



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A DATA FARMING ANALYSIS OF MULTIPLE SALVO EQUATIONS

by

Gokhan Kesler

March 2019

Thesis Advisor:
Second Reader:

Thomas W. Lucas
Paul J. Sanchez

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE March 2019	3. REPORT TYPE AND DATES COVERED Master's thesis		
4. TITLE AND SUBTITLE A DATA FARMING ANALYSIS OF MULTIPLE SALVO EQUATIONS			5. FUNDING NUMBERS	
6. AUTHOR(S) Gokhan Kesler				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) In 1995, retired U.S. Navy Captain Wayne Hughes formulated a salvo model for assessing the military worth of warship capabilities in the missile age. His model suggests that modern naval surface warfare can have regions of instability and that numerical superiority provides a consistent advantage. However, Hughes's model is deterministic and provides no information about the distribution of outcomes that can result from an inherently stochastic salvo exchange. In 2005, Michael Armstrong added random variables to Hughes's model and created a stochastic salvo model. By using mathematical approximations, Armstrong provides closed-form solutions for some stochastic outcomes. This thesis examines the two salvo models using data farming. That is, sophisticated designs of experiments are used to evaluate the models at thousands of input combinations. For each model, responses such as ship losses, proportional ship losses, and fractional exchange ratio (FER) are reformulated as readily interpretable regression and partition tree metamodels of the model's inputs. For the stochastic salvo model, this is also done for the probability that one side wins. The metamodel fits are outstanding, generally explaining over 97 percent of the variance of the experimental results. The data farming results reinforce Hughes's basic findings.				
14. SUBJECT TERMS experimental design, NOLH, salvo model, simulation, regression model, military models			15. NUMBER OF PAGES 145	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

A DATA FARMING ANALYSIS OF MULTIPLE SALVO EQUATIONS

Gokhan Kesler
First Lieutenant, Turkish Air Force
BSE, Turkish Air Force Academy, 2011

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

**NAVAL POSTGRADUATE SCHOOL
March 2019**

Approved by: Thomas W. Lucas
Advisor

Paul J. Sanchez
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In 1995, retired U.S. Navy Captain Wayne Hughes formulated a salvo model for assessing the military worth of warship capabilities in the missile age. His model suggests that modern naval surface warfare can have regions of instability and that numerical superiority provides a consistent advantage. However, Hughes's model is deterministic and provides no information about the distribution of outcomes that can result from an inherently stochastic salvo exchange. In 2005, Michael Armstrong added random variables to Hughes's model and created a stochastic salvo model. By using mathematical approximations, Armstrong provides closed-form solutions for some stochastic outcomes. This thesis examines the two salvo models using data farming. That is, sophisticated designs of experiments are used to evaluate the models at thousands of input combinations. For each model, responses such as ship losses, proportional ship losses, and fractional exchange ratio (FER) are reformulated as readily interpretable regression and partition tree metamodels of the model's inputs. For the stochastic salvo model, this is also done for the probability that one side wins. The metamodel fits are outstanding, generally explaining over 97 percent of the variance of the experimental results. The data farming results reinforce Hughes's basic findings.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	BACKGROUND	2
B.	THESIS OBJECTIVE	3
C.	RESEARCH QUESTIONS	4
D.	SCOPE OF THESIS	4
E.	THESIS FLOW	5
F.	METHODOLOGY	5
II.	BACKGROUND	7
A.	DETERMINISTIC SALVO MODEL (HUGHES).....	7
1.	Model Development	7
2.	Illustration	9
B.	CLOSED-FORM OF STOCHASTIC SALVO MODEL (ARMSTRONG)	11
1.	Model Development	11
2.	Illustration	16
III.	LINEAR MODELS AND DESIGN OF EXPERIMENT	23
A.	LINEAR MODEL	23
B.	DESIGN OF EXPERIMENTS	24
1.	Terminology and Notation of DOE	25
2.	Categories of Factors	25
3.	Experimental Designs	26
C.	ANALYSIS TOOLS USED IN THIS RESEARCH.....	33
1.	JMP® Pro	33
2.	R.....	34
3.	NOB Mixed Design Worksheet.....	34
D.	CHAPTER SUMMARY	35
IV.	ANALYSIS PART 1: DETERMINISTIC SALVO MODEL	37
A.	DESIGNING THE EXPERIMENT	37
1.	Experimental Flow.....	37
2.	Data Farming	38
3.	Analysis of the Design.....	40
B.	SUMMARY OUTPUT ANALYSIS OF HUGHES’S DETERMINISTIC SALVO EXPERIMENTS	41
C.	REGRESSION AND PARTITION MODEL ANALYSIS.....	44

1.	The Actual Number of Ship Loss	44
2.	Loss Ratio	51
D.	CHAPTER SUMMARY.....	60
V.	ANALYSIS PART 2: STOCHASTIC SALVO MODEL (SOLVED BY SIMULATION).....	63
A.	DESIGNING THE EXPERIMENT	63
1.	Experimental Flow.....	63
2.	Data Farming	64
3.	Analysis of Design Point Sets	65
B.	OUTPUT ANALYSIS OF STOCHASTIC SALVO SIMULATION EXPERIMENTS.....	67
C.	REGRESSION MODEL ANALYSIS.....	69
1.	Actual Number of Ship Loss	69
2.	Fractional Exchange Ratio (<i>FER</i>).....	76
3.	Probability of Victory	81
D.	CHAPTER SUMMARY.....	85
VI.	SUMMARY, CONCLUSION, AND IDEAS FOR FUTURE RESEARCH	87
	APPENDIX A. CODE FOR CHAPTER IV.....	89
	APPENDIX B. CODE FOR CHAPTER V.....	95
	APPENDIX C. REGRESSION RESULTS	103
	LIST OF REFERENCES.....	119
	INITIAL DISTRIBUTION LIST	123

LIST OF FIGURES

Figure 1.	The Behavior of FER and Instability Areas for Different Values of Unbounded ΔA and ΔB	10
Figure 2.	Distribution of Loss (Ships) Suffered by <i>Side B</i> (Stochastic Model)	17
Figure 3.	Victory Conditions.....	18
Figure 4.	Probability of Victory for Different Size of B	19
Figure 5.	Standard Deviation for Different Ship Size B	20
Figure 6.	2^2 , 10^2 , and 100^2 Factorial Designs for BMI	28
Figure 7.	Representation of Factorial Designs with Three Factors. Adapted from Sanchez and Wan (2012).	29
Figure 8.	LH Design for BMI.....	30
Figure 9.	Scatterplot Matrices of 4^4 Factorial and NOLH. Adapted from Sanchez and Wan (2012).	31
Figure 10.	Design Comparison. Source: Sanchez and Wan (2012).	32
Figure 11.	Screenshot of JMP® Pro 14.....	33
Figure 12.	Screenshot of RStudio.....	34
Figure 13.	Screenshot of NOB Mixed Design Worksheet	35
Figure 14.	Flow Chart of the Deterministic Experiments	38
Figure 15.	Pairwise Projections of the DPs for the Deterministic Salvo Experiment.....	40
Figure 16.	Correlation Matrix for the Factors in the Deterministic Salvo Experiment.....	41
Figure 17.	Summary Output Distributions from the Deterministic Salvo Model	42
Figure 18.	Predicted ΔB Values from the Metamodel versus the Actual Values Obtained by the Experiments.....	45
Figure 19.	Predicted ΔB Values from the Metamodel Versus the Actual Values Obtained by the Experiments for the Intermediate Cases.....	46

Figure 20.	Prediction Profiler of ΔB (Deterministic)	48
Figure 21.	Heat Map of ΔB (Deterministic)	49
Figure 22.	Tree Diagram of ΔB (Deterministic).....	50
Figure 23.	Predicted $\Delta B/B$ Values Versus the Actual Values Obtained by the Experiments	51
Figure 24.	Predicted $\Delta B/B$ Values Versus the Actual Values Obtained by the Experiments for the Intermediate Cases.	52
Figure 25.	Prediction Profiler of the Loss Ratio (Deterministic)	53
Figure 26.	Tree Model of the Loss Ratio (Deterministic).....	54
Figure 27.	Histogram of the Logarithmically Transformed FER (Deterministic Model).....	55
Figure 28.	Sensitivity Analysis on R-squared for Given Range of FER (Deterministic)	57
Figure 29.	Comparison of Predicted FER Values with Actual Values (Deterministic Model).....	58
Figure 30.	Prediction Profiler for FER (Deterministic Model)	59
Figure 31.	Tree Diagram of FER (Deterministic)	60
Figure 32.	Flow Chart of the Stochastic Simulation Experiments	64
Figure 33.	Pairwise Plots from the Design Matrix of the Stochastic Salvo Experiment.....	66
Figure 34.	Pairwise Correlation Between Factors (Stochastic).....	67
Figure 35.	Summarized Output Data Distributions for the Stochastic Model Simulation	68
Figure 36.	Comparison of Predicted ΔB Values with the Actual Values (Stochastic)	70
Figure 37.	Comparison of Predicted ΔB Values with the Actual Values by Category (Stochastic).....	71
Figure 38.	Predicted ΔB Values versus Simulation Values for the Metamodel Based on Intermediate Outcomes	72

Figure 39.	Comparison of Predicted ΔB Values with Actual Values by Category	73
Figure 40.	Prediction Profiler for ΔB (Stochastic)	74
Figure 41.	Heat Map of ΔB (Stochastic)	75
Figure 42.	Tree Diagram of ΔB (Stochastic)	76
Figure 43.	Histogram of the Logarithmic Transformed FER (Stochastic).....	77
Figure 44.	Sensitivity Analysis on R-squared for Given Range of FER (Stochastic)	78
Figure 45.	Comparison of Predicted FER Values with Actual Values (Stochastic)	79
Figure 46.	Prediction Profiler for FER (Stochastic).....	80
Figure 47.	Tree Graph of FER (Stochastic)	81
Figure 48.	Comparison of Predicted Probability of Win for $Side B$ Values with Actual Values (Stochastic).....	82
Figure 49.	Predicted Winning Probability Values from Logistic Regression versus Simulation Values (Stochastic).....	83
Figure 50.	Prediction Profiler for $Probability of Win$ (Stochastic)	84
Figure 51.	Tree Diagram of $Winning Probability$ (Stochastic)	85

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Hughes’s Model Notation. Adapted from Armstrong (2005).....	8
Table 2.	Armstrong’s Model Notation. Adapted from Armstrong (2005).....	12
Table 3.	Parameters of the Example	16
Table 4.	Mixed NOB Design for Deterministic Salvo Model	39
Table 5.	NOB Design for Stochastic Salvo Model Experiments	65

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ASM	anti-ship missile
BMI	body mass index
CDF	cumulative distribution function
DOE	design of experiment
DP	design point
<i>FER</i>	fractional exchange ratio
GLM	generalized regression model
IID	independent and identically distributed
LH	Latin hypercube
LM	linear model
NPS	Naval Postgraduate School
NOB	nearly orthogonal and balanced
NOLH	nearly orthogonal Latin hypercube
OR	operations research
PDF	probability density function
SAM	surface-to-air missile
SEED	simulation experiments and efficient designs

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Models are used to obtain information affecting how military forces are equipped, organized, and fight. For example, Hughes (1995) created an analytical model designed for analyzing the elements of ship design in modern naval surface combat. In his analysis, he illustrated the interactions between ship numbers and their offensive, defensive, and staying powers. Hughes's salvo equations became famous for their relative simplicity and ease of implementation.

