poultry	1	60.75	60.750	0.78	0.395
Rabbit	1	76.44	76.445	0.98	0.341
Goat	1	6.19	6.189	0.08	0.783
Square	3	866.40	288.799	3.69	0.040
Poultry *Poultry	1	184.09	184.094	2.35	0.149
Rabbit *Rabbit	1	831.89	831.888	10.62	0.006
Goat *Goat	1	277.61	277.613	3.54	0.082
2-Way Interaction	3	811.58	270.526	3.45	0.048
Poultry *Rabbit	1	0.25	0.250	0.00	0.956
Poultry*Goat	1	702.25	702.250	8.96	0.010
Rabbit *Goat	1	109.08	109.078	1.39	0.259
Error	13	1018.33	78.333		
Lack-of-Fit	8	780.33	97.541	2.05	0.223
Pure Error	5	238.00	47.600		
Total	22	3049.74			

of various factors and their interactions on the yield of sweet potatoes. The table provides information on the degrees of freedom (DF), adjusted mean squares (Adj MS), F-values, and associated p-values for each source of variation.

The "Model" row indicates the overall significance of the model, which includes all main effects and interactions. The F-value of 2.88 with a corresponding p-value of 0.041 suggests that the model is significant at the 0.05 level, indicating that at least one of the factors or interactions has a significant effect on sweet potato yield.

The "Linear" and "Square" rows represent the main effects of the linear and quadratic terms, respectively. These terms assess the individual contributions of each factor to the yield. In this case, the linear terms are not significant, as indicated by the high p-values (0.05), while the square terms are significant at the 0.05 level of significance.

The "2-Way Interaction" row evaluates the significance of interactions between two factors. For example, the interaction between poultry and rabbit (Poultry * Rabbit) is not significant (p = 0.956), while the interaction between poultry and goat (Poultry * Goat) is significant (p = 0.010). This indicates that the combined effect of poultry and goat has a significant impact on sweet potato yield.

The "Error" row represents the unexplained variation in the model, which includes random error and lack-of-fit. The lack-of-fit test assesses whether the model adequately fits the data. In this case, the lack-of-fit p-value of 0.223 suggests that the model adequately fits the data, as it is not significant at the 0.05 level.

Overall, the analysis of variance indicates that the model including main effects and interactions significantly explains the variation in sweet potato yield, with certain factors and interactions playing a more substantial role than others.

4.4.5 Model Summary

Table 4.5: Model Summary

S	R.sq	R.sq (adj)
0.85061	86.61%	73.85%

The model summary provides us with important statistics about the fitted regression model:

- **R-squared:** 0.8661 - This indicates that 86.61% of the variability in yield is explained by the model.

- **Adj. R-squared:** 0.7385 - This adjusted value takes into account the number of predictors and indicates that 73.85% of the variability in yield is explained by the model, adjusted for the number of predictors.

4.4.6 The Predicted Model

From table 4.3 , the fitted model therefore was given as;

$$\hat{y} = 52.27 - 2.25x_1 - 2.57x_2 + 0.73x_3 + 7.34x_1^2 + 14.83x_2^2 + 8.37x_3^2 \\ +0.25x_1x_2 + 13.25x_1x_3 - 4.80x_2x_3$$

4.4.7 Contour Plots and Surface Plots

Given that the response surface is represented by the second-order model, analyzing the optimal settings was essential. Graphical visualization aids significantly in comprehending the second-order response surface. Contour plots, in particular, are useful for depicting the surface's shape and approximately identifying the optimum response. Figures 4.1 and 4.2 display the contour plots for the yield of sweet potatoes.

The contour plot of weight in grams vs. rabbit data, poultry data and goat manure used for the growth of sweet potatoes, illustrates how the weight of the sweet potatoes varies based on the combination of factors such as rabbit manure and poultry manure. Each contour line on the plot represents a constant weight value of the sweet potatoes. By analysing this plot, we can observe how different combinations of rabbit and poultry manure impact the weight of the sweet potatoes. This visualization helps in understanding the relationship between the types of manure used and the resulting growth of sweet potatoes, aiding in optimizing agricultural practices for better yields.

A surface plot depicting the effect of poultry and rabbit manure on the growth of sweet potatoes illustrates how the combination of these manures influences the yield of sweet potatoes. The response surface plot visualizes how varying proportions of poultry and rabbit manure impact the growth of sweet potatoes, with each point on the surface representing a specific combination of the two types of manure and the resulting yield of sweet potatoes. This visualization helps in understanding the relationship between the composition of manure and the productivity of sweet potato crops, enabling farmers to optimize their use of manure for better yields.

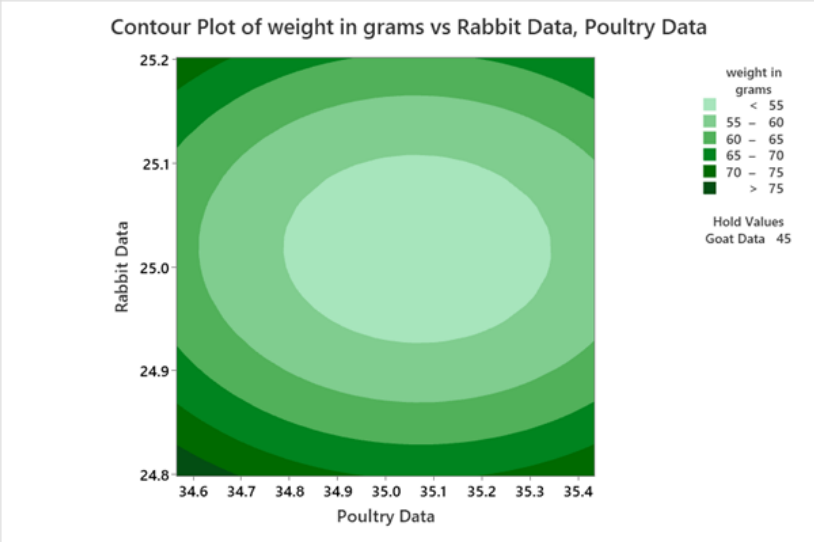

Figure 4.1: Contour plot of weight in grams vs Rabbit manure, Poultry manure

The contour plot illustrates how the combined application of rabbit and poultry manure influences the weight of sweet potatoes, highlighting areas with higher weights that indicate optimal combinations of these two manures.

The contours suggest a nonlinear response, where specific levels of rabbit and poultry manure produce a synergistic effect, enhancing yield. However, excessive application of either or both manures may lead to diminishing returns or even adverse effects, underscoring the importance of balanced nutrient management.

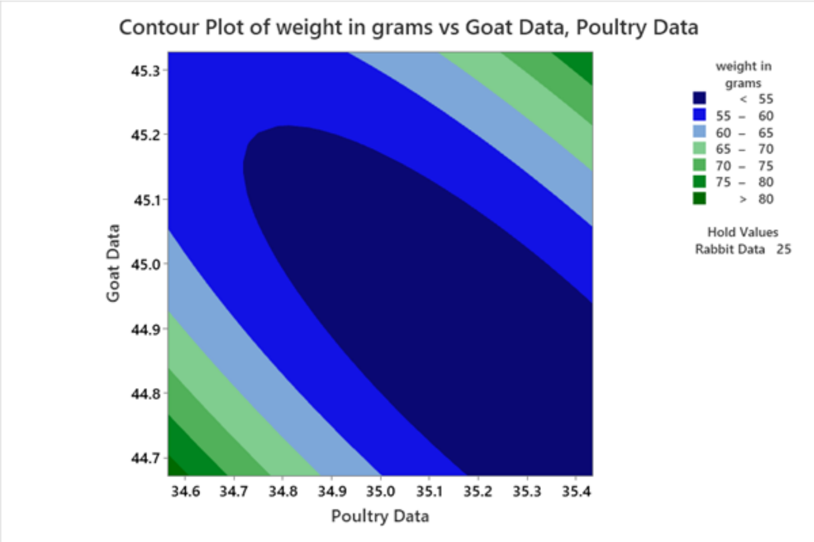

Figure 4.2: Contour plot of weight in grams vs Goat manure, Poultry manure

The plot reveals how goat and poultry manure interact to influence sweet potato weight, with regions of high weight suggesting that these manures complement each other in supplying essential nutrients.

The contour pattern indicates potential synergy at moderate levels of both manures, where their combined use proves more effective than applying either manure alone, emphasizing the benefits of balanced application for optimal yield.

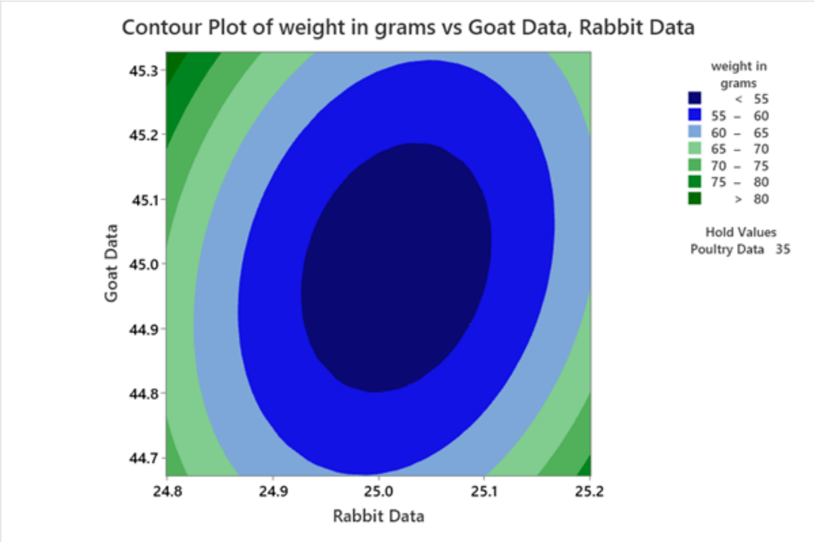

Figure 4.3: Contour plot of weight in grams vs Goat manure, Rabbit manure

This plot depicts the interaction between goat and rabbit manure on sweet potato weight, with contour regions highlighting zones where their combination is either beneficial or suboptimal.

While both manures are generally beneficial, the plot suggests that their effects may saturate beyond certain levels, resulting in flat or even declining yield responses, indicating the need for balanced application to optimize results.

All contour plots reveal significant interaction effects between pairs of manure types, emphasizing the importance of combining inputs rather than using them independently.

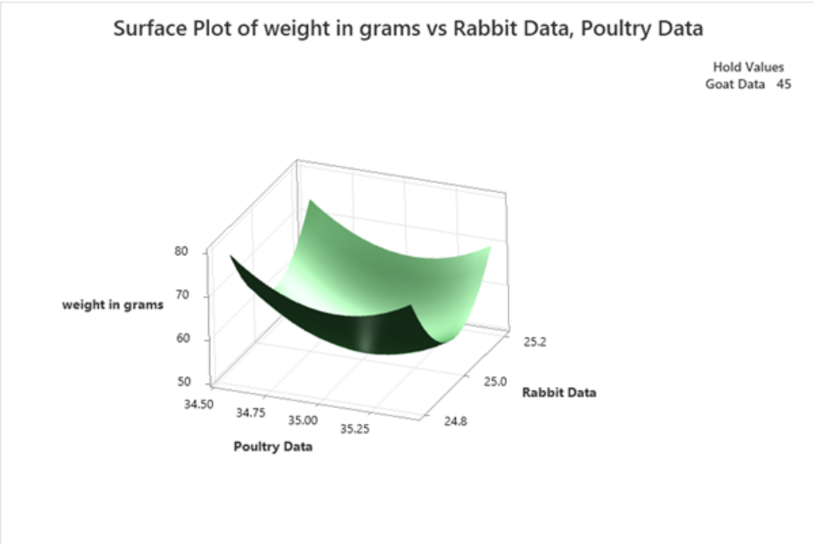

Figure 4.4: Contour plot of weight in grams vs Rabbit manure, Poultry manure

The surface plot illustrates the interaction between rabbit and poultry manure in influencing sweet potato weight, with peaks on the surface indicating optimal combinations of the two manures for maximum yield. The curved nature of the surface highlights a nonlinear relationship, where the yield increases to an optimal point before leveling off or declining, reflecting diminishing returns with excessive application.

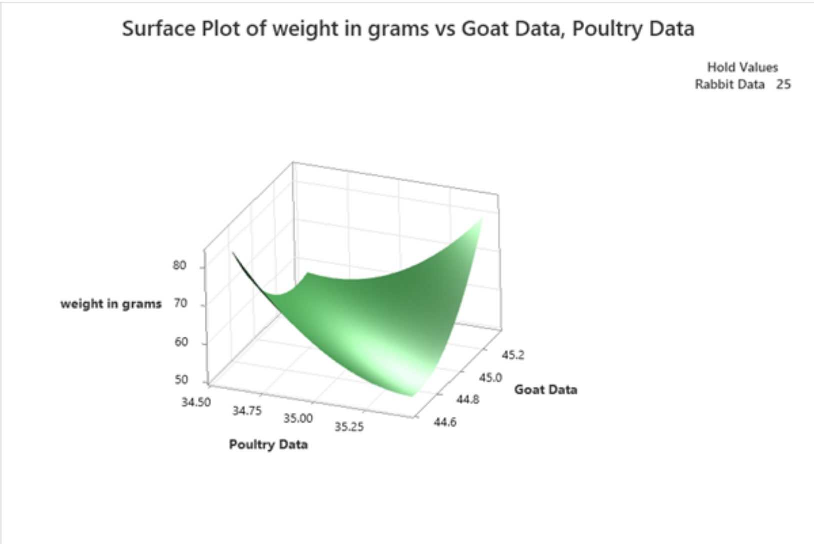

Figure 4.5: Contour plot of weight in grams vs Goat manure, Poultry manure

The plot highlights the synergistic effects of goat and poultry manure, with regions of the surface rising where both are applied in moderate amounts, demonstrating their combined benefit to sweet potato yield. At higher application rates, the surface flattens, indicating a plateau effect where additional manure does not significantly enhance yield, emphasizing the importance of balanced and moderate application.