One of the limitations of Hughes's salvo model is that it is deterministic, and therefore provides no information on the variability inherent in combat. With this limitation in mind, Armstrong (2005) developed a stochastic salvo model that provides outputs in terms of means, variances, and probabilities. This model is similar to Hughes's, except that Armstrong replaced fixed model inputs with random variables.

This thesis examines these two salvo models using data farming. Thousands of carefully chosen input combinations are evaluated to obtain model outputs on ship losses, the proportion of forces lost, and fractional exchange ratio (*FER*). In addition, for the stochastic model, the probability of win is also calculated. The input-to-output mappings are reformulated as readily interpretable metamodels (i.e., statistical models of Hughes's and Armstrong's salvo models), describing the outcomes as functions of the input factors. The metamodels are found using stepwise regression and partition trees. The metamodels generally explain over 97% of the variance of the experimental results. The results reinforce Hughes's basic findings that modern naval surface warfare can have regions of instability and that numerical superiority provides a consistent advantage.

Figure ES-1 shows a summary of the analysis on Hughes's deterministic salvo model. The x-axes represent the regression metamodels' predictions, while the y-axes contain the actual results from 5,120 input combinations on three output measures (ship loss, loss ratio, and *FER*). The diagonal line (equality line) shows the criteria for a perfect fit of the metamodel to the salvo model. Overall, the metamodels for the three outputs fit extremely well, with R^2 values ranging from 0.969 to 0.999.

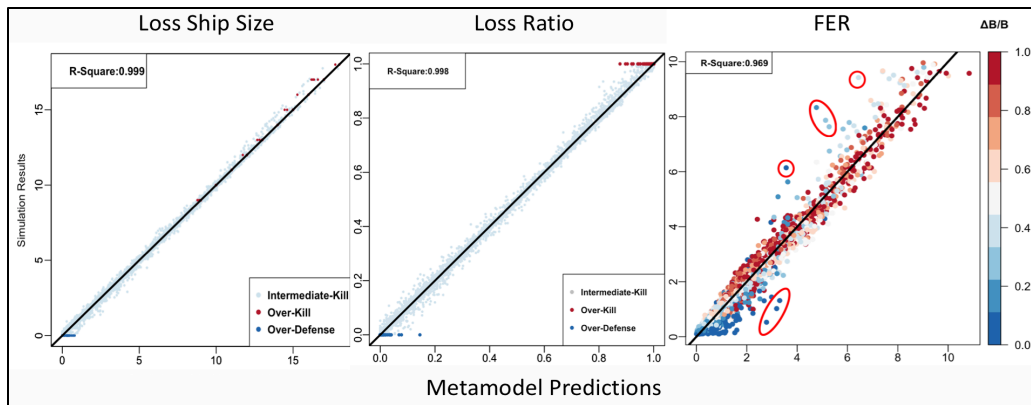


Figure ES-1. Overall Goodness of Fit between the Regression Metamodel and Hughes's Deterministic Salvo Model

Figure ES-2 shows how well the regression metamodels for ship losses, probability of win, and *FER* match the results of simulations of Armstrong's stochastic salvo model. Again, the metamodels almost perfectly match the actual outcomes.

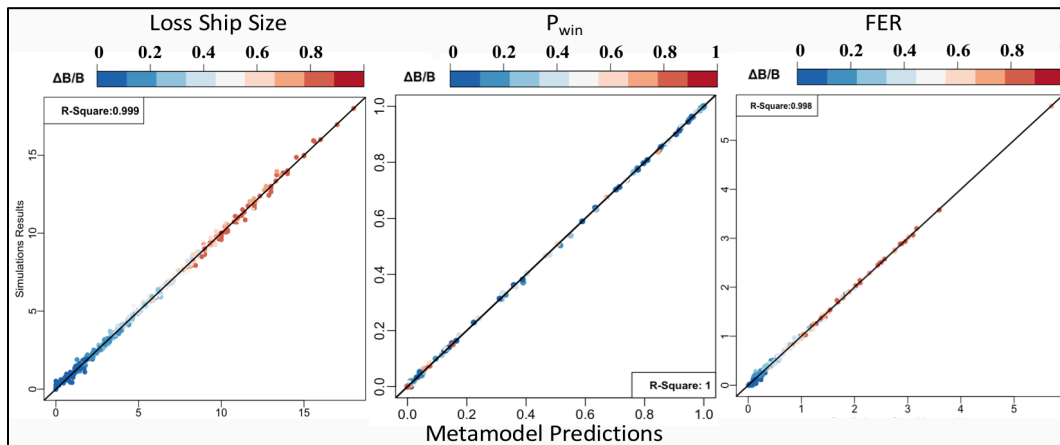


Figure ES-2. Overall Goodness of Fit between the Regression Metamodel and Simulations of Armstrong's Stochastic Salvo Model

References

- Armstrong, M. J. (2005). A stochastic salvo model for naval surface combat. *Operations Research*, 53(5), 830–841.
- Hughes, W. P. (1995). A salvo model of warships in missile combat used to evaluate their staying power. *Naval Research Logistics*, 42(2), 267–289.

ACKNOWLEDGMENTS

It was hard but also a joyful experience for me to be a graduate student of the operations analysis program at the Naval Postgraduate School. I am so grateful to the Turkish Air Force for giving the financial support for me to study here.

First, I give profound and heartfelt gratitude to my valued advisor Professor Thomas W. Lucas and second reader Dr. Paul J. Sanchez. Without their encouragement, guidance, and patience, this thesis would not have been possible. Also, I am much obliged to those who contributed to my education, namely Michael Atkinson, Samuel E. Buttrey, W. Matthew Carlyle, Robert F. Dell, Jim Eagle, Kyle Y. Lin, Connor McLemore, Jeff House, Jeff Hyink, Bob Koyak, Dan Nussbaum, Guillermo Owen, Pante Stanica, Roberto Szechtman, and Lyn Whitaker.

Additional thanks to the International Graduate Programs Office staff, Alan D. Scott, Kathi Noyes, Kimberly Andersen, and Deborah Graham, who are responsible for our administrative affairs. I appreciate the Thesis Processing Office staff of NPS, also known as “dragon ladies,” for pushing me to generate this high-quality thesis. They instilled in me the motto “No pain, no gain,” and contributed to my personal development in academic writing.

Last, I sincerely thank my parents for their unconditional support. Most importantly, I would like to thank with love to my wife, Ezgi, for her passion, presence, and understanding in all steps of this thesis, and without her, I would not have achieved this success.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Our true mentor in life is science.

—Mustafa Kemal Atatürk

Models are indispensable tools for solving real-life problems. A model provides understanding and insights regarding possible outcomes of real-life situations. They can be created in many different varieties, including mathematical equations, as well as scaled or non-scaled versions of real systems, and computer simulations. Their primary purpose, however, is to imitate critical aspects of the behavior of the system on which the model is based.

Models are cost-effective. Sometimes, the cost of building a real system can be extremely expensive, while such a system can be modeled at a small fraction of the cost. For instance, in pilot training, flight simulators are used by many countries' air forces, since the cost of an individual sortie is expensive due to the price of jet fuel and the maintenance of fighter jets. Another reason for building a model rather than executing an actual system is that models are risk-free (e.g., a plane crash in a simulation costs almost nothing), and may provide a valuable lesson learned. A useful model can help to prevent undesired consequences for unpredictable and uncontrollable events, such as experimenting with a nuclear reactor in a modeled environment before actually testing it. A model's ability to scale time is another advantage. For example, executing eight hours of a customer queue system modeled in computer simulation likely takes an only a small amount of time compared to the time needed for an actual observation.

Many models are implemented and studied using computer simulation, herein referred to as "simulation." Simulations used for modeling real systems are an indispensable part of scientific research. Simulation is not only a valuable tool in the modern scientific world but also an indispensable resource within the business and military communities. The models used reflect real-life situations that are generally extraordinarily complex and often involve many factors (i.e., inputs) that potentially affect outcomes.

To understand the effects of multiple factors and their potential interactions, a high-dimensional design of experiments and comprehensive analysis of their results is critical. Simulation experimentation benefits from high-performance computers, since models can consist of thousands of continuous and/or discrete quantitative values and qualitative factors, as well as millions of potential interactions among them. To illustrate this challenge, a simulation having 100 factors with only two levels for each factor requires 205,000 years to be executed using the world's fastest computer, "Oak Ridge National Laboratory's Summit," even if each experiment involves only a single elementary operation (Sanchez and Wan, 2012). Therefore, appropriate methodologies that reduce the number of experiments are necessary to overcome the well-known "curse of dimensionality."

With this backdrop, this thesis uses modern design of experiment methodologies to study models of naval missile salvo warfare.

A. BACKGROUND

Among all types of models, military models are of particular interest to academia and military, because these models ideally represent important complex phenomenon while simultaneously being simple to use. Military models are regularly used to obtain information affecting how forces are equipped, organized, and fight. These decisions can involve billions of dollars and potentially impact many lives.

Different kinds of analytical models of naval surface combat have been developed over the past 120 years. One of them is the classic square law model, devised by J. V. Chase (1902), a military officer at the Naval War College. His model mathematically expresses naval combat as force-on-force attrition. His study was classified by the United States Navy until 1972. For that reason, credit for similar results is usually given to F. W. Lanchester (1914) and M. Osipov (1915), who independently published their square law combat models.

Lanchester's equations have been widely used in warfare models for the past century, primarily for land combat. However, critical assumptions in Lanchester's equations include that attrition occurs continuously and the attrition rate coefficients are

constant. For modeling aggregate-level land combat, these assumptions may be reasonable in some situations, but modern naval platforms are relatively small in number and they attack in salvos (or pulses) of missile fire. In addition, naval platforms are defended by defensive missiles that are not explicitly accounted for in Lanchester's formulation. These differences demand that a different form of analytical model is needed for studying naval missile warfare.

Hughes (1995) offered a new form of an analytical model designed for analyzing the elements of ship design in modern naval surface combat. Instead of a continuous stream of gunfire, he proposed a model using discrete salvos of missiles. In his analysis, he illustrated the interactions between ship numbers and their offensive, defensive, and staying powers. Hughes's salvo equations became famous for their relative simplicity and ease of implementation, with four main factors on each side that are vital for analysts and decision makers.

One of the limitations of Hughes's salvo model is that it is deterministic. That is, it provides no information about variability. This can be misleading in real-life problems (Lucas, 2000). Because of the complex and uncertain nature of warfare, a combat model is usually more informative if it provides estimates of distributions of possible outcomes.

With this limitation in mind, Armstrong (2005) developed an analytic stochastic salvo model that provides outputs in terms of means, variances, and probabilities. Armstrong's stochastic salvo model replaces the fixed inputs to Hughes's deterministic salvo model with random variables (e.g., the amount of damage caused by offensive missiles can vary). During his study, Armstrong also tried to maintain the model's ease of use (similar to Hughes) as much as possible. The stochastic model's main results are broadly similar to those of the original deterministic model. Furthermore, the stochastic version provides probabilistic results, allowing military planners to consider distributions of possibilities rather than point values.

B. THESIS OBJECTIVE

This study examines the naval salvo models created mathematically by Hughes (1995) and Armstrong (2005) using data farming. Data farming uses high-performance

computing and large-scale designed experiments to efficiently “grow” simulation output data (Lucas et al., 2015). Analysis of the outputs can reveal relationships between input factors and responses, thus providing a greater understanding of the models. Much of the learning is done by building metamodels (statistical models of Hughes’s and Armstrong’s models’ behaviors) that map inputs to outputs in readily interpretable forms, such as regression equations and partition trees.

The primary objective of this thesis is to provide an understanding of both the deterministic and stochastic salvo models. In addition to that, methods like regression modeling are used to show how easily model outputs can be formulated as readily interpretable functions of the input parameters. For a further examination, it is essential to have a deeper understanding of the deterministic and stochastic salvo models. This is accomplished by replicating some of the analysis done by Hughes and Armstrong.

C. RESEARCH QUESTIONS

The following questions guide this research:

- What can we learn about Hughes’s and Armstrong’s salvo models using data farming?
- What are the behavioral similarities and differences between these two models?
- Can we reveal interesting relationships that are not directly observable from the mathematical formulations?
- Can we formulate additional outputs, such as probability of winning and fractional exchange ratio (*FER*), for the stochastic model?

D. SCOPE OF THESIS

The scope of this thesis includes:

- Providing an understanding of the deterministic and stochastic salvo models through massive experimentation on them.

- Analyzing Armstrong's stochastic salvo model using simulation and data farming.

E. THESIS FLOW

Thesis progresses as follows:

- Chapter II provides an overview of Hughes's basic deterministic and Armstrong's closed-form stochastic salvo models.
- Chapter III contains a literature review on regression models, design of experiments, and data farming techniques.
- Chapter IV analyzes Hughes's deterministic salvo model using data farming. This includes fitting regression and partition metamodels.
- Chapter V investigates simulation results of Armstrong's stochastic salvo model with data farming. This includes fitting regression and partition metamodels.
- Chapter VI presents a brief summary of the thesis, as well as its bottom-line conclusions and recommendations for future research.

F. METHODOLOGY

This thesis studies the deterministic and stochastic salvo models using data farming. Using state-of-the-art design of experiments, both salvo models are evaluated over a broad range of assumptions and input conditions. The results are used to fit readily interpretable metamodels. Our primary goal is an enhanced understanding of the salvo models.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

This chapter provides detailed information about the naval salvo models created by Hughes and Armstrong. These two models are defined and illustrated with examples.

A. DETERMINISTIC SALVO MODEL (HUGHES)

Retired U.S. Navy Captain Wayne Hughes developed salvo equations offering a transparent and straightforward methodology for understanding critical ship design aspects of modern naval surface combat (Hughes, 1995). The basic idea is to analyze the exchange of *offensive power* with anti-ship missiles (ASM) and *defensive power* with surface-to-air missiles (SAM). Hughes also takes into consideration *staying power*, which is a ship's ability to keep fighting after being struck by enemy weapons. In each discrete time slot, the two sides exchange ASMs and SAMs with simultaneous salvo fire in an attempt to defeat the opponent. With its straightforward modeling, Hughes's model allows users to investigate the best combination of offensive, defensive, and staying power, as well as the value of force size. Among his findings, Hughes concludes that naval warfare can be unstable, especially when ships have weak staying power. He also concludes that numerical superiority is consistently critical in achieving an advantage.

1. Model Development

In this section, Hughes's (1995) deterministic salvo model is briefly reviewed. There are numerous applications using Hughes's model, for example Beall (1990), Cares (1990), and Lucas and McGunnigle (2002).

The deterministic salvo model consists of two opposing sides, *Force Alpha* and *Force Bravo*, referred to as *side A* and *side B*, respectively, battling against each other. The number of the ships on each side is represented with A and B . The ships on each side are homogeneous—that is, all of *side A*'s ships are identical. *Side B*'s ships are also identical to each other, but perhaps quite different from *side A*'s ships. *Offensive power* is the available number of well-aimed attacking ASMs per salvo belonging to each individual ship, labeled as α and β for *side A* and *side B*, respectively. Both sides have a capability to

neutralize the incoming ASMs by using a number of defensive missiles (i.e., their SAMs). The number of effective SAMs per ship per salvo is defined as their *defensive power*, and is represented with y and z in the model. Additionally, each ship has a staying power, which presents a limit of endurance to the non-intercepted ASMs. *Staying power* is represented with w and x . If a *side A* ship gets hit by more than w of *side B*'s missiles, this ship is assumed to be out of action. Similarly, if a *side B* ship gets hit with more than x of *side A*'s missiles, that ship is assumed to be out of action. To sum up, Table 1 gives information about Hughes's model using the notation from Armstrong (2005).

Table 1. Hughes's Model Notation. Adapted from Armstrong (2005).

Symbol	Description
A, B	Number of available ships at the beginning of the battle.
α, β	Number of available ASMs per salvo by each ship (<i>offensive power</i>).
y, z	Number of available SAMs per salvo for each ship (<i>defensive power</i>).
w, x	Number of hits needed to put a single ship out of action (<i>staying power</i>).
$\Delta A, \Delta B$	Number of ships neutralized by the adversary's salvo.

Hughes's equations determine the losses for each side in an aggregated simultaneous exchange of missiles. That is, each side simultaneously fires their ASMs. For incoming missiles, each side uses its available *defensive power* (SAM) to destroy as many ASMs as they can. ASMs that are not countered hit the enemy ships. If a ship reaches its limit to absorb ASM hits, this ship is assumed to be put out of action. Ships that are damaged, but not put out of action, lose their fighting strength (i.e., offensive and defensive powers) linearly and proportionate to their staying power. The mathematical formulation of the change in number of ships (i.e., losses) in a simultaneous salvo exchange for both sides is presented in Equation 1.

$$\Delta B = \frac{\alpha A - zB}{x}, 0 \leq \Delta B \leq B \quad \Delta A = \frac{\beta B - yA}{w}, 0 \leq \Delta A \leq A \quad (1)$$

where ΔA and ΔB represent the number of ship losses for *side A* and *side B*, respectively, in a single salvo. Without the bounds in Equation 1, when ΔA or ΔB are negative, this represents a situation in which the defense power is more than enough to counter all incoming ASMs. We will define this as *over-defense*. Without the bounds in Equation 1, when ΔA or ΔB are greater than A or B , respectively, this represents a situation in which one side is completely wiped out with an excess of offensive power. We will define this as *over-kill*.

Additionally, Hughes uses *Fractional Exchange Ratio (FER)* to quantify the ratio of the combat power of the forces. In layman's terms, *FER* is the proportion of *side B* lost divided by the proportion of *side A* lost. Its mathematical expression is given in Equation 2.

$$FER = \frac{\Delta B/B}{\Delta A/A}, \quad 0 \leq FER \leq +\infty \quad (2)$$

FER ranges from zero to positive infinity. Zero occurs when $\Delta B = 0$ and $\Delta A > 0$ (i.e., no *side B* ships are damaged and some *side A* ships are put out of action). Infinity occurs when $\Delta A = 0$ and $\Delta B > 0$ (i.e., no *side A* forces are lost and some *B* forces are put out of action). When each side has equal strength, *FER* equals to one. Values bigger than one imply that *side A* has the advantage, while values less than one favor *side B*.

2. Illustration

To provide better understanding, let us illustrate the salvo exchange with an example taken from Hughes (1995).