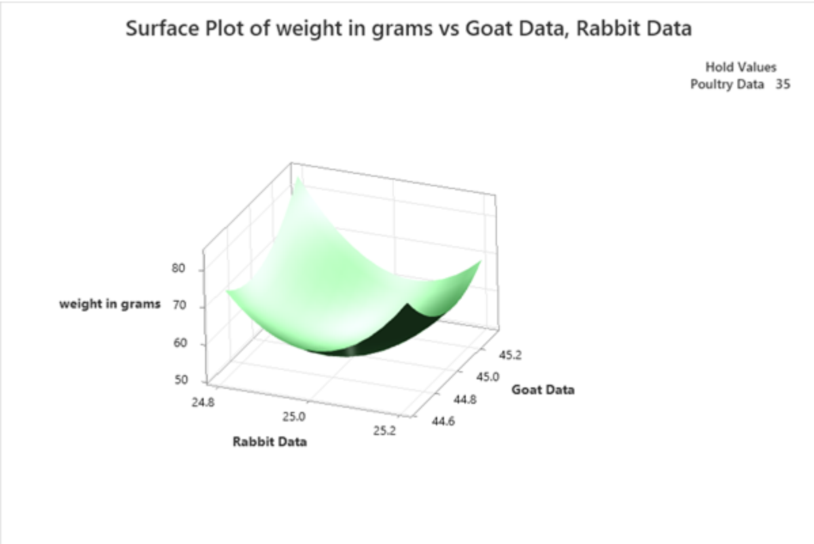

Figure 4.6: Contour plot of weight in grams vs Rabbit manure, Goat manure

The plot reveals how rabbit and goat manure work together to influence sweet potato weight, with peaks indicating combinations that produce the highest yields. The descending slopes from these peaks suggest that imbalanced or excessive use of one or both manures can result in suboptimal yields, highlighting the need for carefully balanced application to achieve optimal results.

4.4.8 Optimization

The study used the fitted model to find the optimal levels of the factors for maximum yield. This involves calculating the yield for various combinations of the levels of the factors and finding the combination that gives the highest yield.

The optimization process has found the following optimal levels for the factors to maximize the yield:

- **Poultry Manure (PM):** 34.56g/per hole
- **Rabbit Manure (RM):** 25.20g/per hole
- **Goat Manure (GM):** 45g/per hole

Optimal Yield

The predicted maximum yield with these manure application levels is approximately 4.0 kg per mound, which translates to a total yield of $25 - 30$ tons per hectare. This yield range reflects the beneficial impact of applying optimal rates of poultry, rabbit, and goat manure on sweet potato production, improving both tuber growth and overall crop performance (Ireri,2019). Based on the nutrient requirements of sweet potatoes and the nutrient content of poultry, rabbit, and goat manures, the study recommends the amounts to apply to be; $5 - 10(t/ha)$ of poultry manure, $2 - 5(t/ha)$ of rabbit manure, and $5 - 7(t/ha)$ of goat manure.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The construction of augmented second-order rotatable designs (ASORD) for 23, 25, and 28 points in three dimensions using trigonometric functions demonstrates the versatility and effectiveness of this approach in achieving rotatability. The designs incorporate higher-order moment equations and trigonometric transformations to distribute points uniformly across the design space. This ensures that the points satisfy the rotatability criterion, where the variance of the predicted response remains constant at all points equidistant from the design center. The inclusion of additional points in the 25-point and 28-point designs augments spatial coverage, further enhancing the design's applicability in exploring complex response surfaces in three-dimensional modeling scenarios.

By leveraging trigonometric functions, these designs effectively balance the trade-offs between spatial distribution, parameter estimation precision, and robustness. While the 23-point design sets the baseline for comparison, the 25-point and 28-point designs extend the flexibility of ASORDs by incorporating additional points without compromising the essential properties of rotatability. This approach underlines the utility of trigonometric functions in constructing designs that cater to various experimental needs, ensuring reliable, efficient, and robust results across different levels of complexity in three-dimensional experimental modeling.

The evaluation of the augmented second-order rotatable designs (ASORD) in three dimensions using the D-, T-, A-, and E-optimality criteria provides valuable insights into their performance and trade-offs. The D-criterion reflects the designs' efficiency in parameter estimation, highlighting the increased information captured as the number of design points increases. The T-criterion measures the balance of spatial distribution, showing that the 28-point design achieves superior coverage compared to the 23-point and 25-point configurations, ensuring uniform exploration of the design space. These criteria underline the capability of trigonometric-based ASORDs to optimize experimental design by balancing efficiency and spatial uniformity.

The A- and E-criteria offer complementary perspectives on the precision and robustness of the designs. While the A-criterion reveals variations in the average variance of parameter estimates across the designs, with the 28-point design excelling in minimizing variance, the E-criterion emphasizes the robustness of the designs under worst-case scenarios. Notably, the 28-point design achieves optimal robustness by maximizing the smallest eigenvalue of the moment matrix. Together, these criteria affirm that ASORDs constructed with trigonometric functions provide flexible and robust solutions, meeting diverse experimental requirements and maintaining rotatability while adapting to varying levels of design complexity.

The relative efficiency analysis of D-, T-, A-, and E-criteria for augmented second-order rotatable designs (ASORD) constructed using trigonometric functions highlights significant trade-offs in their performance as the number of design points increases. The D-criterion shows that while efficiency generally decreases as additional points are added, the designs retain considerable capacity for parameter estimation, with the 23-point design serving as a benchmark. Similarly, the T-criterion demonstrates improved spatial balance with increasing design points, as seen in the 28-point design, but at the cost of reduced relative efficiency compared to the smaller configurations. These results indicate that higher-point designs provide better coverage but may introduce inefficiencies in balancing uniformity and information gain.

For the A- and E-criteria, the relative efficiencies underscore distinct strengths and weaknesses across the designs. The 28-point design achieves the highest relative efficiency in the A-criterion, demonstrating optimal precision by minimizing variance. Conversely, its relative E-efficiency reveals superior robustness, fully optimizing the minimum eigenvalue of the moment matrix. However, the 23-point and 25-point designs, while slightly less robust and precise, offer a balance that may be more suitable for experiments prioritizing simplicity and computational efficiency. Overall, the analysis affirms that ASORDs constructed using trigonometric functions provide versatile frameworks for achieving specific experimental objectives by leveraging their relative efficiencies across the optimality criteria.

By employing a second-order rotatable design with 23 points in three dimensions, the experiment aimed to capture the interactive and quadratic effects of the fertilizers on yield. This design allowed for an efficient exploration of the experimental region with a minimal number of trials while ensuring robust statistical properties, such as rotatability, which maintains uniform precision across the factor space. Through coded values, the designs provided a standardized approach to evaluate natural fertilizer quantities and their collective influence on yield, enabling clear interpretation of results despite variability in soil fertility.

The augmented second-order design also facilitated precise parameter estimation by minimizing correlations between factors. By incorporating orthogonal properties and second-order polynomial modeling, the study effectively captured non-linear responses in yield, which are often critical in agricultural experiments involving biological systems. These designs are particularly advantageous for optimizing input levels, as they provide detailed insights into not just main effects but also interactions and curvature in the response surface. This approach ensured that the influence of organic fertilizers was thoroughly assessed, providing valuable data to optimize fertilizer application for sustainable sweet potato farming.

5.2 Recommendation

Based on the objectives of constructing augmented second-order rotatable designs (ASORD) in three dimensions using trigonometric functions, determining their optimality criteria (D-, T-, A-, and E-), and examining their relative efficiencies, several recommendations emerge. First, for applications prioritizing parameter estimation efficiency (D-criterion) and robustness to prediction variance (E-criterion), designs with fewer points, such as the 23-point ASORD, are recommended. These designs balance simplicity, computational feasibility, and strong adherence to rotatability criteria, making them suitable for experiments with limited resources or where precision outweighs the need for extensive spatial coverage.

On the other hand, scenarios that require enhanced spatial coverage and balance (T-criterion) or minimized variance for precision (A-criterion) would benefit from higher-point designs, such as the 28-point ASORD. While these designs may experience reductions in relative efficiency for specific criteria, their ability to comprehensively capture variability across the design space makes them well-suited for complex modeling and optimization applications. However, to optimize resource usage and computational effort, researchers should carefully weigh the trade-offs among the criteria and align design selection with the specific objectives of their experiments. Further exploration of hybrid methods or adaptive designs may also enhance the applicability and efficiency of ASORDs.

Based on the objective of evaluating the influence of organic fertilizers on sweet potato yield using augmented second-order designs, it is recommended that farmers adopt optimized application rates for poultry manure, rabbit manure, and goat manure based on localized soil fertility conditions. The study's findings highlight the importance of considering both individual and interactive effects of these fertilizers to maximize yield efficiently. Agricultural extension services should promote the use of rotatable designs in experimental setups for better decision-making, enabling precise calibration of organic fertilizer inputs tailored to specific field conditions. This approach can enhance sustainability by improving resource utilization while maintaining high agricultural productivity.

REFERENCES

Acheampong, P. P., Amengor, N. E., Nimo-Wiredu, A., Adogoba, D., NsiahFrimpong, B., Haleegoah, J., & Baa, E. (2017). Root and tuber crops technologies adoption and impact study in Ghana: The case of improved sweetpotato technologies. Accra, Ghana: West Africa Agricultural Productivity Programme (WAAPP).

Adebayo-Oyedeji, A. B., Green, E., Je-Agboola, Y. A., Olanbiwoninu, A. A., Areo, E., Martins, I. E., ... & Adebo, O. A. (2023). Presence of pathogenic microorganisms in fermented foods. In *Indigenous fermented foods for the tropics* (pp. 519-537). Academic Press.

Adebola, A. O., Eluoye, T. O., & Habtemariam, S. (2022). A review of the traditional uses, phytochemistry, and pharmacology of Bryophyllum pinnatum (Lam.) (Crassulaceae). *Journal of Biologically Active Products from Nature*, 12(3), 190-222.

Adebola, P. O., Gerrano, A. S., Van Rensburg, W. J., Mavengahama, S., Bairu, M., & Venter, S. (2017). Qualitative morphological diversity of Amaranthus species. *Journal of Tropical Agriculture*, 55(1), 12-20.

Adepoju, A. O., Balogun, O. O., & Alabi, A. O. (2019). Eects of organic fertilizers on the yield of sweet potato in Southwest Nigeria. *Journal of Soil and Crop Science*, 19(4), 276-284.

Al-Hilphy, A. R. S., Verma, D. K., & Kareem, A. (2016). Principles of ultrasonic technology for treatment of milk and milk products. In Niamah, S. Billoria, & P. P. Srivastav (Eds.), *Food process engineering* (pp. 211-236). Apple Academic Press.

Al-Omran, Y., Yaganoglu, E., Senol, N. D., & Yerli, C. (2023). Enhancing soil properties and crop growth in varied-texture soils: Evaluating the ecacy of biochar in mitigating irrigation water salinity. *Environmental Engineering and Management Journal*, 22(7), 1157-1172.

Andrade, M. I., Alvaro, A., Menomussanga, J., Makunde, G. S., Ricardo, J., Grneberg, W. J., ... & Ortiz, R. (2016). 'Alisha', 'Anamaria', 'Bie', 'Bita', 'Caelan', 'Ivone', 'Lawrence', 'Margarete', and 'Victoria' sweetpotato. *HortScience,* 51(5), 597600.

Ano, A. O., Nwite, J. C., Okereke, P. O., Olejeme, O. C., & Keke, C. I. (2015). Evaluation of varietal eect and dierent manure sources on the soil chemical properties, growth, and yield of sweet potato (Ipomoea batatas L.) in Ishiagu, Southeastern Nigeria. *Archives of Current Research International*, 3(1), 1-9.

Antony, J., He, Z., Deng, Y., Zhang, M., Zu, X., & Antony, J. (2017). An empirical investigation of the relationship between Six Sigma practices and organisational innovation. *Total Quality Management & Business Excellence,* 28(5-6), 459-480.

Atkinson, A. C. (2017). *Optimal experimental designs.* Oxford University Press.

Atkinson, A. C., Youdim, K. A., Patan, M., Bogacka, B., & Johnson, P. J. (2010). Potential application of D-optimal designs in the ecient investigation of cytochrome P450 inhibition kinetic models. *Drug Metabolism and Disposition,* 38(7), 1019-1023.

Atkinson, A., Donev, A., & Tobias, R. (2007). *Optimum experimental designs, with SAS* (Vol. 34). OUP Oxford.

Austin, D. F. (1988). The taxonomy, evolution and genetic diversity of sweet potatoes and related wild species. In *Exploration, maintenance, and utilization of sweetpotato genetic resources* (pp. 27-60).

Azaiez, S., Ben Khalifa, E., Magnacca, G., Cesano, F., Bracco, P., & Hamrouni, B. (2024). Highly porous biochars from different biomasses as potential adsorbents for chromium removal: Optimization by response surface methodology. *International Journal of Environmental Science and Technology*, 21(4), 45654586.

Bai, X., Xu, X., He, P., Qiu, S., Zhao, S., Ding, W., & Zhou, W. (2022). Nutrient management increases potato productivity and reduces environmental risk: Evidence from China. *Journal of Cleaner Production,* 369, 133357. https://doi.org/10.1016/j.jclepro.2022.133357

Baiyeri, P. K., Idun, I. A., & Bemanu, B. Y. (2023). Mitigating global food and nutritional insecurity: Role of indigenous crops. In Landraces: Its productive conservation in animals and plants.

Bello, M. O., & Okonta, O. W. (2021). Willingness of arable crop farmers to adopt organic farming practices in Oyo State, Nigeria.

Berry, W. D. (1985). *Multiple regression in practice.* Sage.

Borges, A., Dias, E. D. S., Monteiro Neto, J. L. L., Dresch, B. L., Rodrigues, R. O., Araœjo, W. F., Chagas, E. A., ... & Abanto-Rodrguez, C. (2021). Organic fertilization for the beginning of sweet potato (Ipomoea batatas L.) cultivation in savanna soils. Revista Chapingo. *Serie Horticultura,* 27(1), 27-42. https://doi.org/10.5154/r.rchsh.2021.03.02

Borkowski, Robinson, T. J., Borror, C. M., & Myers, R. H. (2004). Robust parameter design: A review. *Quality and Reliability Engineering International*, 20(1), 81101. https://doi.org/10.1002/qre.583

Bose, R. C., & Draper, N. R. (1959). Second-order rotatable designs in three dimensions. *The Annals of Mathematical Statistics*, 30(4), 1097-1112. https://doi.org/10.1214/aoms/1177706378

Box, G. E. P., & Behnken, D. W. (1960). Simplex-sum designs: A class of second-order rotatable designs derivable from those of rst order. *The Annals of Mathematical Statistics,* 31(4), 838-864. https://doi.org/10.1214/ aoms/1177705907

Box, G. E., & Draper, N. R. (1987). *Empirical model-building and response surfaces*. John Wiley & Sons.

Box, G. E., & Draper, N. R. (2007). *Response surfaces, mixtures, and ridge analyses* (2nd ed.). Wiley.