Force size (<i>number of ships</i>) :	$A = 2$	$B = 6$
ASMs (<i>offensive power</i>) :	$\alpha = 24$	$\beta = 6$
SAMs (<i>defensive power</i>) :	$y = 16$	$z = 1$
Endurance (<i>staying power</i>) :	$w = 2$	$x = 1$

$$\Delta B = \left\lceil \frac{2 \times 24 - 1 \times 6}{1} \right\rceil = 42 \cong 6, \quad 0 \leq \Delta B \leq 6$$

$$\Delta A = \left\lceil \frac{6 \times 6 - 2 \times 16}{2} \right\rceil = 2, \quad 0 \leq \Delta A \leq 2$$

In this example, *side A* is able to destroy *side B* seven times over, while *side B* can destroy *side A* once. According to raw *FER*, *side A* has an overwhelming advantage over *side B*—with seven times more combat power. Unfortunately, from *side A*'s perspective, offensive power is over-used and redundant, whereas defensive and staying powers are insufficient to protect *side A* from total destruction. From *side B*'s perspective, the number of *side B*'s ships have to be increased until they reach more than 24 for any to survive. However, this calculation is *theoretical*. In actuality, *FER* equals one since both sides are totally destroyed.

This example shows the potential instability of naval missile warfare when a side has enormously more *offensive power* than the adversary's *defensive power* and *staying power*. Colored cells in Figure 1 show instability areas according to ΔA and ΔB when ΔA and ΔB are not bounded between 0 and *A* or *B*, respectively.

	$\Delta B < 0$	$0 \leq \Delta B \leq B$	$\Delta B > B$
$\Delta A < 0$	(0, +Inf)	(-Inf, 0]	(-Inf, 0)
$0 \leq \Delta A \leq A$	(-Inf, 0)	[0, +Inf)	(1, +Inf)
$\Delta A > A$	(-Inf, 0)	[0, 1)	(0, +Inf)

Figure 1. The Behavior of *FER* and Instability Areas for Different Values of Unbounded ΔA and ΔB

In Figure 1, ΔA values determine the rows, while ΔB values define the columns. Blue areas indicate that at least one side has excessive *defensive power*, whereas red areas represent overuse of *offensive power*. Purple areas are a mix of extra *defensive power* for one side and extra *offensive power* for the other side. In such cases, one side is wiped out while the other is unscathed. The intersections of the rows and columns show the interval that can be numerically calculated for *FER* when ΔA and ΔB are allowed to be negative. For example, when ΔA is between 0 and A , while ΔB is between 0 and B , the minimum value for *FER* approaches 0 as ΔB goes to 0, and the maximum value of *FER* approaches positive infinity, which occurs as ΔA goes to 0.

B. CLOSED-FORM OF STOCHASTIC SALVO MODEL (ARMSTRONG)

One of the drawbacks of Hughes's deterministic salvo model is that it provides no information on the randomness and variability that is inherent in warfare. This fact inspired Armstrong (2005) to introduce a stochastic version of the basic salvo model by representing Hughes's *offensive power*, *defensive power*, and *staying power* with random variables. By using mathematical approximations, Armstrong provides closed-form solutions to his model.

In Armstrong's salvo model, which will be referred to as the *stochastic salvo model*, all of the constant parameters in Hughes's model (e.g., α , z , and v) (which is the mean of the damage inflicted = $1/x$) are random variables. Another data farming study on Armstrong's model can be found in Li (2018).

1. Model Development

Armstrong's stochastic salvo model is developed by using Hughes's deterministic salvo model as a baseline. The difference is that multiplications in Equation 1 are replaced with summations of random variables. From now on, the model is explained for *side A* since the calculations for *side B* are symmetric, and therefore essentially the same. Armstrong's notation is presented in Table 2.

Table 2. Armstrong's Model Notation. Adapted from Armstrong (2005).

Symbol	Description
A, B	Number of available ships at the beginning of the battle.
α_i, β_i	<i>Offensive power</i> for each ship.
y_j, z_j	<i>Defensive power</i> for each ship.
w, x	Number of hits needed to take a single ship out of action.
u_k, v_k	Number of ships lost per ASM hit.
n_a, n_β	Number of available ASMs to be fired by each ship.
p_a, p_β	Probability of successful launch and targeting of a single ASM.
Off_A, Off_B	Total number of fired ASMs for each side.
n_y, n_z	Number of available SAMs to intercept ASMs for each ship.
p_a, p_β	Probability of successful interception for a single SAM.
Def_A, Def_B	Total number of successfully engaging defensive SAMs.
Net_{BA}, Net_{AB}	Total nominal number of non-intercepted ASMs.
F_{NetBA}, G_{NetAB}	CDF of nominal number of non-intercepted ASMs.
f_{NetBA}, g_{NetAB}	PDF of nominal number of non-intercepted ASMs.
A_I^*, B_I^*	Nominal number of surviving ships after single iteration.
F_{BI}^*, G_{BI}^*	CDF of nominal surviving ships.
f_{BI}^*, g_{BI}^*	PDF of nominal surviving ships.
A_I, B_I	Actual number of surviving ships after single iteration.

In the model, Off_A represents the random total of offensive ASMs fired by *side A*, while Def_B is the random total of defensive SAMs fired by *side B*, and Net_{AB} represents the number offensive missiles from *side A* that are not intercepted by *side B*'s defenses:

$$Net_{AB} = Off_A - Def_B \quad (3)$$

where, using Armstrong's notation

$$Off_A \equiv A \alpha = \sum_{i=1}^A \alpha_i = \int_0^A \alpha dt \quad Def_B \equiv B z = \sum_{i=1}^B z_i = \int_0^B z dt \quad (4)$$

To determine Net_{AB} , Off_A , and Def_B , let us redefine the parameters of the model. The parameter α_i is the number of successfully targeted offensive missiles for the i^{th} ship on *side A*. Each α_i is a non-negative independent and identically distributed (*iid*) random variable with mean μ_α and standard deviation σ_α . Thus, the offensive power of *side A* is the summation of the total number of missiles launched successfully for each ship. So, the mean and variance for the offensive power of *side A* is as shown in Equation 5.

$$E[Off_A] = \sum_{i=1}^A \mu_{\alpha_i} = A \mu_\alpha \quad Var[Off_A] = \sum_{i=1}^A \sigma_{\alpha_i}^2 = A \sigma_\alpha^2 \quad (5)$$

Each ship has a maximum offensive missile capacity, which can be represented with n_α . If each missile has a probability p_α of successful launch, independent of the others, then, for each ship, the number of successfully targeted missiles follows a binomial distribution with mean $\mu_\alpha = n_\alpha p_\alpha$ and variance $\sigma_\alpha^2 = n_\alpha p_\alpha (1 - p_\alpha)$. The overall expected offensive power of *side A* and its variance are therefore:

$$E[Off_A] = A n_\alpha p_\alpha \quad Var[Off_A] = A n_\alpha p_\alpha (1 - p_\alpha) \quad (6)$$

The same rules apply for the defensive power of *side B*, also binomially distributed as:

$$E[Def_B] = B n_z p_z \quad Var[Def_B] = B n_z p_z (1 - p_z) \quad (7)$$

To illustrate a numerical example, let us define parameters from Armstrong (2005): $n_\alpha = 8$ and $p_\alpha = 0.5$, so that $\mu_\alpha = 4$ ASMs per ship, and likewise $n_z = 4$ and $p_z = 0.5$, so that $\mu_z = 2$ SAMs per ship. Each side has six ships, with $A = B = 6$. It follows that

$E[Off_A] = 24$ and $Var[Off_A] = 12$ and $E[Def_B] = 12$ and $Var[Def_B] = 6$. As the net missiles from *side A* to *side B* is the number of non-intercepted ASMs, $E[Net_{AB}] = 24 - 12 = 12$ and the $Var[Net_{AB}] = 18$.

The number of non-intercepted offensive missiles launched from *side A* to *side B* is the difference between the *defensive power* of *side B* from the *offensive power* of *side A*, as aforementioned. However, since the difference between two independent binomials is quite complicated, Armstrong uses normal approximations of the two binomials. Moreover, the difference between two independent normal random variables is itself a normal random variable with easily obtainable mean and variance. Thus, the representation of the net incoming missiles (i.e., Net_{AB}) from *side A* to *side B* is approximated as a normal random variable with mean $A \mu_a - B \mu_z$ and variance $A \sigma_a^2 + B \sigma_z^2$. It is worthwhile to emphasize that normal random variables are continuous rather than discrete (the difference between two binomials is discrete). In addition, the support of a normal random variable is the entire real line. Of course, in reality, the number of missiles that can be fired, and thus hit, is limited to the integers in the interval $[0, A n_a]$.

Another random factor Armstrong includes is the damage caused from each non-intercepted ASM. This randomness is included by assuming that the damage “ v ” inflicted by each ASM that leaks is independent and identically distributed as a normal random variable with mean μ_v and variance σ_v^2 . The mean damage is derived from the staying power of Hughes’s salvo model, with $\mu_v = I/x$.

To sum up, with the random number of nominal hits and random amount of damage per hit, the nominal surviving *side B* force strength after one salvo can be modelled as in Equation 8, which is the transformation of Equation 1 from the deterministic model into an equivalent form in which a summation (or integral) of the random damage suffered from each missile hit replaces the multiplication of ship size by missile number:

$$B_1^* = B - \sum_{i=0}^{NET_{AB}} v_i = \int_0^{NET_{AB}} v dt \quad 0 \leq B_1^* \leq B \quad (8)$$

With the provided prior information, the mean and the variance of the surviving force are calculated in Equations 9 and 10, respectively:

$$\mu_{B1^*} = B - E[Net_{AB}]E[v] = B - (A\mu_\alpha - B\mu_z)\mu_z \quad (9)$$

$$\begin{aligned} \sigma_{B1^*}^2 = & (A\mu_\alpha - B\mu_z)\sigma_v^2 + (A\sigma_\alpha^2 - B\sigma_z^2)\mu_v^2 - 2\sigma_v^2(A\mu_\alpha - B\mu_z)G_{Net_{AB}}(0) \\ & + 2\sigma_v^2(A\sigma_\alpha^2 - B\sigma_z^2)g_{Net_{AB}}(0) \end{aligned} \quad (10)$$

Note: in what Armstrong refers to as nominal losses, the losses are not bounded by zero and B .

To understand ongoing model development, let us set the average damage as $\mu_v = 0.5$ ships loss per hit and the standard deviation as $\sigma_v = 0.2$ ships. Applying equations (9) and (10) to our numerical example yields an expected nominal surviving force of $\mu_{B1^*} = 6 - 12 \times 0.5 = 0$, and has $\sigma_{B1^*} = 2.23$ as its standard deviation. In this example, with Hughes's model from Equation 1, *side B* is annihilated with certainty.

At this point, Armstrong adjusted the model by using a basic stochastic inventory model from Chopra and Meindl (2004) and continuity corrections from Johnson and Kotz (1969) to derive the actual surviving force strength " B_I " from the nominal surviving force strength. The actual surviving force is bounded between zero and B . It is obtained by truncating the distribution of the nominal losses at zero and B . With all these parameters in place:

- The mean actual surviving force strength is calculated as:

$$E[B_1] = \int_{0+\frac{\mu_v}{2}}^{B-\frac{\mu_v}{2}} t g_{B1^*}(t) dt + B(1 - G_{B1^*}(B - \mu_v/2)) \quad (11)$$

and alternatively, can be expressed as:

$$= \mu_{B1^*} \left[G_{B1^*} \left(B - \frac{\mu_v}{2} \right) - G_{B1^*} \left(0 + \frac{\mu_v}{2} \right) \right] - \sigma_{B1^*}^2 \left[g_{B1^*} \left(B - \frac{\mu_v}{2} \right) - g_{B1^*} \left(0 + \frac{\mu_v}{2} \right) \right] \quad (12)$$

$$+B \left(1 - G_{B1^*} \left(B - \frac{\mu_v}{2} \right) \right)$$

- The variance of the actual surviving force strength is calculated as:

$$Var[B_1] = \int_{0+\frac{\mu_v}{2}}^{B-\frac{\mu_v}{2}} t^2 g_{B1^*}(t) dt + B^2(1 - G_{B1^*}(B - \mu_v/2)) - (E[B_1])^2 \quad (13)$$

which also can be presented as follows:

$$= (\mu_{B1^*}^2 + \sigma_{B1^*}^2) \left[G_{B1^*} \left(B - \frac{\mu_v}{2} \right) - G_{B1^*} \left(0 + \frac{\mu_v}{2} \right) \right] + B^2 \left(1 - G_{B1^*} \left(B - \frac{\mu_v}{2} \right) \right) \quad (14)$$

$$- (E[B_1])^2 - \sigma_{B1^*}^2 \left[\left(B - \frac{\mu_v}{2} + \mu_{B1^*} \right) g_{B1^*} \left(B - \frac{\mu_v}{2} \right) - \left(0 + \frac{\mu_v}{2} + \mu_{B1^*} \right) g_{B1^*} \left(0 + \frac{\mu_v}{2} \right) \right]$$

Applying equations 11 and 13 to the numerical example above results in an average of 0.88 ships surviving with a standard deviation of 1.29.

2. Illustration

To compare Hughes's deterministic salvo model with Armstrong's stochastic salvo model, let us demonstrate some results as force strength B varies. In this example, the ship force size B varies between 1 and 15. All other parameters remain fixed as in Table 3.

Table 3. Parameters of the Example

Stochastic Parameters		Deterministic Parameters	
A	6	A	6
n_α, n_β	8	α, β	8
p_a, p_b	0.68		
n_y, n_z	3	y, z	3
p_y, p_z	0.68		
μ_u, μ_v	0.33	w, x	3
σ_v, σ_v	0.13		

To better understand the numerical results, a single example scenario is considered where the force size on both sides is six (i.e., $A = B = 6$), and all parameters are fixed as aforementioned. By implementing Equation 1 for the deterministic salvo model, the result is that both sides destroy each other entirely, as shown in the calculation:

$$\Delta A = \frac{6 \times 8 - 6 \times 3}{3} = 10, \quad \Delta B = \frac{6 \times 8 - 6 \times 3}{3} = 10$$

On the other hand, the stochastic model provides the exciting probabilistic results depicted in Figure 2.

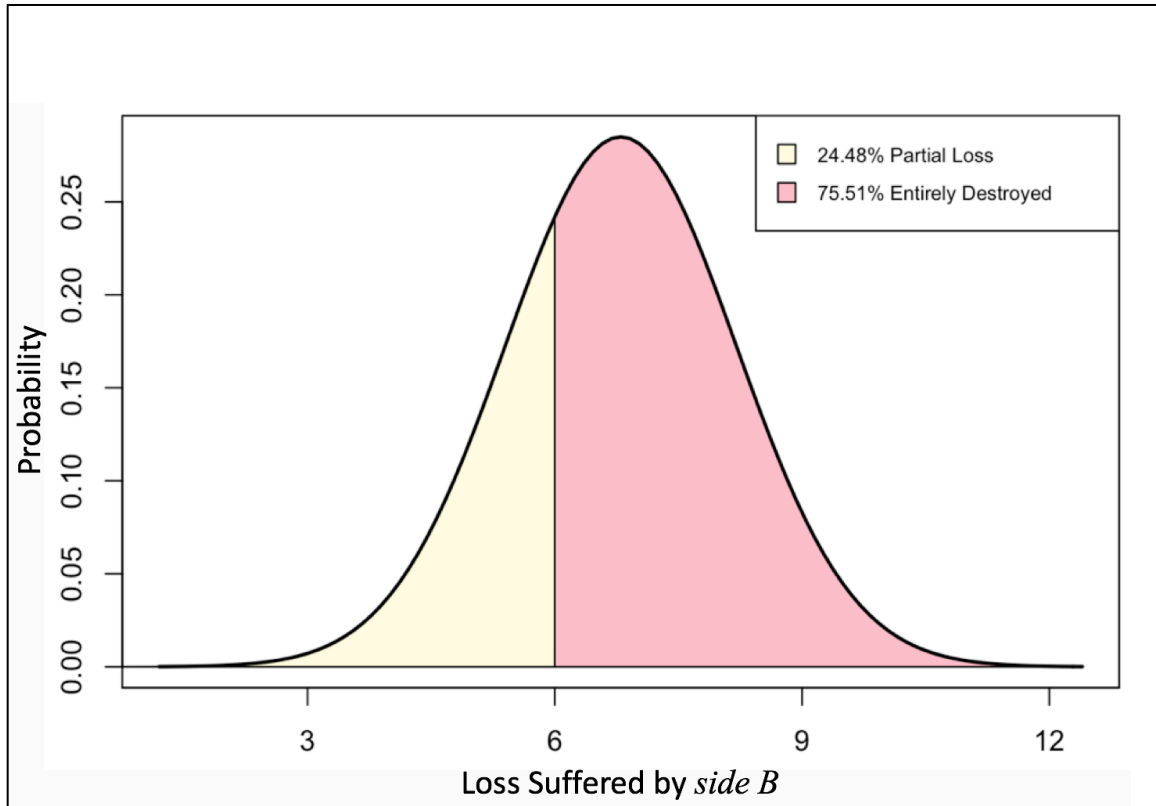


Figure 2. Distribution of Loss (Ships) Suffered by *Side B* (Stochastic Model)

The same *6-on-6* example in the stochastic salvo model has a nominal mean of 6.8 ships lost, with a standard deviation of 1.40. In an actual combat, the feasible result space for ship loss in Figure 2 is colored with yellow (the interval between 0 and 6), whereas the

red area represents entire destruction (or *over-kill*). The graph shows that, with rounding, the probability of partial loss is 0.24 while that of complete destruction is 0.76. These calculations show how the results of the stochastic model are probability distributions while the deterministic model provides a single rigid output.

Another useful comparison is to examine how the probability of victory changes for both the deterministic and stochastic model as *side B*'s strength increases from 1 to 15. "Victory" is defined as the entire destruction of the opponent's side while some friendly forces survive. Figure 3 shows the victory condition matrix.

		<i>side A</i>	
		No Loss or Partial Loss	Completely Destroyed
<i>side B</i>	No Loss or Partial Loss	Intermediate Outcome	Victory of <i>B</i>
	Completely Destroyed	Victory of <i>A</i>	Complete Annihilation

Figure 3. Victory Conditions

To give a single numerical example using the earlier values ($A = B = 6$), $P(\text{Victory for side } A) = (0.76) \times (1 - 0.76) = 0.18$ and the probability of victory for *side B* is the same since both forces are identical.

Figure 4 shows results for both the deterministic and stochastic salvo models as the number of ships on *side B* is varied between 1 and 15. The deterministic and stochastic results are presented with red and blue bars and lines for *side A* and *side B*, respectively.