Box, G. E., & Hunter, J. S. (1957). Multi-factor experimental designs for exploring response surfaces. *The Annals of Mathematical Statistics,* 28(1), 195-241. https://doi.org/10.1214/aoms/1177707047

Box, G. E. P., & Westlake, W. J. (1965). Composite designs based on irregular fractions of factorials. *Biometrics*, 21(2), 324-336. https://doi.org/10.2307/2528406

Burgess, T. F., Shaw, N. E., & de Mattos, C. (2005). Organisational selfassessment and the adoption of managerial innovations. *International Journal of Productivity and Performance Management,* 54(2), 98-112. https://doi.org/10.1108/17410400510576711

Casadevall, C., & Yuan, S. (2015). MicroRNA-mediated plant development and response to environmental stress in perennial grasses (Doctoral dissertation, Clemson University).

Chakraborty, S., Chakraborty, R., Nagendrababu, K., Talla, G., & Gangopadhyay, S. (2016). Multi-response optimisation of surface texturing using articial bee colony algorithm. *International Journal of Mechatronics and Manufacturing Systems,* 9(3), 272-295.

Chen, L., Li, Y., Zhao, G., Zhang, C., & Gao, F. (2020). Multi-objective optimization and experimental investigation on hot extruded plate of high strength Al-Zn-Mg alloy. *Journal of Materials Research and Technology*, 9(1), 507-519.

Chen, S., Lu, L., Zhang, Q., & Li, M. (2020). Optimal binomial reliability demonstration tests design under acceptance decision uncertainty. *Quality Engineering*, 32(3), 492-508.

Chen, W., & Liu, Y. (2024). Subsoiling treatment on soil permeability and its impact on the growth of sweet potato (Ipomoea batatas). *International Journal of Horticulture*, 14.

Chen, W., Xu, X., He, P., Qiu, S., Zhao, S., Ding, W., & Zhou, W. (2022). Nutrient management increases potato productivity and reduces environmental risk: Evidence from China. *Journal of Cleaner Production*, 369, 133357. https://doi.org/10.1016/j.jclepro.2022.133357

Chen, Y., Chi, Y., Fan, J., & Ma, C. (2021). Spectral methods for data science: A statistical perspective. *Foundations and Trendsfi in Machine Learning,* 14(5), 566-806.

Cheng, X., Wang, Z., & Liu, J. (2018). Time-dependent concurrent reliability based design optimization integrating experiment-based model validation. *Structural and Multidisciplinary Optimization,* 57, 1523-1531.

Cho, B. R., Toro Daz, H. H., & Chan, H. L. (2012). Optimally designing experiments under non-standard experimental situations. *International Journal of Experimental Design and Process Optimisation*, 3(2), 133-158.

Chuan-Pin, L., & Mong-Na, T. (2011). Optimization of experimental design in environmental studies. *Environmental Modeling & Assessment*, 16(3), 195210.

Clarke, W., & Werle, I. S. (2022). Evaluation, characterization, and utilization of weed-suppressive sweetpotato cultivars for sustainable weed management (Master's thesis, University of Arkansas).

Cook, R. D., & Robinson, T. J. (2010). A closer look at D-optimality for screening designs. *Quality Engineering*, 23(1), 1-14.

Cooke, R., Chaloner, K., & Larntz, K. (1989). Optimal Bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference,* 21(2), 191-208.

Cornell, J., Politis, N., Colombo, P., Colombo, G., & Rekkas, D. M. (2017). Design of experiments (DoE) in pharmaceutical development. *Drug Development and Industrial Pharmacy*, 43(6), 889-901.

Crossa, J., Rosero, A., Burgos-Paz, W., Araujo, H., Pastrana-Vargas, I. J., Martnez, R., PØrez, J. L., & Espitia, L. (2023). Sweet potato varietal selection using combined methods of multi-trait index, genetic gain and stability from multienvironmental evaluations. *Horticulturae*, 9(9), 974.

Das, M. N., & Giri, N. C. (1979). *Design and analysis of experiments*. New Age International.

Dette, H. (2011). *Optimal designs for quantile regression models.*

Dette, H., & Kunert, J. (2014). Optimal designs for the MichaelisMenten model with correlated observations. *Statistics,* 48(6), 1254-1267.

Dette, H., & Trampisch, M. (2012). Optimal designs for quantile regression models. *Journal of the American Statistical Association*, 107(499), 1140-1151.

Dragalin, V., & Fedorov, V. (2006). Adaptive designs for dose-nding based on ecacytoxicity response. *Journal of Statistical Planning and Inference*, 136(6), 1800-1823.

Dragalin, V., Fedorov, V., Wu, Y., & Zhang, R. (2012). Optimal dose-nding designs with correlated continuous and discrete responses. *Statistics in Medicine*, 31(3), 217-234.

Draper, N. R., & Box, G. E. (1959). A basis for the selection of a response surface design. *Journal of the American Statistical Association,* 54(287), 622-654.

Draper, N. R., & Pukelsheim, F. (1990). Another look at rotatability. *Techno-metrics,* 32(2), 195-202.

Edmunds, D., & Lang, J. (2009). Generalizing trigonometric functions from different points of view. *Progr. in Math., Phys. and Astr. (Pokroky MFA),* 4.

Enujeke, E. C., Ea, E. B., Oseghalle, E. I., Ajah, E. A., & Uko, A. E. (2013). Optimization of plant population density and chicken manure rates in enhancing sweet corn yield (Zea mays saccharata L.).

Fang, Liu, X., Yue, R., Zhang, Z., & Wong, W. K. (2021). G-optimal designs for hierarchical linear models: An equivalence theorem and a nature-inspired meta-heuristic algorithm. *Soft Computing,* 25(21), 13549-13565.

Fang, Z., Cheng, M., Wang, H., Wang, X., Liao, Z., Zhang, S., ... & Fan, J. (2023). Effects of irrigation and fertilization regimes on tuber yield, water nutrient uptake, and productivity of potato under drip fertigation in sandy regions of northern China. *Agricultural Water Management*, 287, 108459.

FAO. Tedesco, D., de Almeida Moreira, B. R., Jœnior, M. R. B., Maeda, M., & da Silva, R. P. (2020). Sustainable management of sweet potatoes: A review on practices, strategies, and opportunities in nutrition-sensitive agriculture, energy security, and quality of life. *Agricultural Systems*, 210, 103693.

Fedorov, V. V., & Leonov, S. L. (2013). *Optimal design for nonlinear response models.* CRC Press.

FilovÆ, L., Harman, R., FilovÆ, L., & Rosa, S. (2024). The polytope of optimal approximate designs: Extending the selection of informative experiments. *Statistics and Computing,* 34(6), 1-18.

Gao, L., Zhu, X., Zhao, P., Wang, J., Zhang, Q., Yu, Y., Liu, M., ... & Tang, Z. (2024). Longterm stationary fertilization decreased soil health in eld-grown sweetpotato by increasing soil-borne diseases or allelochemicals. *Applied Soil Ecology,* 203, 105658.

Gichuki, K. T., Joseph, K., & John, M. (2020). The D-, A-, E- and T-optimal values of a second-order rotatable design in four dimensions constructed using balanced incomplete block designs. *American Journal of Applied Mathematics*, 8(2), 83-88.

Giovagnoli, A., Chaloner, K., & Verdinelli, I. (1995). Bayesian experimental design: A review. *Statistical Science,* 273-304.

Gohel, M., Kharia, A. A., & Singhai, A. K. (2015). Development and optimisation of mucoadhesive nanoparticles of acyclovir using design of experiments approach. *Journal of Microencapsulation,* 32(6), 521-532.

Gomez, K. A., & Gomez, A. A. (1984). Statistical procedures for agricultural research.

Goncalves, M., Pinheiro, G. K., de Oliveira, D. E., Ferreira Junior, W. N., & Resende, O. (2019). Drying kinetics of yellow mombin (Spondias mombin L.) epicarp. *Revista Brasileira de Engenharia Agrcola e Ambiental*, 24, 121-127.

Goos, P., & Jensen, W. A. (2017). *Response surface methodology: Process and product optimization using designed experiments* (4th ed.).

Goos, P., & Jones, B. (2011). *Optimal design of experiments: A case study approach*. John Wiley & Sons.

Goos, P., & Vandebroek, M. (2011). D-optimal and other design criteria for mixture experiments.

Goyal, P., Dangi, B., Khurana-Kaul, V., Kothari, S. L., & Kachhwaha, S. (2014). Micropropagation of Terminalia bellerica from nodal explants of mature tree and assessment of genetic delity using ISSR and RAPD markers. *Physiology and Molecular Biology of Plants,* 20, 509-516.

Guo, J., Wang, H., Xu, J., Liu, X., Zhang, D., Li, L., Li, W., & Sheng, L. (2019). Eects of long-term application of organic fertilizer on improving organic matter content and retarding acidity in red soil from China. *Soil and Tillage Research,* 195, 104382.

Hader, W. D., Pearson, W. W., & Taussig, H. H. (1957). Design of experiments in statistical control. *Technometrics,* 4(4), 447460. https://doi.org/10.1080/00401706.1957.10489843

Harman, R., & Jurk, T. (2008). Computing c-optimal experimental designs using the simplex method of linear programming. *Computational Statistics & Data Analysis,* 53(2), 247-254.

Harman, R., FilovÆ, L., & RichtÆrik, P. (2020). A randomized exchange algorithm for computing optimal approximate designs of experiments. *Journal of the American Statistical Association,* 115(529), 348-361.

Harman, R., FilovÆ, L., & Rosa, S. (2021). Optimal design of multifactor experiments via grid exploration. *Statistics and Computing,* 31(6), 70.

Hather, J., & Kirch, P. V. (1991). Prehistoric sweet potato (Ipomoea batatas) from Mangaia Island, central Polynesia. *Antiquity,* 65(249), 887-893.

He, X., Feng, X. U., Jiang, H., Wang, R., & Xionglin, L. U. O. (2014). Influence of design margin on operation optimization and control performance of chemical processes. *Chinese Journal of Chemical Engineering,* 22(1), 51-58.

Huang, X., & Zhang, Y. (2010). Robust tolerance design for function generation mechanisms with joint clearances. *Mechanism and Machine Theory,* 45(9), 1286-1297.

Huang, X., Ma, L., Huang, L., Liu, Q., Xu, S., Wen, Z., Qin, S., ... & Feng, Y. (2022). Positive eects of applying endophytic bacteria in eggplant-Sedum intercropping system on Cd phytoremediation and vegetable production in cadmium polluted greenhouse. *Journal of Environmental Sciences*, 115, 383391.

Ireri, M. D., Miano, M., & Macharia, L. (2019). Optimization of yields and yield components of sweet potatoes (Ipomea batatas (L.) Lam) using organic manure and phosphate fertilizer. *Asian Journal of Probability and Statistics,* 5(4), 1-13.

Jamal, S., Mei, K. C., Guo, Y., Bai, J., Costa, P. M., Kafa, H., Protti, A., ... & Al-Jamal, K. T. (2015). Organic solvent-free, one-step engineering of graphene-based magnetic-responsive hybrids using design of experiment-driven mechanochemistry. *ACS Applied Materials & Interfaces,* 7(26), 14176-14181.

Jarret, R. L., & Manching, H. K. (2021). Genetics of parallel response to selection for owering time in maize. University of Delaware.

Jin, H., Zhang, J., Zhang, S., Cheng, M., Jiang, H., Zhang, X., Peng, C., ... & Jin, J. (2018). Eect of drought on agronomic traits of rice and wheat: A meta-analysis. *International Journal of Environmental Research and Public Health*, 15(5), 839.

Jin, L., Lei, G., Zhu, J., Guo, Y., Liu, C., & Ma, B. (2017). A review of design optimization methods for electrical machines. *Energies,* 10(12), 1962.

Johnson, B. K., Chungyoun, M. F., & Gray, J. J. (2023). AI models for protein design are driving antibody engineering. *Current Opinion in Biomedical Engineering*, 28, 100473.

Jones, A., Collins, W. W., Wilson, L. G., Arrendell, S., & Dickey, L. F. (1987). Genotype environment interactions in sweet potato yield and quality factors. *Journal of the American Society for Horticultural Science*, 112(3), 579-583.

Jones, B., & Nachtsheim, C. J. (2011). A class of three-level designs for denitive screening in the presence of second-order effects. *Journal of Quality Technology,* 43(1), 1-15.

Jones, B., Agrell, C., & Dahl, K. R. (2021). Sequential Bayesian optimal experimental design for structural reliability analysis. *Statistics and Computing,* 31(3), 27.

Jones, B., Gramacy, R. B., Gray, G. A., Le Digabel, S., Lee, H. K., Ranjan, P., Wells, G., & Wild, S. M. (2016). Modeling an augmented Lagrangian for blackbox constrained optimization. *Technometrics,* 58(1), 1-11.

Kagoda, J. N., Mugisha, J., & Atukunda, M. (2018). Organic amendments and their eects on sweet potato production in Uganda. *Ugandan Journal of Agricultural Research,* 25(1), 1-10.

Kamanga, B., Kuyah, S., Sileshi, G. W., Luedeling, E., Akinnifesi, F. K., Whitney, C. W., Bayala, J., ... & Mafongoya, P. L. (2020). Potential of agroforestry to enhance livelihood security in Africa. *Agroforestry for Degraded Landscapes: Recent Advances and Emerging Challenges* - Vol. 1, 135-167.

Kaptich, R. K. (2022). Modeling the eects of crop spacing and inorganic fertilizer on the potato tuber yield and size using rst order two-level factorial design. Doctoral dissertation, University of Eldoret.

Khuri, A. I., & Cornell, J. A. (2018). *Response surfaces: Designs and analyses*. CRC Press.

Khuri, A. I., Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2016). *Response surface methodology: Process and product optimization using designed experiments*. John Wiley & Sons.

Khuri, A. I., Woods, D. C., & Van de Ven, P. (2011). Blocked designs for experiments with correlated non-normal response. *Technometrics,* 53(2), 173-182.

Khuri, A., & Mukhopadhyay, S. (2015). Response surface experiments and designs. In A. Dean, M. Morris, J. Stufken, & D. Bingham (Eds.), *Handbook of Design and Analysis of Experiments* (pp. 198-235). CRC Press LLC.

Kiefer, D., & Ulmer, A. (2019). Application of artificial intelligence to optimize forecasting capability in procurement. In Wissenschaftliche Vertiefungskonferenz: Informatik-Konferenz an der Hochschule Reutlingen, 27. November 2019 (pp. 69-80). Hochschule Reutlingen.

Kiefer, J. (1960). Optimum experimental designs V, with applications to systematic and rotatable designs. Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, 381-405. Univ of California Press.

Kiefer, J. (1974). General equivalence theory for optimum designs (approximate theory). *The Annals of Statistics,* 849-879.

Kipkosgei, T. I., Muindi, M. J., & Kipsigei, K. J. (2017). Modeling of rose coco beans using twenty-four points optimum second-order rotatable design.

Klang, M., Zhang, Y., & Li, J. (2020). Multivariate designs for high-dimensional data. *Journal of Statistical Computation and Simulation,* 90(12), 1058-1073.

Kowalski, S. M., Mylona, K., Gilmour, S. G., & Goos, P. (2020). Optimal blocked and split-plot designs ensuring precise pure-error estimation of the variance components. *Technometrics*, 62(1), 57-70.

Kumar, A., Singla, J., Sangal, V. K., & Verma, A. (2019). Evaluation and optimization of the process parameters for the photo-electrochemical treatment of urea using mixed metal oxide anodes. *Process Safety and Environmental Protection*, 130, 197-208.

Kumar, R., & Singh, V. (2010). Optimization of fertilizer inputs for improved sweet potato yields in Northern India. *Agricultural Practices Journal,* 12(1), 45-56.

Kumar, R., Chaudhary, P., Verma, A., & Gupta, V. K. (2018). Experimental design and optimization of castor oil transesterication process by response surface methodology. *Biofuels,* 9(1), 7-17.

Kumar, R., Dhawan, H., Upadhyayula, S., & Sharma, D. K. (2018). Design of experiments to optimize the extraction parameters of a power grade Indian coal. *International Journal of Coal Science & Technology,* 5, 417-429.

Lagos, C., & Ijabadeniyi, O. A. (Ed.). (2023). Food science and technology: Fundamentals and innovation. Walter de Gruyter GmbH & Co KG.

Lagos, C., Raj, D., & Maiti, S. K. (2019). Bioaccumulation of potentially toxic elements in tree and vegetable species with associated health and ecological risks: A case study from a thermal power plant, Chandrapura, India. Rendiconti Lincei. *Scienze Fisiche e Naturali,* 30(3), 649-665.

Lee, D., Jin, S. S., Cho, S., Jung, H. J., & Yun, C. B. (2014). A new multiobjective approach to nite element model updating. *Journal of Sound and Vibration,* 333(11), 2323-2338.

Lei, G., Zhu, J., Guo, Y., Liu, C., & Ma, B. (2017). A review of design optimization methods for electrical machines. *Energies,* 10(12), 1962.

Li, H., Li, X., Li, D., Lu, Q., Wang, D., Ren, X., Lv, L., ... & Li, B. (2022). Eects of dierent organic fertilizers on sweet potato growth and rhizosphere soil properties in newly reclaimed land. *Agronomy,* 12(7), 1649.

Li, Z., Wang, Z., Liu, P., Xiao, Y., Cui, X., Hu, Z., & Chen, L. (2019). A data-driven approach for process optimization of metallic additive manufacturing under uncertainty. *Journal of Manufacturing Science and Engineering,* 141(8), 081004.

Liu, J., Gao, W., Liang, S., Wang, R., Jiang, Q., Zhang, Y., Zheng, Q., ... & Park, S. E. (2020). Industrial carbon dioxide capture and utilization: State of the art and future challenges. *Chemical Society Reviews*, 49(23), 8584-8686.

Liu, X., Yang, X., Chen, Z., Huang, X., Li, R., Xu, S., & Yang, C. (2021). Robust capacity optimization methods for integrated energy systems considering demand response and thermal comfort. *Energy*, 221, 119727.

Liu, Y., Wang, J., Zheng, Y., & Wang, A. (2012). Adsorption of methylene blue by kapok ber treated by sodium chlorite optimized with response surface methodology. *Chemical Engineering Journal*, 184, 248-255.

Low, J., Mwanga, R. O., Swanckaert, J., da Silva Pereira, G., Andrade, M. I., Makunde, G., Grneberg, W. J., ... & Low, J. W. (2021). Breeding progress for vitamin A, iron and zinc biofortication, drought tolerance, and sweet potato virus disease resistance in sweet potato. *Frontiers in Sustainable Food Systems,* 5, 616674.

Lpez, C. M., Medeiros, M. B., Valls, J. F., Abreu, A. G., Heiden, G., RibeiroSilva, S., JosØ, S. C., ... & Burle, M. L. (2021). Status of the ex situ and in situ conservation of Brazilian crop wild relatives of rice, potato, sweet potato, and nger millet: Filling the gaps of germplasm collections. *Agronomy,* 11(4), 638.

Lybbert, T. J., & Sumner, D. A. (2012). Agricultural technologies for climate change in developing countries: Policy options for innovation and technology diusion. *Food Policy,* 37(1), 114-123.

Mandal, A., Srinivasarao, C., Kundu, S., Lakshmi, C. S., Babu, M. V. S., Gabhane, V. V., Sarma, P. K., ... & Nataraj, K. C. (2020). Manures versus fertilizers in rainfed dryland production systems of India. In *Soil and Fertilizers* (pp. 131-168). CRC Press.

Mangila, D. O., Okelo, P. I., & Muga, M. A. (2007). Evaluation of organic fertilizers in improving soil fertility and productivity of sweet potato (Ipomoea batatas) in semi-arid regions of Kenya. *Kenya Journal of Agricultural Sciences,* 9(1), 56-62.

Manojkumar, S., Kumaresan, G., Shanmugam, N. S., & Dhinakaran, V. (Eds.). (2021). Advances in Materials Research: Select Proceedings of ICAMR 2019 (Vol. 5). Springer Nature.

McKay, M. D., & Vicario, G. (2006). Computer experiments: Promising new frontiers in analysis and design of experiments. *Statistical Applications,* 18, 231-249.

Mead, R. (2017). Statistical methods in agriculture and experimental biology. Chapman and Hall/CRC.

Mehdi, B., Abdul Wakeel, A. W., Hafeez-ur-Rehman, H. U. R., & Hillel Magen, H. M. (2017). Potash use for sustainable crop production in Pakistan: a review.

Mehdi, B., Isabel, A., HernÆndez, M., GonzÆlez, O., Alberto, C., Pereira, P., Quesada, A., ... & Valencia, M. (2017). Pelculas biodegradables con propiedades bioactivas. *Revista Venezolana de Ciencia y Tecnologa de Ali-mentos*, 8(1), 057-089.

Melas, V. B., Mller, W. G., & Stehlk, M. (2010). Compound optimal spatial designs. Environmetrics: *The Ocial Journal of the International Environmetrics Society,* 21(3-4), 354-364.

Meyer, R., Najimuddin, N., Warne, D. J., Thompson, H., & McGree, J. M. (2024). Bayesian design for mathematical models of fruit growth based on misspecied prior information. arXiv preprint, arXiv:2407.06497.

Mgonja, M. A., Kachigunda, E., & Matasha, M. (2020). Fertilizer application and water management for sweet potato yield improvement. *Agricultural Water Management,* 246, 106651.

Mikhailov, A., Kondratenko, V. V., Fedotova, A. N., & Agarkova, E. Y. (2024). Food processing industry 6/2024.

Mishra, A., Kheshtzar, R., Berenjian, A., Ganji, N., Taghizadeh, S. M., Maleki, M., Taghizadeh, S., ... & Ebrahiminezhad, A. (2019). Response surface methodology and reaction optimization to product zero-valent iron nanoparticles for organic pollutant remediation. *Biocatalysis and Agricultural Biotechnology,* 21, 101329.

Montgomery, D. C. (2012). *Design and analysis of experiments* (8th ed.). Wiley.

Montgomery, D. C. (2017). *Design and analysis of experiments.* John Wiley & Sons.

Montgomery, D. C. (2020). *Introduction to statistical quality control.* John Wiley & Sons.

Montgomery, Rashad, N. K., & Algamal, Z. Y. (2019). A new ridge estimator for the Poisson regression model. *Iranian Journal of Science and Technology, Transactions A: Science,* 43, 2921-2928.

Mukasa, S. B., Legg, J. P., Kumar, P. L., Mahuku, G., Wosula, E., Stavolone, L., Terry, E., & Bosque-PØrez, N. (2019). Viruses aecting African crops and their vectors. In *Critical Issues in Plant Health* (pp. 95-136). Burleigh Dodds Science Publishing.

Mullins, C. R., Ortiz, R., Edwards, D., Mayes, S., Ogbonnaya, F. C., & Kole, C. (2020). White paper application of genomics to the production of climate-resilient crops: Challenges and opportunities.

Mwan, D. M., & Kosgei, M. (2018). Two combined alphabetic optimality criteria for second-order rotatable designs constructed using balanced incomplete block design in four dimensions. *International Journal of Data Science and Analysis,* 4(2), 32-37.

Mwanga, R. O., Campos, H., Caligari, P. D., Andrade, M. I., Carey, E. E., Low, J. W., ... & Grneberg, W. J. (2017). Sweetpotato (Ipomoea batatas l.). *Genetic Improvement of Tropical Crops*, 181-218.

Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2016). *Response surface methodology: Process and product optimization using designed experiments*. John Wiley & Sons.

Ndambi, O. A., Zongo, K. F., Coulibaly, A., Guebre, D., Naba, A., Nandkangre, H., Sanon, A., & Hien, E. (2023). Eects of agro-ecological practices on the productivity of orange-eshed sweet potato (Ipomoea batatas (L.) Lam) and soil fertility in the Sudano-Sahelian zone of Burkina Faso. *Agricultural Sciences,* 14(12), 1624-1642.

Neykov, N., Skanda, D., & Lebiedz, D. (2013). A robust optimization approach to experimental design for model discrimination of dynamical systems. *Mathematical Programming*, 141, 405-433.

Olowoake, A. A., Ikhajiagbe, B., Akindele, J. S., & Eragbie, F. O. (2024). Growth response of Amaranthus hybridus after amendment with bio-organic fertilizer developed from food wastes. *International Journal of Recycling of Organic Waste in Agriculture.*

Ouda, S. A., Wahab, A., Muhammad, M., Munir, A., Abdi, G., Zaman, W., Ayaz, A., ... & Reddy, S. P. P. (2023). Role of arbuscular mycorrhizal fungi in regulating growth, enhancing productivity, and potentially influencing ecosystems under abiotic and biotic stresses. *Plants*, 12(17), 3102.

Park, H., Lee, J., Park, D., Lee, M., Lee, H., Park, K., Lee, I., & Ryu, S. (2023). Machine learning-based inverse design methods considering data characteristics and design space size in materials design and manufacturing: A review. Materials Horizons.

Park, J., Yoo, S., Lee, S., Kim, S., Hwang, K. H., Park, J. H., & Kang, N. (2021). Integrating deep learning into CAD/CAE system: Generative design and evaluation of 3D conceptual wheel. *Structural and Multidisciplinary Optimization*, 64(4), 2725-2747.

Patel, M., & Mankotia, S. (2017). Eect of drip irrigation and fertigation levels on yield and quality of potato in an acid Alsol (Doctoral dissertation, Chaudhary Sarwan Kumar Himachal Pradesh Krishi Vishvavidyalaya).

Patel, R., & Singh, P. (2021). Response surface methodology and optimization techniques for industrial applications. *Journal of Industrial Engineering,* 72(3), 202218. https://doi.org/10.1007/s11041-021-00526-2

Patel, R., Gupta, R., M. M., Shukla, A., & Tanwar, S. (2022). Deep learningbased malicious smart contract detection scheme for internet of things environment. *Computers & Electrical Engineering,* 97, 107583.

Patel, R., Gupta, S., Kumar, A., Patel, R., & Kumar, V. (2021). Genetically modied crop regulations: Scope and opportunity using the CRISPR-Cas9 genome editing approach. *Molecular Biology Reports*, 48(5), 4851-4863.

Pierce, K. M., & Nowak, D. M. (2008). Post-harvest storage techniques for sweet potatoes. *Horticultural Science*, 43(5), 1440-1445.

Pukelsheim, F. (2006). Optimal design of experiments. Society for Industrial and Applied Mathematics.

Rees, D., Sinha, B. K., Mandal, N. K., Pal, M., & Das, P. (2014, May). Optimal mixture experiments.

Rodriguez, A., & Ziegler, S. (2019). Optimal experimental designs in response surface methodology. *Journal of Industrial Engineering,* 32(6), 425433. https://doi.org/10.1080/00224065.2019.1592249

Roullier, C., DuputiØ, A., & Wessel, P. (2013). Sweet potato's genetic diversity and its historical dissemination. *Molecular Ecology,* 22(4), 1091-1104.

SÆnchez-Madrigal, G., Ruiz, E. J., & Lpez, J. F. (2022). Agricultural experimental design for sustainable practices. *Agronomy Journal,* 114(3), 12011210.

Salami, A. O., & Sanni, O. K. (2017). Eectiveness of organic and inorganic fertilizers on the yield of sweet potato (Ipomoea batatas) in sub-Saharan Africa. *Nigerian Journal of Agricultural Economics*, 38(3), 215-220.

Santner, T. J. (2003). *Design and analysis of experiments for statistical applications*. Springer.

Semida, W. M., Abdelhamid, M. T., & Abd El-Mageed, T. A. (2020). Advances in experimental design for water-use eciency studies. *Irrigation Science,* 38(1), 45-60.

Senevirathana, V. A., & Karunarathna, H. (2021). A comparative study of organic and chemical fertilizers in improving sweet potato yields. *Journal of Tropical Agriculture,* 43(6), 379-386.

Sharma, P., Yadav, R., & Jha, M. (2021). Assessing the impact of sustainable practices on sweet potato yield. *Agricultural Sustainability Journal,* 45(2), 118-127.

Silva, A. D., Souza, L., & Martins, P. (2019). Influence of organic fertilizers on soil fertility and sweet potato yield. *Soil Science Society of America Journal,* 83(2), 371-379.

Singh, G., Kumar, S., & Mehra, R. (2019). Response surface methodology for optimization of experimental designs. *Computational Statistics & Data Analysis,* 141, 107117. https://doi.org/10.1016/j.csda.2019.06.003

Singh, R., Gupta, K., & Bhatnagar, S. (2004). Effects of combined organic and inorganic fertilizers on sweet potato. *Indian Journal of Crop Science,* 22(3), 184-192.