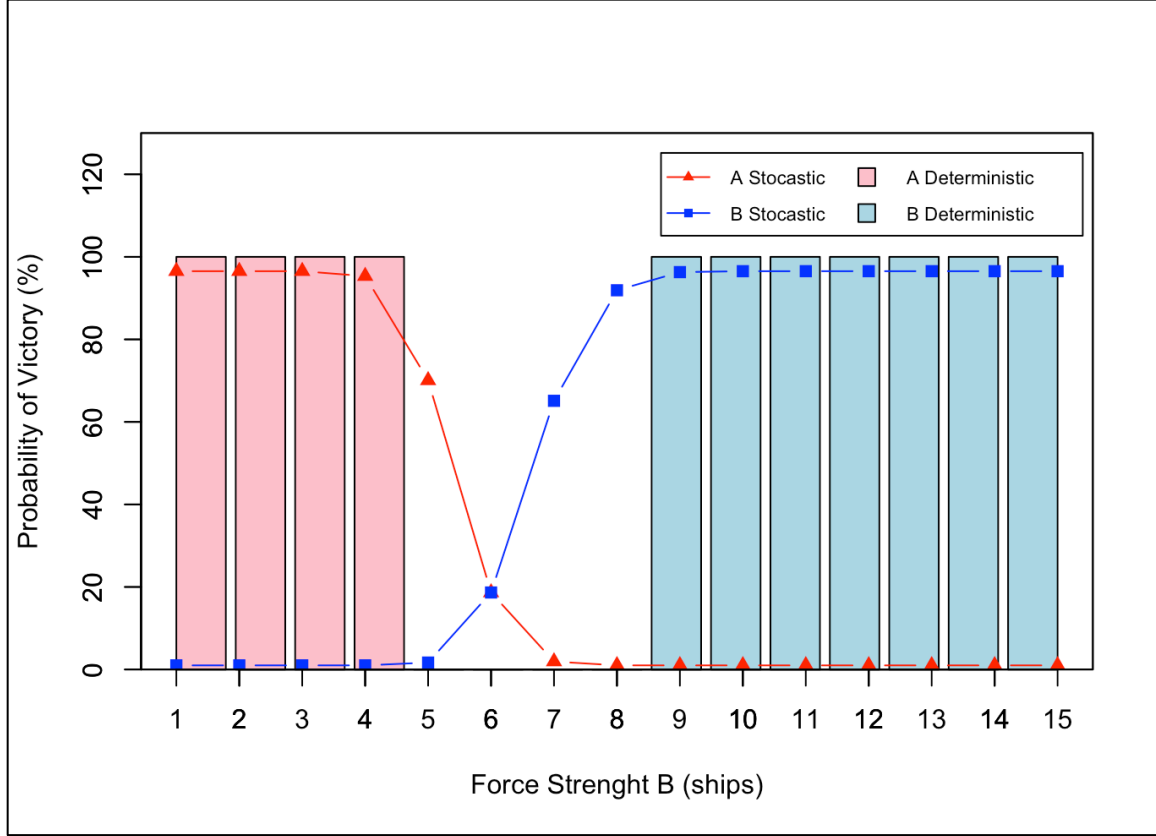


Figure 4. Probability of Victory for Different Size of B

According to Figure 4, the deterministic results show that when *side B*'s force level is between one and four the salvo is concluded with the total destruction of *side B* while some of *side A* remains (i.e., *side A* wins). When *side B*'s force level is nine or more, *side A* is annihilated while some of *side B* remains (i.e., *side B* wins). When *side B* is between five and eight, inclusive, both sides are destroyed, so there is no winner. These results are rigid and there is no variation in outcome.

On the other hand, the stochastic model gives a probability of victory for each side in every situation, even if the difference in the number of ships is large. For instance, when $A = 6$ and $B = 1$, the probability of victory for *side B* is 0.001. As the force sizes get closer to each other, the chances of winning become balanced.

In Figure 5, the standard deviation is presented for both sides as *side B*'s force level ranges from one to 15. As can be seen, the minimum of the standard deviations is maximized when the forces are at parity. Also, when one side dominates, the standard deviation of the losing side is small.



Figure 5. Standard Deviation for Different Ship Size B

Armstrong's (2005) conceptual stochastic salvo model may be impossible to solve in closed-form. To enable a closed-form solution, Armstrong uses simplifying approximations (e.g., using the normal approximation of the binomial). To assess the goodness of his assumptions, Armstrong (2011) compares his closed-form approximate solution, which was detailed above, to what is obtained by simulation of the conceptual model across 486 scenarios (or input combinations). He concludes that the approximations

“generally performed very well.” Each scenario was simulated 50,000 times in order to make his resulting estimates extremely precise.

C. CHAPTER SUMMARY

A brief background was provided for both the deterministic and stochastic naval salvo models. A demonstration of both models was presented with the numerical examples, and the results contrasted.

THIS PAGE INTENTIONALLY LEFT BLANK

III. LINEAR MODELS AND DESIGN OF EXPERIMENT

Chapter II detailed the two salvo models of naval surface combat that we are investigating, one stochastic and one deterministic. This chapter covers the primary statistical tools used in our study—regression models and design of experiment (DOE) methods.

A. LINEAR MODEL

This thesis uses linear regression models to build readily interpretable metamodels of the salvo equations’ responses as a function of their inputs. Linear models are a mainstay of statistical practice and are widely utilized in areas such as engineering, business, military, and the physical and social sciences.

A regression model describes a quantitative *response* (or *dependent variable*) in terms of various *predictors* (*independent variables*). Regression models are often used to predict future responses or to provide an understanding of the relationship between the response and the predictors. The latter use is our primary objective. The most common form of its mathematical expression is:

$$Y = f(X_1, X_2, \dots, X_n) + \varepsilon \quad (15)$$

In this form, “ f ” is a function to be determined that produces the best fitting response (i.e., “ Y ”) given inputs $X_1, X_2, \dots, X_n$. “ ε ” is an error term that is often assumed to be normally distributed. The function “ f ” can take an infinite number of forms, depending on the relationship being fit, but the most basic representation of “ f ” in a linear regression model is:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \varepsilon \quad (16)$$

where $\beta_i, i = 1, 2, 3, \dots, n$ are unknown constant parameters and β_0 is called the *intercept term* (Faraway, 2015a). The β_i parameters are usually estimated to minimize the sum of the squared differences between the observed and fitted responses given the data. In a linear

regression model, the fit is linear in the parameters, but not necessarily the input factors. For instance:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 x_3 + \beta_3 x_4^2 + \beta_4 \sqrt{x_5} + \beta_5 \log(x_6) + \varepsilon$$

is a linear a model, while:

$$Y = \beta_1 x_1^{\beta_2} + \varepsilon$$

is not. However, some nonlinear relationships can be transformed into linear relationships. By taking logs, the above nonlinear expression is turned into a linear model.

The term “linear” may be perceived as something simple by casual observers. However, a regression model can be transformed and combined in many ways to allow users to fit a variety of complex relationships (Faraway 2015b).

Another property of a regression model is that it is built using observations, so data is an important part. The source of observations may be actual systems, simulations, or both. The more data that is available, the better the underlying relationships can be determined. Since our data is generated from the salvo models, we want to conduct lots of experiments. However, the forms of the input data can limit the regression models that can be fit. For example, if a continuous input is sampled at only two levels, only a linear relationship can be fit. Therefore, academics have been seeking designs that provide the best factor settings for experiments since before computers were invented. Of course, “best” depends on the experimenter’s goals and the nature of the data and underlying relationships, and using design in a simulation context provides some unique challenges and opportunities. The next section provides some background on the design of experiment methodologies used in this simulation-based research.

B. DESIGN OF EXPERIMENTS

Design of experiments (DOE) is a crucial discipline for researchers aiming to describe or explain output under the variation of input. DOE enables us to efficiently vary inputs (called *factors*) to observe how outputs (called *responses*) are affected by them. Experimenters can have many goals. These may include identifying the most influential

factors and identifying interactions among them. These goals affect the designs that researchers should use.

1. Terminology and Notation of DOE

This part explains the basic terminology and notation of DOE that are used and mentioned in this study.

- A ***predictor*** is an input variable that is used for predicting outputs in a regression model.
- A ***response*** is a variable that is dependent on the predictors in a functional relationship (i.e., an output variable).
- A ***factor*** is an input variable that is varied by the experimenter. Each factor can potentially be represented as a predictor in a regression model.
- A ***level*** is a specific value that a factor may take.
- A ***design point (DP)*** is a unique combination of settings for all the factors.

2. Categories of Factors

In order to create good designs, it is essential to understand the factors' and predictors' categories according to their specification under four different dimensions (Sanchez, and Wan., 2012). Sanchez and Wan's dimensions are presented in the following sections.

a. Quantitative versus Qualitative

Quantitative factors consist of data giving numerical information about quantities; that is, information can be measured according to specific units and expressed as numbers (e.g., length, weight, speed, and counts). On the other hand, qualitative factors give information about a quality that is not naturally measured numerically (e.g., color). To ease calculations, numerical values can be assigned to code qualitative factors.

b. Discrete versus Continuous

This categorization is only feasible for quantitative factors. Discrete factors have a finite number of specified levels in any bounded interval. For example, the number of ships on a side is an excellent example of a discrete factor. Continuous factors are not restricted to distinct values, but can take any value within defined ranges. Between any two real numbers, there are an infinite number of values. For instance, the probability of successful launch of an offensive or defensive missile is a continuous value limited between zero and one. There are an infinite number of values that can be chosen for this factor. From among all possibilities, the DOE must specify which values to select in the experiments.

c. Binary versus Non-Binary

Binary factors can only take two values, such as success and failure or zero and one. Non-binary values can have more than two values, like the number of ASMs.

d. Controllable versus Uncontrollable

In real-life, some factors are under our control (i.e., they can be set), such as determining the number of ASMs to be loaded on a ship. Other factors cannot be controlled, such as the enemy's tactics or the weather. However, these factors can be specified and used as input variables to our models.

3. Experimental Designs

When designing an experiment, we must take into account factors and response characteristics in terms of those four categories. Here are descriptions of some commonly used designs.

a. Factorial Designs

Factorial designs are “gridded” samples at equally spaced points between minimum and maximum levels, either for quantitative factors or at the distinct levels of qualitative factors. They are often represented as $(\#levels)^{(\#factors)}$. For example, a 3^5 design consists of all 243 combinations of five factors at three levels each. An important property of factorial designs is that they provide the ability for the researcher to investigate interactions among

the predictors. In addition, factorial designs are orthogonal—which provides good statistical properties for fitting metamodels. Unfortunately, the number of DPs increases exponentially as the number of factors or levels increases.

(1) Coarse Grid Design 2^k

The simplest experimental design is a coarse grid design with two levels for each of k factors. Conceptually, the 2^k factorial design takes its samples from the upper and lower bounds of each factor. Unfortunately, for continuous factors, only a linear relationship can be fit.

(2) Finer Grid Design m^k

Examining each of the factors at only two levels (e.g., maximum and minimum) prohibits the analyst from obtaining data on the simulation’s behavior between the maximum and minimum values. Let us explain the situation with an example. Ancel Keys introduced *Body Mass Index* (BMI), whose formulation consists of the ratio of weight (in kg) to the square of height (in cm), in 1972 with an article titled *Indices of Relative Weight and Obesity*. He defined BMI as a metric that categorizes an individual as whether they are *underweight*, *normal weight*, *overweight*, or *obese*. Today, BMI is widely used by physicians to determine whether a person is healthy or is at risk in terms of weight.