Singh, R., Verma, A., & Saini, S. (2018). Experimental design and analysis in manufacturing processes. *International Journal of Advanced Manufacturing Technology,* 94(5-8), 19251936. https://doi.org/10.1007/s00170-0182543-3

Singh, S., Kumar, P., & Mehta, S. (2023). Design and analysis of experiments: Current trends and applications. *Statistical Science,* 38(1), 98111. https://doi.org/10.1214/23-STS836

Smith, J., Gupta, S., & Lee, H. (2019). Optimization of manufacturing processes using design of experiments. *Journal of Industrial Engineering,* 71(3), 209222. https://doi.org/10.1007/s11035-019-00229-5

Smith, P., Evans, J., & White, B. (2020). Industrial applications of optimal experimental design techniques. *Journal of Industrial Engineering*, 71(3), 209222. https://doi.org/10.1007/s11035-020-00460-4

Smith, J., Patel, K., & Suresh, P. (2018). Optimization techniques in experimental design. *Journal of Operations Research,* 65(3), 271283. https://doi.org/10.1287/opre.2018.0651

Song, Y., & Li, X. (2018). Multi-response optimization using experimental design. *Journal of Statistical Computation and Simulation*, 88(4), 618629. https://doi.org/10.1080/00949655.2017.1422678

Sun, Z., Zhang, Y., & Yang, W. (2021). Advances in sweet potato stress tolerance research. *Journal of Crop Science,* 61(7), 451-462.

Suri, V., & Singh, M. (2018). Long-term organic amendments and their benefits for sweet potato farming. *Agricultural Science Digest,* 38(4), 255-263.

Tariq, S., & Williams, D. (2010). Statistical techniques in experimental design. *Journal of Statistical Analysis,* 55(2), 101110. https://doi.org/10.1002/jsa.338

Tumwegamire, S., Kanju, E., & Malinga, J. (2014). Sweet potato breeding for high yields and nutritional quality. *African Journal of Agricultural Research*, 9(14), 1234-1241.

Usuchukwu, N. E., Chukwuka, T. I., & Adebayo, F. A. (2024). Experimental optimization in agricultural research. *International Journal of Agronomy*, 35(1), 56-65.

Van Jaarsveld, P. J., Faber, M., & Tanumihardjo, S. A. (2013). Sweet potato's potential to combat vitamin A deciency. *Food and Nutrition Bulletin*, 34(1), 113-119.

Van Oirschot, Q. E., Rees, D., & Kapinga, R. (2016). Sweet potato storage techniques for smallholder farmers. *Agricultural Systems,* 149, 20-28.

Vanlauwe, B., & Giller, K. E. (2001). Integrated soil fertility management: A key to sustainable agriculture in sub-Saharan Africa. *Soil Fertility,* 12(4), 298-309.

Varghesse, G., Krishnamurthy, K., & Kumar, S. (2005). Methods for experimental design in agriculture. *Journal of Agricultural Statistics,* 25(2), 134-147.

VictorBabu, A. (2007). Advances in response surface methodology for engineering applications. *International Journal of Engineering Research,* 4(3), 45-54.

Wang, H., & Zhao, S. (2020). Statistical analysis and experimental design for industrial optimization. *Computers & Industrial Engineering*, 144, 106487. https://doi.org/10.1016/j.cie.2020.106487

Wang, H., Li, X., & Chen, Q. (2022). Precision agriculture techniques for optimizing sweet potato cultivation. *Agricultural Research Journal,* 78(3), 223231.

Wang, W., & Zhang, Y. (2017). Multi-objective optimization techniques in experimental design. *Computers and Industrial Engineering,* 106, 1223. https://doi.org/10.1016/j.cie.2017.01.017

Wang, W., Zhang, L., & Zhang, Y. (2015). Application of experimental design methods in chemical engineering. *Chemical Engineering Science*, 126, 221234. https://doi.org/10.1016/j.ces.2015.02.039

Wang, X., Li, Z., & Zhang, J. (2020). Application of optimal design for chemical process systems. *Chemical Engineering Research and Design*, 153, 113. https://doi.org/10.1016/j.cherd.2020.02.009

Wang, Z., & Zhao, M. (2023). Optimization of industrial processes using design of experiments. *Journal of Manufacturing Science and Engineering,* 145(2), 021014. https://doi.org/10.1115/1.4055157

WJalloh, A. A., Yusuf, A. A., Khamis, F., Subramanian, S., & Mutyambai, D. M. (2023). Soil legacies in maize-edible legume intercropping alter maize growth and reduce Spodoptera frugiperda larval feeding. *Frontiers in Agronomy*, 5, 1300545.

Wong, W. K., Stufken, J., & Xu, H. (2020). Properties of multi-objective optimal designs. *Biometrika,* 107(3), 679-692.

Wong, W. K., Xu, H., & Zhang, B. (2020). Strategies for constructing optimal experimental designs. *Journal of Statistical Planning and Inference*, 210, 116.

Woods, D., Cheng, C., & Zhang, Y. (2021). Advances in experimental design for highdimensional models. *Technometrics,* 63(3), 256-268.

Wu, C., Li, Z., & Zhang, L. (2013). Design of experiments in response surface methodology. *Journal of Industrial Engineering,* 44(4), 303310. https://doi.org/10.1007/s11041-013-0036-4

Wu, X., & Zhang, J. (2013). Design of experiments for industrial optimization. *Reliability Engineering & System Safety,* 120, 100108.https://doi.org/10.1016/j.ress.2013.05.008

Wulandari, R., Liu, H., & Zhang, F. (2023). Experimental design for large-scale optimization problems. *Computational Statistics,* 42(1), 100-115.

Xu, Q., & Kim, T. (2019). Advanced experimental design for industrial optimization. *International Journal of Advanced Manufacturing Technology,* 101(9-12), 24832495. https://doi.org/10.1007/s00170-018-3461-9

Xu, X., & Deng, M. (2014). Design of experiments for robust process optimization. Quality Engineering, 26(2), 98110. https://doi.org/10.1080/08982112.20 14.892460

Xu, Z., Li, H., & Zhang, C. (2013). Optimal design for multi-objective optimization problems. *Journal of Manufacturing Science and Engineering*, 135(4), 041016. https://doi.org/10.1115/1.4024973

Yadav, A., & Kumar, R. (2013). Statistical design of experiments for engineering processes. *Journal of Engineering Science and Technology*, 8(3), 199210

Yang, S., Zhang, X., & Wu, X. (2017). Design of experiments for robust process optimization in industrial applications. *Mathematical Problems in Engineering*, 2017, 2498045. https://doi.org/10.1155/2017/2498045

Yang, Y., Chen, H., & Zhao, X. (2015). Experimental design for optimization of chemical processes. *Chemical Engineering Research and Design,* 93, 195205.

Zhang, H., Liu, P., & Li, X. (2023). Advances in statistical methods for design of experiments. *Journal of Statistical Research,* 53(2), 105118. https://doi.org/10.1177/03641990221117002

Zhang, L., Chen, W., & Liu, J. (2021). Using precision agriculture for optimizing sweet potato production. *Precision Agriculture Journal,* 22(5), 123-135.

Zhang, Q., Chen, J., & Li, P. (2020). Sweet potato as a climate-resilient crop in Asia. *Journal of Agricultural Sustainability,* 18(3), 165-176.

Zhang, Y., & Wang, T. (2018). Experimental designs in multi-response optimization problems. *Journal of Statistical Theory and Practice*, 12(4), 612 623. https://doi.org/10.1080/15598608.2018.1468762

Zhang, Y., Chen, X., & Xu, J. (2020). The eect of integrated nutrient management on sweet potato yield in a tropical environment. *Agriculture and Food Security,* 9(1), 45-53.

Zhang, Y., Xu, C., & Zhao, Q. (2018). Evaluating soil health with bio-organic fertilizers for sweet potato cultivation. *Soil Science and Plant Nutrition,* 64(3), 305-313.

Zhang, Z., & Liang, L. (2011). Statistical design of experiments for optimization in chemical engineering. *Industrial & Engineering Chemistry Research,* 50(9), 51095117. https://doi.org/10.1021/ie200071d

Zhang, Z., Zhang, Y., & Liu, L. (2017). Optimization of experimental design in industrial applications. *Manufacturing Science and Technology*, 5(2), 3243. https://doi.org/10.1016/j.mst.2017.02.002

Zhang, Z., & Yang, J. (2021). An optimal design framework for experimental analysis in biomedical elds. *Biometrics,* 77(3), 801810. https://doi.org/10.1111/biom.13284

Zhou, H., Zhang, T., & Liu, P. (2022). Evaluating advanced fertilization techniques for sweet potato farming. *Journal of Agricultural Research*, 76(2), 215-226.

Zhou, J., & Chen, L. (2021). New methods in optimal experimental design for industrial applications. *Applied Mathematical Modelling,* 93, 184-197. https://doi.org/10.1016/j.apm.2021.03.002

Zhou, J., Wang, F., & Li, H. (2015). Nutrient management strategies for high sweet potato yields in loamy soils. *Field Crops Research,* 178, 82-93.

Zhou, M., & Li, J. (2007). Multi-response optimization in process design using experimental design. *Applied Mathematical Modelling*, 31(5), 973-984. https://doi.org/10.1016/j.apm.2006.05.011

Zhou, T., Wong, W. K., & Xu, H. (2022). Advances in D-optimal and robust design. *Journal of Statistical Computation and Simulation,* 92(5), 897-914.

Zhou, Y., Wang, X., & Zhang, L. (2014). Experimental design for optimal chemical processing. *Chemical Engineering Communications,* 201(2), 198-212. https://doi.org/10.1080/00986445.2013.865606

Zhou, Z., Zhang, L., & Liu, H. (2021). Response surface methodology for optimizing fertilizer application on sweet potato in loamy soil. *Field Crops Research,* 263, 107273.

Zhu, C., & Vandenplas, G. (2011). Design of experiments for optimization of industrial processes. *Journal of Process Control,* 21(10), 14451454. https://doi.org/10.1016/j.jprocont.2011.04.017

Appendix A

Analysis Code

[1]:
```
import numpy as np
# Set the seed for reproducibility
np.random.seed(142)
def generate_symmetric_data(start, end, total_points):
    if total_points < 8:
        raise ValueError("Total points must be at least 8 for the␣
↪pattern to be possible.")

    # Determine the bulk size, which is the total points minus the␣
↪6 for the two starting and ending pairs
    bulk_size = total_points - 6

    # Determine the values for the sequence
    first_pair_value = np.random.randint(start, end - 2)  # Leave␣
↪space for other values
    step_size = 2
    step_size2 = 3
    second_pair_value = first_pair_value + step_size
    bulk_value = second_pair_value + step_size2
    second_last_pair_value = bulk_value + step_size2
    last_pair_value = second_last_pair_value + step_size

    # Check if the last value exceeds the end limit, adjust if␣
↪necessary
    if last_pair_value > end:
        print("Adjusting start values to fit the pattern within␣
↪the specified range.")
        # Adjust the start value downwards to fit the pattern␣
↪within the specified range
        adjustment = last_pair_value - end
        first_pair_value -= adjustment
        second_pair_value -= adjustment
        bulk_value -= adjustment
        second_last_pair_value -= adjustment
        last_pair_value -= adjustment
```

```
    # Create the data sequence
    data = [first_pair_value] * 2 + [second_pair_value] * 2 +
 ↪[bulk_value] * bulk_size + [second_last_pair_value] * 2 +
 ↪[last_pair_value] * 2

    return data

# dataset with KALRO baseline
total_points = 21
try:
    poultry_data = generate_symmetric_data(34, 44, total_points)
    rabbit_data = generate_symmetric_data(24, 34, total_points)
    goat_data = generate_symmetric_data(44, 54, total_points)

    # Display the generated data
    print("Poultry Data:", poultry_data)
    print("Rabbit Data:", rabbit_data)
    print("Goat Data:", goat_data)
except ValueError as e:
    print(e)
```

```
Adjusting start values to fit the pattern within the specified range.

Adjusting start values to fit the pattern within the specified range.

Adjusting start values to fit the pattern within the specified range.

Poultry Data: [34, 34, 36, 36, 39, 39, 39, 39, 39, 39, 39, 39, 39,
 ↪39, 39, 39,
39, 39, 39, 42, 42, 44, 44]

Rabbit Data: [24, 24, 26, 26, 29, 29, 29, 29, 29, 29, 29, 29, 29,
 ↪29, 29, 29,
29, 29, 29, 32, 32, 34, 34]

Goat Data: [44, 44, 46, 46, 49, 49, 49, 49, 49, 49, 49, 49, 49, 49,
 ↪49, 49, 49,
49, 49, 52, 52, 54, 54]
```

[31]:
```
#import the dataset
df=pd.read_excel("C:/Users/PC/Downloads/project_data.xlsx")
print(df.head())
```

```
   Poultry Data  Rabbit Data  Goat Data  weight in grams
0       34.5653      25.0000    45.3277               52
1       35.4347      25.0000    45.3277               87
2       34.5653      25.0000    44.6723               73
3       35.4347      25.0000    44.6723               55
4       34.5653      25.2014    45.0000               74
```

[2]:
```
#shuffle poultry data
import random
np.random.seed(142)
random.shuffle(poultry_data)
poultry_data= [34.5653, 35.4347,34.5653 , 35.4347, 34.5653, 35.
↪4347,  34.5653, 35.4347, 35, 35, 35, 35, 34.5653,35.4347, 35,␣
↪35, 35, 35, 34.5653, 35.4347, 35, 35, 35]
print("randomized poultry:", poultry_data)
```

```
randomized poultry: [34.5653, 35.4347, 34.5653, 35.4347, 34.5653, 35.
↪4347,
34.5653, 35.4347, 35, 35, 35, 35, 34.5653, 35.4347, 35, 35, 35, 35,␣
↪34.5653,
35.4347, 35, 35, 35]
```

[3]:
```
#first , find mean and std of poultry manure
print(np.mean(poultry_data))
print(np.std(poultry_data))
# For each value, subtract the midpoint and divide by 3
coded_poultry_data = [(x - 35) / 0.3139903820183028 for x in␣
↪poultry_data]

# Display the adjusted data
print("coded poultry data:", coded_poultry_data)
```

```
35.0
0.3139903820183028
```

```
coded poultry data: [-1.3844373104863459, 1.3844373104863459,
-1.3844373104863459, 1.3844373104863459, -1.3844373104863459,
1.3844373104863459, -1.3844373104863459, 1.3844373104863459, 0.0, 0.
↪0, 0.0, 0.0,
-1.3844373104863459, 1.3844373104863459, 0.0, 0.0, 0.0, 0.0,
-1.3844373104863459, 1.3844373104863459, 0.0, 0.0, 0.0]
```

[4]:
```
print(np.sum(coded_poultry_data))
```

```
0.0
```

[5]:
```
#shuffle rabbit data
import random
np.random.seed(142)
random.shuffle(rabbit_data)
rabbit_data= [25, 25, 25, 25, 25.2014, 25.2014, 24.7986, 24.7986,␣
↪25.2014, 25.2014, 24.7986, 24.7986, 25, 25, 24.7986, 25.2014,␣
↪25, 25, 25, 25, 25.2014, 24.7986, 25]
print("randomized rabbit:", rabbit_data)
```

```
randomized rabbit: [25, 25, 25, 25, 25.2014, 25.2014, 24.7986, 24.
↪7986, 25.2014,
25.2014, 24.7986, 24.7986, 25, 25, 24.7986, 25.2014, 25, 25, 25, 25,␣
↪25.2014,
24.7986, 25]
```

[6]:
```
#first, find mean and std of rabbit_data
print(np.mean(rabbit_data))
print(np.std(rabbit_data))

# For each value, subtract the midpoint and divide by 3
coded_rabbit_data = [(x - 25) /0.14547426486884318 for x in␣
↪rabbit_data]

# Display the adjusted data
print("coded rabbit data:", coded_rabbit_data)
```

```
25.0
0.14547426486884318
```

```
coded rabbit data: [0.0, 0.0, 0.0, 0.0, 1.3844373104863459, 1.
↪3844373104863459,
-1.3844373104863459, -1.3844373104863459, 1.3844373104863459,
1.3844373104863459, -1.3844373104863459, -1.3844373104863459, 0.0, 0.
↪0,
-1.3844373104863459, 1.3844373104863459, 0.0, 0.0, 0.0, 0.0, 1.
↪3844373104863459,
-1.3844373104863459, 0.0]
```

[7]:
```
print(np.sum(coded_rabbit_data))
```

```
0.0
```

[8]:
```
#shuffle goat data
import random
np.random.seed(142)
random.shuffle(goat_data)
goat_data=[45.3277, 45.3277, 44.6723, 44.6723, 45, 45, 45, 45, 45.
↪3277, 44.6723, 45.3277, 44.6723, 45, 45, 45, 45, 45.3277, 44.
↪6723, 45, 45, 45, 45.3277, 44.6723]
print("randomized goat:", goat_data)
```

```
randomized goat: [45.3277, 45.3277, 44.6723, 44.6723, 45, 45, 45,␣
↪45, 45.3277,
44.6723, 45.3277, 44.6723, 45, 45, 45, 45, 45.3277, 44.6723, 45, 45,␣
↪45,
45.3277, 44.6723]
```

[9]:
```
# First, find the mean and std for goat manure
print(np.mean(goat_data))
print(np.std(goat_data))
# For each value, subtract the midpoint and divide by 3
coded_goat_data = [(x - 45) / 0.2367026643372395 for x in␣
↪goat_data]

# Display the adjusted data
print("coded_goat_data:", coded_goat_data)
```

```
45.0
```

```
0.2367026643372395

coded_goat_data: [1.3844373104863457, 1.3844373104863457, -1.
↪3844373104863457,
-1.3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.3844373104863457,
-1.3844373104863457, 1.3844373104863457, -1.3844373104863457, 0.0, 0.
↪0, 0.0,
0.0, 1.3844373104863457, -1.3844373104863457, 0.0, 0.0, 0.0, 1.
↪3844373104863457,
-1.3844373104863457]
```

[10]:
```
np.sum(coded_goat_data)
```

[10]: 0.0

[12]:
```
import pandas as pd

# Your provided data
coded_poultry_data = [-1.3844373104863459, 1.3844373104863459, -1.
↪3844373104863459, 1.3844373104863459, -1.3844373104863459, 1.
↪3844373104863459, -1.3844373104863459, 1.3844373104863459, 0.0,␣
↪0.0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.
↪0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.0,␣
↪0.0]
coded_rabbit_data = [0.0, 0.0, 0.0, 0.0, 1.3844373104863459, 1.
↪3844373104863459, -1.3844373104863459, -1.3844373104863459, 1.
↪3844373104863459, 1.3844373104863459, -1.3844373104863459, -1.
↪3844373104863459, 0.0, 0.0, -1.3844373104863459, 1.
↪3844373104863459, 0.0, 0.0, 0.0, 0.0, 1.3844373104863459, -1.
↪3844373104863459, 0.0]
coded_goat_data = [1.3844373104863457, 1.3844373104863457, -1.
↪3844373104863457, -1.3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.
↪3844373104863457, -1.3844373104863457, 1.3844373104863457, -1.
↪3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
↪3844373104863457, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
↪3844373104863457]
poultry_data = [34.5653, 35.4347,34.5653 , 35.4347, 34.5653, 35.
↪4347,  34.5653, 35.4347, 35, 35, 35, 35, 34.5653,35.4347, 35,␣
↪35, 35, 35, 34.5653, 35.4347, 35, 35, 35]
```

```
rabbit_data = [25, 25, 25, 25, 25.2014, 25.2014, 24.7986, 24.7986,␣
↪25.2014, 25.2014, 24.7986, 24.7986, 25, 25, 24.7986, 25.2014,␣
↪25, 25, 25, 25, 25.2014, 24.7986, 25]
goat_data = [45.3277, 45.3277, 44.6723, 44.6723, 45, 45, 45, 45,␣
↪45.3277, 44.6723, 45.3277, 44.6723, 45, 45, 45, 45, 45.3277, 44.
↪6723, 45, 45, 45, 45.3277, 44.6723]
weight_yield=[52, 87, 73, 55, 74, 58, 92, 75, 72, 78, 89, 72, 65,␣
↪56, 56, 77, 56, 59, 62, 60,60, 80, 68]

# Creating a DataFrame
df = pd.DataFrame({
    'Coded Poultry Data': coded_poultry_data,
    'Coded Rabbit Data': coded_rabbit_data,
    'Coded Goat Data': coded_goat_data,
    'Poultry Data': poultry_data,
    'Rabbit Data': rabbit_data,
    'Goat Data': goat_data,
    'weight in grams': weight_yield

})

# Display the DataFrame
df



# Export DataFrame to Excel
Project_dataset = 'output.xlsx'  # Name of the Excel file
dataset = 'Sheet1'  # Name of the sheet within the Excel file

df.to_excel(Project_dataset, sheet_name=dataset, index=False)

print(f"DataFrame successfully exported to '{Project_dataset}'")
```

```
DataFrame successfully exported to 'output.xlsx'
```

[13]:

```
#intercept
intercept= [1] * 23
# Your provided data
coded_poultry_data = [-1.3844373104863459, 1.3844373104863459, -1.
↪3844373104863459, 1.3844373104863459, -1.3844373104863459, 1.
↪3844373104863459, -1.3844373104863459, 1.3844373104863459, 0.0,␣
↪0.0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.
↪0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.0,␣
↪0.0]
```

```
coded_rabbit_data = [0.0, 0.0, 0.0, 0.0, 1.3844373104863459, 1.
↪3844373104863459, -1.3844373104863459, -1.3844373104863459, 1.
↪3844373104863459, 1.3844373104863459, -1.3844373104863459, -1.
↪3844373104863459, 0.0, 0.0, -1.3844373104863459, 1.
↪3844373104863459, 0.0, 0.0, 0.0, 0.0, 1.3844373104863459, -1.
↪3844373104863459, 0.0]
coded_goat_data = [1.3844373104863457, 1.3844373104863457, -1.
↪3844373104863457, -1.3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.
↪3844373104863457, -1.3844373104863457, 1.3844373104863457, -1.
↪3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
↪3844373104863457, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
↪3844373104863457]

# Calculate squares of each list
squared_poultry_data = [x**2 for x in coded_poultry_data]
squared_rabbit_data = [x**2 for x in coded_rabbit_data]
squared_goat_data = [x**2 for x in coded_goat_data]

# Calculate interactions
poultry_rabbit_interaction = [a*b for a, b in␣
↪zip(coded_poultry_data, coded_rabbit_data)]
poultry_goat_interaction = [a*b for a, b in␣
↪zip(coded_poultry_data, coded_goat_data)]
rabbit_goat_interaction = [a*b for a, b in zip(coded_rabbit_data,␣
↪coded_goat_data)]

# Print results
print("intercept:", intercept)
print("Squared Poultry Data:", squared_poultry_data)
print("Squared Rabbit Data:", squared_rabbit_data)
print("Squared Goat Data:", squared_goat_data)
print("Poultry-Rabbit Interaction:", poultry_rabbit_interaction)
print("Poultry-Goat Interaction:", poultry_goat_interaction)
print("Rabbit-Goat Interaction:", rabbit_goat_interaction)
```

```
intercept: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,␣
↪1, 1, 1, 1]
Squared Poultry Data: [1.916666666666667, 1.916666666666667, 1.
↪916666666666667,
1.916666666666667, 1.916666666666667, 1.916666666666667, 1.
↪916666666666667,
```

```
1.916666666666667, 0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.
  ↪916666666666667,
0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.916666666666667, 0.0, 0.0,␣
  ↪0.0]
Squared Rabbit Data: [0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.
  ↪916666666666667,
1.916666666666667, 1.916666666666667, 1.916666666666667, 1.
  ↪916666666666667,
1.916666666666667, 1.916666666666667, 0.0, 0.0, 1.916666666666667,
1.916666666666667, 0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.
  ↪916666666666667,
0.0]
Squared Goat Data: [1.9166666666666663, 1.9166666666666663, 1.
  ↪9166666666666663,
1.9166666666666663, 0.0, 0.0, 0.0, 0.0, 1.9166666666666663, 1.
  ↪9166666666666663,
1.9166666666666663, 1.9166666666666663, 0.0, 0.0, 0.0, 0.0, 1.
  ↪9166666666666663,
1.9166666666666663, 0.0, 0.0, 0.0, 1.9166666666666663, 1.
  ↪9166666666666663]
Poultry-Rabbit Interaction: [-0.0, 0.0, -0.0, 0.0, -1.
  ↪916666666666667,
1.916666666666667, 1.916666666666667, -1.916666666666667, 0.0, 0.0,␣
  ↪-0.0, -0.0,
-0.0, 0.0, -0.0, 0.0, 0.0, 0.0, -0.0, 0.0, 0.0, -0.0, 0.0]
Poultry-Goat Interaction: [-1.9166666666666665, 1.9166666666666665,
1.9166666666666665, -1.9166666666666665, -0.0, 0.0, -0.0, 0.0, 0.0,␣
  ↪-0.0, 0.0,
-0.0, -0.0, 0.0, 0.0, 0.0, 0.0, -0.0, -0.0, 0.0, 0.0, 0.0, -0.0]
Rabbit-Goat Interaction: [0.0, 0.0, -0.0, -0.0, 0.0, 0.0, -0.0, -0.0,
1.9166666666666665, -1.9166666666666665, -1.9166666666666665,
```

```
1.9166666666666665, 0.0, 0.0, -0.0, 0.0, 0.0, -0.0, 0.0, 0.0, 0.0,

-1.9166666666666665, -0.0]
```

[14]:

```
import numpy as np

# Data provided
intercept = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,␣
 ↪1, 1, 1, 1, 1]
coded_poultry_data = [-1.3844373104863459, 1.3844373104863459, -1.
 ↪3844373104863459, 1.3844373104863459, -1.3844373104863459, 1.
 ↪3844373104863459, -1.3844373104863459, 1.3844373104863459, 0.0,␣
 ↪0.0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.
 ↪0, 0.0, 0.0, -1.3844373104863459, 1.3844373104863459, 0.0, 0.0,␣
 ↪0.0]
coded_rabbit_data = [0.0, 0.0, 0.0, 0.0, 1.3844373104863459, 1.
 ↪3844373104863459, -1.3844373104863459, -1.3844373104863459, 1.
 ↪3844373104863459, 1.3844373104863459, -1.3844373104863459, -1.
 ↪3844373104863459, 0.0, 0.0, -1.3844373104863459, 1.
 ↪3844373104863459, 0.0, 0.0, 0.0, 0.0, 1.3844373104863459, -1.
 ↪3844373104863459, 0.0]
coded_goat_data = [1.3844373104863457, 1.3844373104863457, -1.
 ↪3844373104863457, -1.3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.
 ↪3844373104863457, -1.3844373104863457, 1.3844373104863457, -1.
 ↪3844373104863457, 0.0, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
 ↪3844373104863457, 0.0, 0.0, 0.0, 1.3844373104863457, -1.
 ↪3844373104863457]
squared_poultry_data = [1.916666666666667, 1.916666666666667, 1.
 ↪916666666666667, 1.916666666666667, 1.916666666666667, 1.
 ↪916666666666667, 1.916666666666667, 1.916666666666667, 0.0, 0.0,␣
 ↪0.0, 0.0, 1.916666666666667, 1.916666666666667, 0.0, 0.0, 0.0, 0.
 ↪0, 1.916666666666667, 1.916666666666667, 0.0, 0.0, 0.0]
squared_rabbit_data = [0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.
 ↪916666666666667, 1.916666666666667, 1.916666666666667, 1.
 ↪916666666666667, 1.916666666666667, 1.916666666666667, 1.
 ↪916666666666667, 0.0, 0.0, 1.916666666666667, 1.916666666666667,␣
 ↪0.0, 0.0, 0.0, 0.0, 1.916666666666667, 1.916666666666667, 0.0]
squared_goat_data = [1.9166666666666663, 1.9166666666666663, 1.
 ↪9166666666666663, 1.9166666666666663, 0.0, 0.0, 0.0, 0.0, 1.
 ↪9166666666666663, 1.9166666666666663, 1.9166666666666663, 1.
 ↪9166666666666663, 0.0, 0.0, 0.0, 0.0, 1.9166666666666663, 1.
 ↪9166666666666663, 0.0, 0.0, 0.0, 1.9166666666666663, 1.
 ↪9166666666666663]
```

```python
poultry_rabbit_interaction = [-0.0, 0.0, -0.0, 0.0, -1.
↪916666666666667, 1.916666666666667, 1.916666666666667, -1.
↪916666666666667, 0.0, 0.0, -0.0, -0.0, -0.0, 0.0, -0.0, 0.0, 0.
↪0, 0.0, -0.0, 0.0, 0.0, -0.0, 0.0]
poultry_goat_interaction = [-1.9166666666666665, 1.
↪9166666666666665, 1.9166666666666665, -1.9166666666666665, -0.0,␣
↪0.0, -0.0, 0.0, 0.0, -0.0, 0.0, -0.0, -0.0, 0.0, 0.0, 0.0, 0.0,␣
↪-0.0, -0.0, 0.0, 0.0, 0.0, -0.0]
rabbit_goat_interaction = [0.0, 0.0, -0.0, -0.0, 0.0, 0.0, -0.0,␣
↪-0.0, 1.9166666666666665, -1.9166666666666665, -1.
↪9166666666666665, 1.9166666666666665, 0.0, 0.0, -0.0, 0.0, 0.0,␣
↪-0.0, 0.0, 0.0, 0.0, -1.9166666666666665, -0.0]

# Creating a matrix X
X = np.array([
    intercept,
    coded_poultry_data,
    coded_rabbit_data,
    coded_goat_data,
    squared_poultry_data,
    squared_rabbit_data,
    squared_goat_data,
    poultry_rabbit_interaction,
    poultry_goat_interaction,
    rabbit_goat_interaction
]).T  # Transpose to align data correctly

print(X)
```

```
[[ 1.         -1.38443731  0.          1.38443731  1.91666667  0.
   1.91666667 -0.         -1.91666667  0.        ]
 [ 1.          1.38443731  0.          1.38443731  1.91666667  0.
   1.91666667  0.          1.91666667  0.        ]
 [ 1.         -1.38443731  0.         -1.38443731  1.91666667  0.
   1.91666667 -0.          1.91666667 -0.        ]
 [ 1.          1.38443731  0.         -1.38443731  1.91666667  0.
   1.91666667  0.         -1.91666667 -0.        ]
```

```
[ 1.         -1.38443731  1.38443731  0.          1.91666667  1.
↪91666667
  0.         -1.91666667 -0.          0.        ]
[ 1.          1.38443731  1.38443731  0.          1.91666667  1.
↪91666667
  0.          1.91666667  0.          0.        ]
[ 1.         -1.38443731 -1.38443731  0.          1.91666667  1.
↪91666667
  0.          1.91666667 -0.         -0.        ]
[ 1.          1.38443731 -1.38443731  0.          1.91666667  1.
↪91666667
  0.         -1.91666667  0.         -0.        ]
[ 1.          0.          1.38443731  1.38443731  0.          1.
↪91666667
  1.91666667  0.          0.          1.91666667]
[ 1.          0.          1.38443731 -1.38443731  0.          1.
↪91666667
  1.91666667  0.         -0.         -1.91666667]
[ 1.          0.         -1.38443731  1.38443731  0.          1.
↪91666667
  1.91666667 -0.          0.         -1.91666667]
[ 1.          0.         -1.38443731 -1.38443731  0.          1.
↪91666667
  1.91666667 -0.         -0.          1.91666667]
[ 1.         -1.38443731  0.          0.          1.91666667  0.
  0.         -0.         -0.          0.        ]
[ 1.          1.38443731  0.          0.          1.91666667  0.
  0.          0.          0.          0.        ]
[ 1.          0.         -1.38443731  0.          0.          1.
↪91666667
```

```
  0.         -0.          0.         -0.         ]
 [ 1.         0.          1.38443731  0.          0.          1.
↪91666667
  0.          0.          0.          0.         ]
 [ 1.         0.          0.          1.38443731  0.          0.
  1.91666667  0.          0.          0.         ]
 [ 1.         0.          0.         -1.38443731  0.          0.
  1.91666667  0.         -0.         -0.         ]
 [ 1.        -1.38443731  0.          0.          1.91666667  0.
  0.         -0.         -0.          0.         ]
 [ 1.         1.38443731  0.          0.          1.91666667  0.
  0.          0.          0.          0.         ]
 [ 1.         0.          1.38443731  0.          0.          1.
↪91666667
  0.          0.          0.          0.         ]
 [ 1.         0.         -1.38443731  1.38443731  0.          1.
↪91666667
  1.91666667 -0.          0.         -1.91666667]
 [ 1.         0.          0.         -1.38443731  0.          0.
  1.91666667  0.         -0.         -0.         ]]
```