Figure 6 illustrates the *space-filling* properties of three factorial designs made on BMI. The left plot pictures a two-level design for the factors weight and height, while the middle and right plots represent ten and hundred level designs, respectively. The dots show where the samples are taken. The colors show the response. For the 2^2 factorial design, we only see four outcomes and two response values. The 10^2 factorial design conveys much more information. The 100^2 factorial design has even higher resolution and enables us to understand the contours of BMI index. As we experiment on more levels, more information is forthcoming, but it requires more total experiments—in this case from four to 100 to 10,000 DPs.

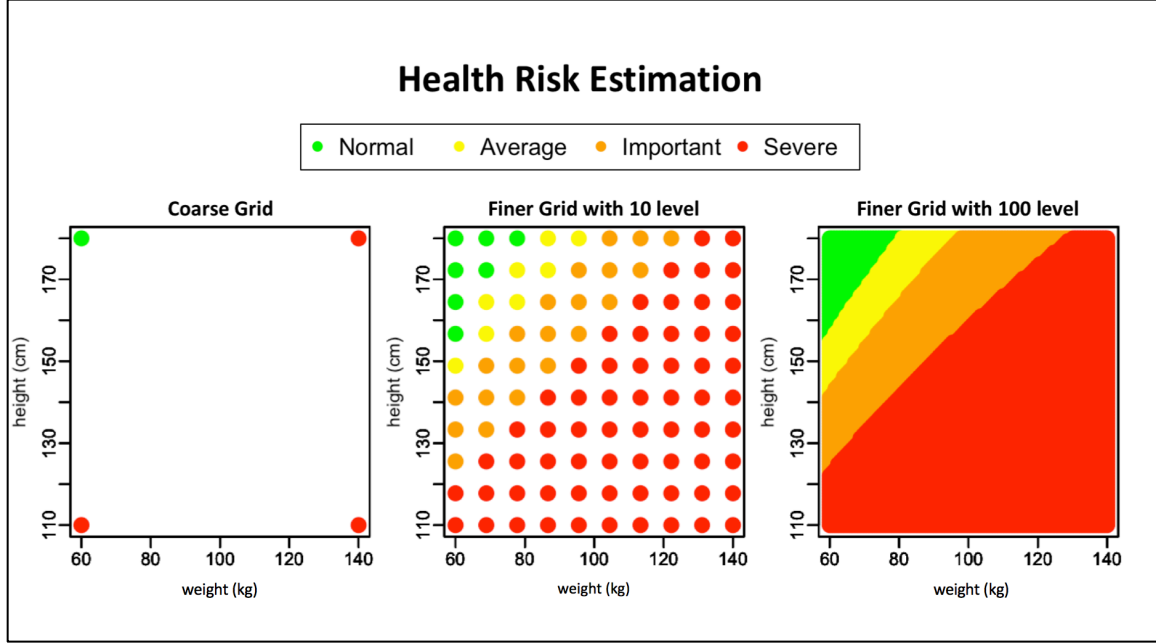


Figure 6. 2^2 , 10^2 , and 100^2 Factorial Designs for BMI

As the number of levels, m , increases, the design better fills the experiment space and provides finer detail for understanding the results. However, *fine grids* rapidly become infeasible as the number of factors grows. For instance, ten factors with 11 levels each requires more than 10 billion DPs in a full factorial design.

(3) *Fractional Factorial Design and Central Composite Designs*

In some situations, it may be appropriate to assume that high-order interactions are of less importance or even negligible. In such cases, the number of design points can be limited by using a fractional factorial or central composite design (Montgomery, 2005).

Figure 7 visually shows a comparison of factorial designs of various resolution and a central composite design. Fractional factorial design points capture every face of the cube with at least two points. Also, it is seen that the coarse grid with eight DPs and the fractional factorial design with four DPs comprise the same pairwise projections on the scatter plots. As the number of factors increases, the difference between the number of DPs needed for these two designs gets larger. For example, 100 factors with two levels produces 1.26×10^{30} DPs, which requires 205,000 years to be executed on the world's fastest computer for a

simulation consisting of a single elementary operation, while a low-resolution fractional factorial requires only 32,768 DPs (Sanchez and Sanchez, 2005).

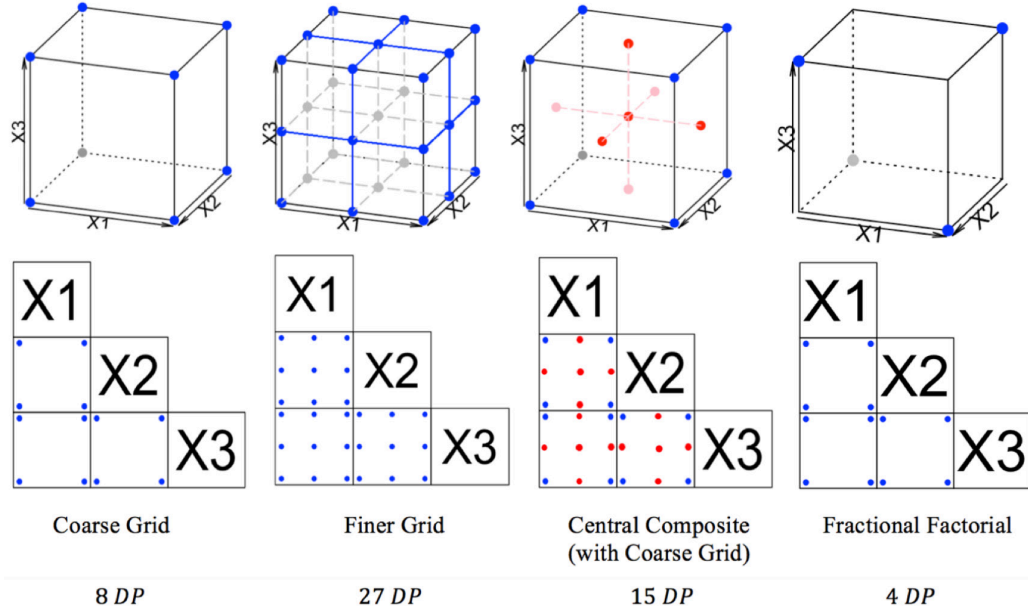


Figure 7. Representation of Factorial Designs with Three Factors.
Adapted from Sanchez and Wan (2012).

b. *Space-filling Designs*

Aside from a factorial design with a fine grid, there are other methods derived to efficiently sample from the interior of the design space. Perhaps the most widely used in simulation are *Latin Hypercube* (LH) designs. LHs uniformly spread the samples for each factor across its range.

Figure 8 provides a picture of a random LH design for the BMI example. Unlike 2^2 coarse grid designs, an LH provides much more detail between the upper and lower bounds with less samples required than for a 10^2 factorial design. This advantage is more pronounced in higher dimensions. Moreover, LHs enable users to fit a diverse set of metamodels on multiple different responses given the experimental data.

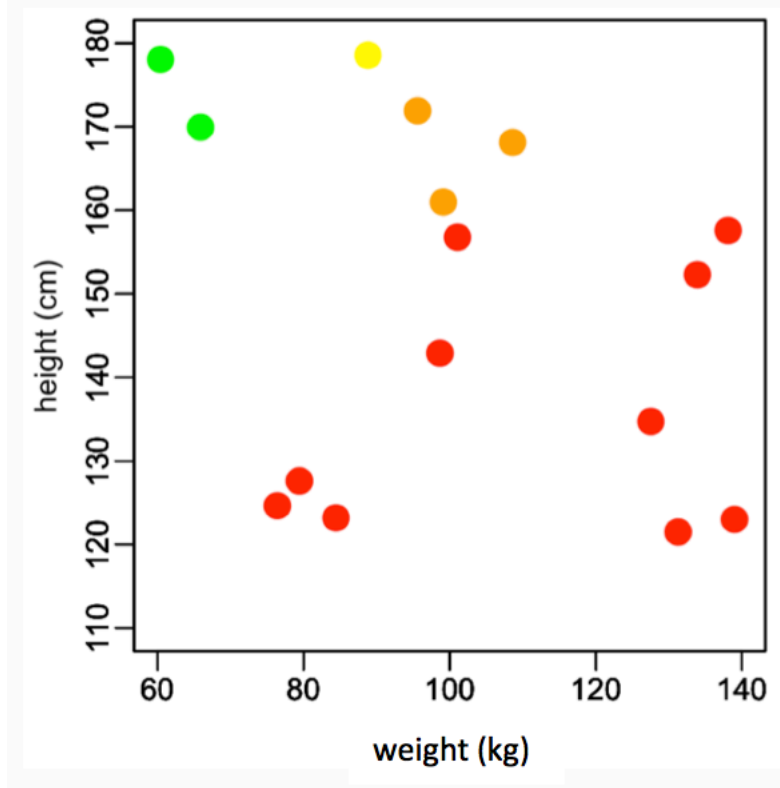


Figure 8. LH Design for BMI

However, random LHs do not necessarily perform well due to the randomness in their construction. With this motivation, Cioppa and Lucas (2007) developed an algorithm to construct *nearly orthogonal Latin hypercube* (NOLH) designs with good space-filling properties. Their NOLHs have been catalogued for designs up to 29 factors. For more extensive or finer NOLHs, see Hernandez et al. (2012) and MacCalman et al. (2017).

Figure 9 shows the pairwise projections of two different designs: a 4^4 on the left with 256 DPs, and a NOLH on the right with 257 DPs. As can be seen, the NOLH has much better space-filling. As the number of factors increases, the efficiency of NOLHs gets more significant when compared with finer grids. According to Sanchez and Wan (2012), in one of their studies: “Each replication of a 29-factor experiment would take under five minutes using a NOLH design, but over 17 years using a 2^{29} factorial design.”