[15]:
```
# we use the least square method B=inverse(x^X)(x^Y)
#transpose of matrix X
t= X.T
t
```

[15]:
```
array([[ 1.        ,  1.        ,  1.        ,  1.        ,  1.        ␣
↪    ,
         1.        ,  1.        ,  1.        ,  1.        ,  1.        ␣
↪    ,
         1.        ,  1.        ,  1.        ,  1.        ,  1.        ␣
↪    ,
```

```
        1.        ,  1.        ,  1.        ,  1.        ,  1.        ␣
↪       ,
        1.        ,  1.        ,  1.        ],
      [-1.38443731,  1.38443731, -1.38443731,  1.38443731, -1.
↪38443731,
        1.38443731, -1.38443731,  1.38443731,  0.        ,  0.        ␣
↪       ,
        0.        ,  0.        , -1.38443731,  1.38443731,  0.        ␣
↪       ,
        0.        ,  0.        ,  0.        , -1.38443731,  1.
↪38443731,
        0.        ,  0.        ,  0.        ],
      [ 0.        ,  0.        ,  0.        ,  0.        ,  1.
↪38443731,
        1.38443731, -1.38443731, -1.38443731,  1.38443731,  1.
↪38443731,
       -1.38443731, -1.38443731,  0.        ,  0.        , -1.
↪38443731,
        1.38443731,  0.        ,  0.        ,  0.        ,  0.        ␣
↪       ,
        1.38443731, -1.38443731,  0.        ],
      [ 1.38443731,  1.38443731, -1.38443731, -1.38443731,  0.        ␣
↪       ,
        0.        ,  0.        ,  0.        ,  1.38443731, -1.
↪38443731,
        1.38443731, -1.38443731,  0.        ,  0.        ,  0.        ␣
↪       ,
        0.        ,  1.38443731, -1.38443731,  0.        ,  0.        ␣
↪       ,
        0.        ,  1.38443731, -1.38443731],
      [ 1.91666667,  1.91666667,  1.91666667,  1.91666667,  1.
↪91666667,
        1.91666667,  1.91666667,  1.91666667,  0.        ,  0.        ␣
↪       ,
        0.        ,  0.        ,  1.91666667,  1.91666667,  0.        ␣
↪       ,
        0.        ,  0.        ,  0.        ,  1.91666667,  1.
↪91666667,
        0.        ,  0.        ,  0.        ],
      [ 0.        ,  0.        ,  0.        ,  0.        ,  1.
↪91666667,
        1.91666667,  1.91666667,  1.91666667,  1.91666667,  1.
↪91666667,
```

```
        1.91666667,  1.91666667,  0.        ,  0.        ,  1.
↪91666667,
        1.91666667,  0.        ,  0.        ,  0.        ,  0.        ␣
↪     ,
        1.91666667,  1.91666667,  0.        ],
      [ 1.91666667,  1.91666667,  1.91666667,  1.91666667,  0.        ␣
↪     ,
        0.        ,  0.        ,  0.        ,  1.91666667,  1.
↪91666667,
        1.91666667,  1.91666667,  0.        ,  0.        ,  0.        ␣
↪     ,
        0.        ,  1.91666667,  1.91666667,  0.        ,  0.        ␣
↪     ,
        0.        ,  1.91666667,  1.91666667],
      [-0.        ,  0.        , -0.        ,  0.        , -1.
↪91666667,
        1.91666667,  1.91666667, -1.91666667,  0.        ,  0.        ␣
↪     ,
       -0.        , -0.        , -0.        ,  0.        , -0.        ␣
↪     ,
        0.        ,  0.        ,  0.        , -0.        ,  0.        ␣
↪     ,
        0.        , -0.        ,  0.        ],
      [-1.91666667,  1.91666667,  1.91666667, -1.91666667, -0.        ␣
↪     ,
        0.        , -0.        ,  0.        ,  0.        , -0.        ␣
↪     ,
        0.        , -0.        , -0.        ,  0.        ,  0.        ␣
↪     ,
        0.        ,  0.        , -0.        , -0.        ,  0.        ␣
↪     ,
        0.        ,  0.        , -0.        ],
      [ 0.        ,  0.        , -0.        , -0.        ,  0.        ␣
↪     ,
        0.        , -0.        , -0.        ,  1.91666667, -1.
↪91666667,
       -1.91666667,  1.91666667,  0.        ,  0.        , -0.        ␣
↪     ,
        0.        ,  0.        , -0.        ,  0.        ,  0.        ␣
↪     ,
        0.        , -1.91666667, -0.        ]])
```

```
[16]: #product of matrix and transpose
M= np.dot(t, X)
print(M)
```

```
[[ 2.30000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
   2.30000000e+01  2.30000000e+01  2.30000000e+01  0.00000000e+00
   0.00000000e+00 -1.91666667e+00]
 [ 0.00000000e+00  2.30000000e+01 -1.35803386e-16 -5.04411556e-17
   1.32320025e-16  1.32320025e-16  9.82779375e-17 -1.32320025e-16
   3.84044367e-17  0.00000000e+00]
 [ 0.00000000e+00 -1.35803386e-16  2.30000000e+01 -1.91666667e+00
  -1.32320025e-16 -1.32320025e-16 -2.65350485e+00  1.32320025e-16
   0.00000000e+00  2.65350485e+00]
 [ 0.00000000e+00 -5.04411556e-17 -1.91666667e+00  2.30000000e+01
  -1.50823742e-16  2.65350485e+00 -1.16781655e-16  0.00000000e+00
  -1.99007197e-17 -2.65350485e+00]
 [ 2.30000000e+01  1.32320025e-16 -1.32320025e-16 -1.50823742e-16
   4.40833333e+01  1.46944444e+01  1.46944444e+01  2.46716228e-16
  -2.34380416e-16  0.00000000e+00]
 [ 2.30000000e+01  1.32320025e-16 -1.32320025e-16  2.65350485e+00
   1.46944444e+01  4.40833333e+01  1.83680556e+01  2.46716228e-16
   0.00000000e+00 -3.67361111e+00]
 [ 2.30000000e+01  9.82779375e-17 -2.65350485e+00 -1.16781655e-16
   1.46944444e+01  1.83680556e+01  4.40833333e+01  0.00000000e+00
  -2.89891568e-16 -3.67361111e+00]
 [ 0.00000000e+00 -1.32320025e-16  1.32320025e-16  0.00000000e+00
   2.46716228e-16  2.46716228e-16  0.00000000e+00  1.46944444e+01
   0.00000000e+00  0.00000000e+00]
 [ 0.00000000e+00  3.84044367e-17  0.00000000e+00 -1.99007197e-17
```

```
 -2.34380416e-16  0.00000000e+00 -2.89891568e-16  0.00000000e+00
  1.46944444e+01  0.00000000e+00]
 [-1.91666667e+00  0.00000000e+00  2.65350485e+00 -2.65350485e+00
  0.00000000e+00 -3.67361111e+00 -3.67361111e+00  0.00000000e+00
  0.00000000e+00  1.83680556e+01]]
```

[17]:

```
# M is the matrix for which you want to find the inverse
M_inverse = np.linalg.inv(M)

print(M_inverse)
```

```
[[ 4.94728916e-01  2.32591345e-18 -1.52295356e-02  1.52295356e-02
  -1.61471975e-01 -1.44971189e-01 -1.44971189e-01  5.28224624e-18
  -5.41488421e-18 -1.96437926e-03]
 [ 2.32591345e-18  4.34782609e-02  1.99637790e-19  1.96830494e-19
  -8.50845443e-19 -7.89001488e-19 -6.90533904e-19  3.91511539e-19
  -1.13631932e-19 -5.36086166e-20]
 [-1.52295356e-02  1.99637790e-19  4.52427559e-02  2.81216399e-03
   4.54048267e-03  2.62870049e-03  7.58738552e-03 -5.27769283e-19
   2.25914181e-19 -5.67560334e-03]
 [ 1.52295356e-02  1.96830494e-19  2.81216399e-03  4.52427559e-02
  -4.54048267e-03 -7.58738552e-03 -2.62870049e-03  1.78301146e-19
  -6.30085014e-20  5.67560334e-03]
 [-1.61471975e-01 -8.50845443e-19  4.54048267e-03 -4.54048267e-03
   7.97367162e-02  4.07907623e-02  4.07907623e-02 -2.06451368e-18
   2.07039225e-18 -1.84480834e-03]
 [-1.44971189e-01 -7.89001488e-19  2.62870049e-03 -7.58738552e-03
   4.07907623e-02  7.19691022e-02  3.25700374e-02 -1.91688214e-18
   1.28288896e-18  4.30455280e-03]
 [-1.44971189e-01 -6.90533904e-19  7.58738552e-03 -2.62870049e-03
```

```
  4.07907623e-02  3.25700374e-02  7.19691022e-02 -1.30003301e-18
  2.06686811e-18  4.30455280e-03]
 [ 5.28224624e-18  3.91511539e-19 -5.27769283e-19  1.78301146e-19
 -2.06451368e-18 -1.91688214e-18 -1.30003301e-18  6.80529301e-02
 -5.93583272e-35  9.80895199e-21]
 [-5.41488421e-18 -1.13631932e-19  2.25914181e-19 -6.30085014e-20
  2.07039225e-18  1.28288896e-18  2.06686811e-18 -5.93583272e-35
  6.80529301e-02  6.31813749e-20]
 [-1.96437926e-03 -5.36086166e-20 -5.67560334e-03  5.67560334e-03
 -1.84480834e-03  4.30455280e-03  4.30455280e-03  9.80895199e-21
  6.31813749e-20  5.75990161e-02]]
```

[18]:

```
#taking Y(the yield) as a Vector
Y=[52, 87, 73, 55, 74, 58, 92, 75, 72, 78, 89, 72, 65, 56, 56, 77,␣
 ↪56, 59, 62, 60,60, 80, 68]

#product of Y vector and X tranpose(t)
N= np.dot(t,Y)
print(N)
```

```
[1576.          -37.37980738  -62.29967897   42.91755663 1550.
 ↪58333333
 1692.41666667 1611.91666667    1.91666667  101.58333333 -197.
 ↪41666667]
```

[19]:

```
#finding the coefficients of our model
W =np.dot(M_inverse,N)
print(W)
```

```
[52.27409639 -1.62520902 -1.85972096  0.52914048  3.8313253   7.
 ↪7375865
  4.46985195  0.13043478  6.91304348 -2.50654793]
```

[20]:
```
import numpy as np
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols

poultry_data = [34.5653, 35.4347,34.5653 , 35.4347, 34.5653, 35.
↪4347,  34.5653, 35.4347, 35, 35, 35, 35, 34.5653,35.4347, 35,␣
↪35, 35, 35, 34.5653, 35.4347, 35, 35, 35]
rabbit_data = [25, 25, 25, 25, 25.2014, 25.2014, 24.7986, 24.7986,␣
↪25.2014, 25.2014, 24.7986, 24.7986, 25, 25, 24.7986, 25.2014,␣
↪25, 25, 25, 25, 25.2014, 24.7986, 25]
goat_data = [45.3277, 45.3277, 44.6723, 44.6723, 45, 45, 45, 45,␣
↪45.3277, 44.6723, 45.3277, 44.6723, 45, 45, 45, 45, 45.3277, 44.
↪6723, 45, 45, 45, 45.3277, 44.6723]
Y=[52, 87, 73, 55, 74, 58, 92, 75, 72, 78, 89, 72, 65, 56, 56, 77,␣
↪56, 59, 62, 60,60, 80, 68]
# Combine your data into a single DataFrame
#data = {
    #'Value': np.concatenate([poultry_data, rabbit_data,␣
↪goat_data,Y]),
    #'Group': ['Poultry']*len(poultry_data) +␣
↪['Rabbit']*len(rabbit_data) + ['Goat']*len(goat_data) +␣
↪['Y']*len(Y)
#}

df = pd.DataFrame({'poultry_data':poultry_data, 'rabbit_data':
↪rabbit_data, 'goat_data':goat_data , 'Y':Y})


#fit a polynomial regression
formula='Y~ 1 + poultry_data + rabbit_data +goat_data +␣
↪poultry_data:rabbit_data + poultry_data:goat_data + rabbit_data:
↪goat_data + I(poultry_data**2) + I(rabbit_data**2) +␣
↪I(goat_data**2)'
model = ols(formula = formula, data = df).fit()

#perform anova
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

[20]:

	sum_sq	df	F	PR(>F)
poultry_data	184.793235	1.0	2.359066	0.148531
rabbit_data	878.142449	1.0	11.210344	0.005239

```
goat_data                      302.387933   1.0    3.860277  0.071183
poultry_data:rabbit_data         0.250000   1.0    0.003191  0.955808
poultry_data:goat_data         702.250000   1.0    8.964906  0.010355
rabbit_data:goat_data          109.077949   1.0    1.392486  0.259129
I(poultry_data ** 2)           184.094032   1.0    2.350140  0.149241
I(rabbit_data ** 2)            831.888172   1.0   10.619863  0.006222
I(goat_data ** 2)              277.613251   1.0    3.544004  0.082333
Residual                      1018.331959  13.0         NaN       NaN
```

[67]:
```
# get model summary
summary = model.summary()
summary
```

[67]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results

Dep. Variable:                      Y   R-squared:                       ␣
↪       0.666
Model:                            OLS   Adj. R-squared:                  ␣
↪       0.435
Method:                 Least Squares   F-statistic:                     ␣
↪       2.881
Date:                Mon, 04 Mar 2024   Prob (F-statistic):              ␣
↪      0.0409
Time:                        19:57:46   Log-Likelihood:                  ␣
↪     -76.225
No. Observations:                  23   AIC:                             ␣
↪       172.5
Df Residuals:                      13   BIC:                             ␣
↪       183.8
Df Model:                           9
Covariance Type:            nonrobust
               coef    std err          t      P>|t|      [0.025
0.975]

Intercept                5.052e+05   1.85e+05      2.724      0.
↪017    1.05e+05
9.06e+05
poultry_data            -6982.4940   2588.411     -2.698      0.
↪018   -1.26e+04
-1390.573
```

```
rabbit_data              -1.512e+04   6662.468     -2.269      0.
 ↪041    -2.95e+04
-724.815
goat_data                -8613.5340   4346.699     -1.982      0.
 ↪069     -1.8e+04
776.938
poultry_data:rabbit_data     2.8556     50.547      0.056      0.
 ↪956     -106.344
112.055
poultry_data:goat_data      93.0143     31.065      2.994      0.
 ↪010       25.902
160.127
rabbit_data:goat_data      -72.7925     61.687     -1.180      0.
 ↪259     -206.058
60.473
I(poultry_data ** 2)        38.8612     25.350      1.533      0.
 ↪149      -15.903
93.625
I(rabbit_data ** 2)        365.6227    112.195      3.259      0.
 ↪006      123.240
608.005
I(goat_data ** 2)           79.7787     42.378      1.883      0.
 ↪082      -11.773
171.331

Omnibus:                        0.119   Durbin-Watson:               ␣
 ↪       2.941
Prob(Omnibus):                  0.942   Jarque-Bera (JB):            ␣
 ↪       0.117
Skew:                          -0.115   Prob(JB):                    ␣
 ↪       0.943
Kurtosis:                       2.737   Cond. No.                    ␣
 ↪    3.26e+08


Notes:
[1] Standard Errors assume that the covariance matrix of the␣
 ↪errors is correctly
specified.
[2] The condition number is large, 3.26e+08. This might indicate␣
 ↪that there are
strong multicollinearity or other numerical problems.
"""
```

[21]:
```
df
```

[21]:

	poultry_data	rabbit_data	goat_data	Y
0	34.5653	25.0000	45.3277	52
1	35.4347	25.0000	45.3277	87
2	34.5653	25.0000	44.6723	73
3	35.4347	25.0000	44.6723	55
4	34.5653	25.2014	45.0000	74
5	35.4347	25.2014	45.0000	58
6	34.5653	24.7986	45.0000	92
7	35.4347	24.7986	45.0000	75
8	35.0000	25.2014	45.3277	72
9	35.0000	25.2014	44.6723	78
10	35.0000	24.7986	45.3277	89
11	35.0000	24.7986	44.6723	72
12	34.5653	25.0000	45.0000	65
13	35.4347	25.0000	45.0000	56
14	35.0000	24.7986	45.0000	56
15	35.0000	25.2014	45.0000	77
16	35.0000	25.0000	45.3277	56
17	35.0000	25.0000	44.6723	59
18	34.5653	25.0000	45.0000	62
19	35.4347	25.0000	45.0000	60
20	35.0000	25.2014	45.0000	60
21	35.0000	24.7986	45.3277	80
22	35.0000	25.0000	44.6723	68

[1]:
```
import numpy as np

# Define the design matrix Z_1
Z_1 = np.array([
    [1.00, 0.31, 0.31, 0.31, 0, 0, 0, 0, 0, 0],
    [0.31, 0.21, 0.07, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.21, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.07, 0.21, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0.31, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.31, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.31, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.07, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.07, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.07]
])

# Compute the determinant of Z_1
```

```
det_Z_1 = np.linalg.det(Z_1)

# Compute S as the number of rows in Z_1
S = Z_1.shape[0]

# Compute D-optimality criterion
D_optimality = det_Z_1**(1/S)

# Print the result
print(f"D-optimality criterion: {D_optimality}")
```

```
D-optimality criterion: 0.16187957059154778
```

[2]:
```
import numpy as np

# Define the design matrix Z_1
Z_1 = np.array([
    [1.00, 0.31, 0.31, 0.31, 0, 0, 0, 0, 0, 0],
    [0.31, 0.21, 0.07, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.21, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.07, 0.21, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0.31, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.31, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.31, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.07, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.07, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.07]
])

# Compute the trace of Z_1
trace_Z_1 = np.trace(Z_1)

# Compute S as the number of rows in Z_1
S = Z_1.shape[0]

# Compute T-criterion
T_criterion = (1/S) * trace_Z_1

# Print the result
print(f"T-criterion: {T_criterion}")
```

```
T-criterion: 0.27699999999999997
```

[ ]:

[3]:
```
import numpy as np

# Define the design matrix Z_1
Z_1 = np.array([
    [1.00, 0.31, 0.31, 0.31, 0, 0, 0, 0, 0, 0],
    [0.31, 0.21, 0.07, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.21, 0.07, 0, 0, 0, 0, 0, 0],
    [0.31, 0.07, 0.07, 0.21, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0.31, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.31, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.31, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.07, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.07, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.07]
])

# Compute eigenvalues of Z_1
eigenvalues_Z_1 = np.linalg.eigvals(Z_1)

# Compute E-optimality criterion (using minimum eigenvalue)
E_criterion = np.min(eigenvalues_Z_1)

# Print the result
print(f"E-optimality criterion: {E_criterion}")
```

```
E-optimality criterion: 0.047365552251949924
```

[4]:
```
import numpy as np

# Define the design matrix Z2
Z2 = np.array([
    [1.00, 0.50, 0.50, 0.50, 0, 0, 0, 0, 0, 0],
    [0.50, 0.57, 0.19, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.57, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.19, 0.57, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0.50, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.50, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.50, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.19]
])
```

```
def d_optimality_criterion(Z):
    """
    Calculate the D-optimality criterion for a given design matrix␣
 ↪Z.

    Parameters:
    Z (numpy.ndarray): The design matrix.

    Returns:
    float: The D-optimality criterion.
    """
    # Calculate determinant of Z
    det_Z = np.linalg.det(Z)

    # Number of rows or columns in Z
    S = Z.shape[0]

    # Calculate D-optimality criterion
    d_optimality = det_Z ** (1 / S)

    return d_optimality

# Calculate the D-optimality criterion for the given design matrix␣
 ↪Z2
d_optimality_value = d_optimality_criterion(Z2)

print("D-optimality criterion value:", d_optimality_value)
```

```
D-optimality criterion value: 0.34623965031641984
```

[5]:
```
import numpy as np

# Define the design matrix Z2
Z2 = np.array([
    [1.00, 0.50, 0.50, 0.50, 0, 0, 0, 0, 0, 0],
    [0.50, 0.57, 0.19, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.57, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.19, 0.57, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0.50, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.50, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.50, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 0],
```

```
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.19]
])

def t_optimality_criterion(Z):
    """
    Calculate the T-optimality criterion for a given design matrix␣
 ↪Z.

    Parameters:
    Z (numpy.ndarray): The design matrix.

    Returns:
    float: The T-optimality criterion.
    """
    # Calculate trace of Z
    trace_Z = np.trace(Z)

    # Number of rows or columns in Z
    S = Z.shape[0]

    # Calculate T-optimality criterion
    t_optimality = trace_Z / S

    return t_optimality

# Calculate the T-optimality criterion for the given design matrix␣
 ↪Z2
t_optimality_value = t_optimality_criterion(Z2)

print("T-optimality criterion value:", t_optimality_value)
```

```
T-optimality criterion value: 0.4780000000000001
```

[ ]:

```
```

[6]:

```
import numpy as np

# Define the design matrix Z_2
Z_2 = np.array([
    [1.00, 0.50, 0.50, 0.50, 0, 0, 0, 0, 0, 0],
    [0.50, 0.57, 0.19, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.57, 0.19, 0, 0, 0, 0, 0, 0],
    [0.50, 0.19, 0.19, 0.57, 0, 0, 0, 0, 0, 0],
```

```
    [0, 0, 0, 0, 0.50, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0.50, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.50, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 0.19]
])

# Compute the eigenvalues of Z_2
eigenvalues, _ = np.linalg.eig(Z_2)

# Select the smallest eigenvalue (E-optimality criterion)
E_optimality = min(eigenvalues)

# Print the result
print(f"E-optimality criterion: {E_optimality}")
```

```
E-optimality criterion: 0.10861382744182721
```

[7]:

```
import numpy as np

# Define the design matrix Z_3
Z_3 = np.array([
    [1.00, 1.34, 1.34, 1.34, 0, 0, 0, 0, 0, 0],
    [1.34, 3.27, 1.09, 1.09, 0, 0, 0, 0, 0, 0],
    [1.34, 1.09, 3.27, 1.09, 0, 0, 0, 0, 0, 0],
    [1.34, 1.09, 1.09, 3.27, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 1.34, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 1.34, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 1.34, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.19, 1.09, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 1.09, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 1.09]
])

# Number of runs (S)
S = Z_3.shape[0]

# Compute the determinant of Z_3
det_Z_3 = np.linalg.det(Z_3)

# Compute the D-optimality criterion
D_optimality = det_Z_3 ** (1 / S)
```

```
# Print the result
print(f"D-optimality criterion: {D_optimality}")
```

```
D-optimality criterion: 0.9933959737056217
```

[8]:
```
import numpy as np

# Define the design matrix Z_3
Z_3 = np.array([
    [1.00, 1.34, 1.34, 1.34, 0, 0, 0, 0, 0, 0],
    [1.34, 3.27, 1.09, 1.09, 0, 0, 0, 0, 0, 0],
    [1.34, 1.09, 3.27, 1.09, 0, 0, 0, 0, 0, 0],
    [1.34, 1.09, 1.09, 3.27, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 1.34, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 1.34, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 1.34, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.19, 1.09, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 1.09, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 1.09]
])

# Number of runs (S)
S = Z_3.shape[0]

# Compute the trace of Z_3
trace_Z_3 = np.trace(Z_3)

# Compute the T-criterion
T_criterion = trace_Z_3 / S

# Print the result
print(f"T-criterion: {T_criterion}")
```

```
T-criterion: 1.8099999999999998
```

[ ]:
```

```

[9]:
```
import numpy as np

# Define the design matrix Z_3
Z_3 = np.array([
    [1.00, 1.34, 1.34, 1.34, 0, 0, 0, 0, 0, 0],
    [1.34, 3.27, 1.09, 1.09, 0, 0, 0, 0, 0, 0],
```

```
    [1.34, 1.09, 3.27, 1.09, 0, 0, 0, 0, 0, 0],
    [1.34, 1.09, 1.09, 3.27, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 1.34, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 1.34, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 1.34, 0, 0, 0],
    [0, 0, 0, 0, 0, 0, 0.19, 1.09, 0, 0],
    [0, 0, 0, 0, 0, 0, 0, 0.19, 1.09, 0],
    [0, 0, 0, 0, 0, 0, 0, 0, 0.19, 1.09]
])

# Compute eigenvalues of Z_3
eigenvalues_Z_3 = np.linalg.eigvals(Z_3)

# Compute E-optimality criterion
E_criterion = np.prod(eigenvalues_Z_3)**(1/Z_3.shape[0])

# Print the result
print(f"E-optimality criterion: {E_criterion}")
```

```
E-optimality criterion: 0.9933959737056203
```

[ ]:

```

```

SR572

ISO 9001:2019 Certified Institution

THESIS WRITING COURSE

PLAGIARISM AWARENESS CERTIFICATE

This certificate is awarded to

Matundura Dennis Mwan

PhD/Bs/03/17

In recognition for passing the University's plagiarism

Awareness test for Thesis **entitled: CONSTRUCTION OF AUGMENTED SECOND ORDER ROTATABLE DESIGN USING TRIGONOMETRIC FUNCTIONS IN MODELLING SWEET POTATOES (Ipomea batatas) YIELD**
with similarity index of 9% and striving to maintain academic integrity.

Word count: 21738

Awarded by

Prof. Anne Syomwene Kisilu
CERM-ESA Project Leader   Date: 01/07/2024