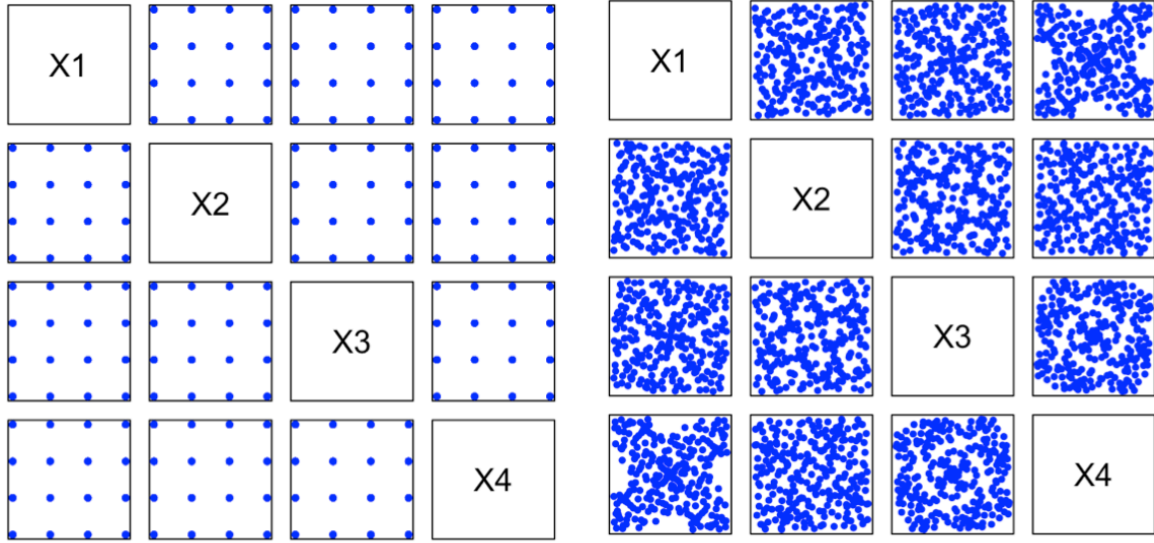


Figure 9. Scatterplot Matrices of 4^4 Factorial and NOLH.
Adapted from Sanchez and Wan (2012).

Nevertheless, when a factor consists of discrete values within a limited number of levels, a NOLH may no longer be the most effective way of designing the experiment. For that reason, Vieira Jr. et al. (2011a, 2011b, 2012) created the *nearly orthogonal and balanced* (NOB) method, enabling analysts to derive nearly orthogonal designs with 512 DPs for a mixture of discrete and continuous variables for up to 300 factors.

Figure 10 (from Sanchez and Wan, 2012) provides a comparison of some design techniques to give guidance to analysts. An updated version of this chart is available at Naval Postgraduate School's (NPS) SEED (simulation experiments and efficient designs) Center for Data Farming (<https://my.nps.edu/web/seed/welcome>).

After considering many designs, a NOB design was chosen for this thesis since the experiment is planned for factors with a limited number of discrete levels in ship numbers A and B . In addition, a NOB design provides 512 DPs, allowing many degrees of freedom for fitting metamodels.

In an experimental design, one should consider specifications of all input parameters. Many factors may have many different levels with different features, such as binary, discrete, or continuous. The main aim of the data farmer is to determine the best

design for their study. Designing an experiment is like cooking a meal. It is important to use the right ingredients in the right amounts.

	2^k factorials	m^k factorials, $3 \leq m \leq 5$	m^k factorials, $6 \leq m \leq 10$	R3FF, orthogonal arrays	Foldover designs	R4FF, Plackett-Burman designs	R5FF	Central composite with full factorial	Central composite with R5FF	random LH with $n \gg k$	smallest possible NOLH (i.e., very few extra columns)	larger NOLH	crossed NOLHs	FFCSB (main effects)	FFCSB-X or Hybrid method
FACTOR CHARACTERISTICS															
Total number of factors: 2-6	B*	L*													
Total number of factors: 7-10		1					B*								
Total number of factors: 11-29							B*								
Total number of factors: 30-99							2		2						
Total number of factors: 100-300							2		2						
Total number of factors: 300-1000															
Total number of factors: 1000-2000															
Binary factors	B*														
Qualitative factors with 3 or more levels		L*													
Discrete or continuous factors treated as binary	1			1	1	1	1								
Discrete factors, 3-5 levels of interest	1														
Continuous factors, or discrete with many levels															
Decision factors (controllable in real world)															
Noise factors (uncontrollable in real world)					4	4	4			5					2
RESPONSE CHARACTERISTICS															
Main effects only (initial screening)	2			1										1	
Main effects (valid w/ 2-way interactions exist)	2							2							
Main effects and all 2-way interactions	2														
Main effects and many interactions		2	2												
Quadratic effects		2	2												
Thresholds / non-smooth effects		2	2												
Flexible modeling - not all pre-specified		2	2												
OTHER CONSIDERATIONS															
Batch mode unavailable - all runs through GUI		2		3					3		3				
■ Provides additional modeling flexibility or allows some assumptions to be assessed B* Good design choice for binary factors L* Good design choice for factors with a limited number of qualitative or discrete levels C* Good design choice for continuous factors, discrete factors with many levels ● Works well 1 Assumes that interactions are negligible or that they'll show up with the main effects - must follow up with confirmation runs 2 For FFCSB-X, "many" means 2 or 3 levels 3 Smaller designs are the only ones feasible until this gets "fixed" - work with the developer 4 Design's correlation structure must be checked - stacking many designs may be an alternative 5 Degrees of freedom limit the number of terms that can be estimated simultaneously, so not all main effects and two-way interactions can be estimated simultaneously. ○ Consider these designs if additional computing resources are available ○1 These require many more runs than other designs unless k is small. Consider NOLH designs. ○2 Start with 2 replications and see if you can eliminate any factors - each time you do, you effectively double the number of replications for factors that remain. ○3 Same as above, but to avoid overly-large designs you may want to consider saturated or nearly-saturated NOLH ■ Potential designs that provide additional modeling flexibility or allow some assumptions to be assessed, but typically require many more design points ● Potential designs, but better designs exist for this purpose ○1 Unless used for initial screening, it may be a good idea to explore at least 3 levels ○2 These require many more runs than other designs unless k is small. Consider R5FF (for binary) or NOLH designs ○3 Easier to use a larger NOLH (if all factors are quantitative) or else cross a full factorial for factors with just a few levels with an NOLH ○4 Since you do not need to estimate interactions among noise factors, use a screening design like R3FF or a small NOLH ○5 In the spirit of keeping noise factor designs small, you might prefer an NOLH ○6 If you're interested in screening and want to keep the number of runs down, go for one of the smaller LH designs															

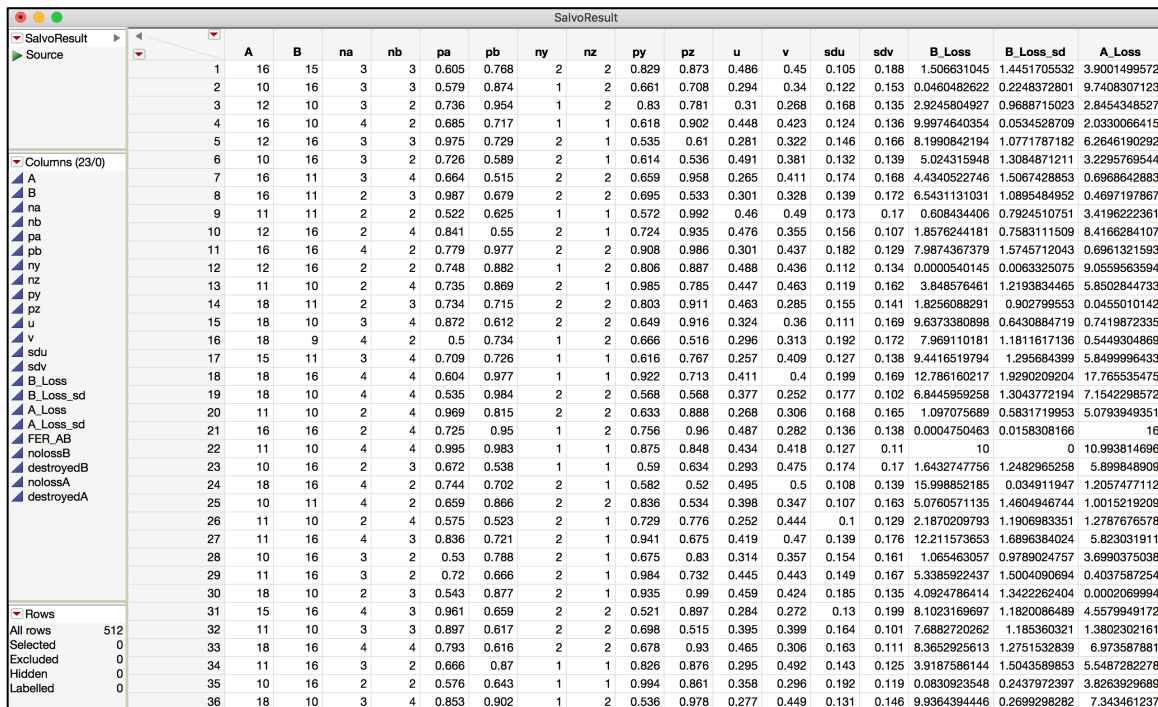
Figure 10. Design Comparison. Source: Sanchez and Wan (2012).

C. ANALYSIS TOOLS USED IN THIS RESEARCH

This part describes the analysis tools utilized in this research. There are three main analysis tools that this thesis uses: the NOB Mixed Design Worksheet for generating the design points to use as inputs for the simulation, R for designing the simulation algorithm, and R and JMP® Pro 14 for analyzing the outputs. Detailed information is provided in the following paragraphs.

1. JMP® Pro

JMP® Pro 14 (Figure 11) is a computer program used for statistical analysis as well as creating a model. JMP® Pro, developed by SAS Institute, offers statistical tools that are highly beneficial and easy to use without requiring advanced coding knowledge. It is widely used for the design of experiments and scientific research as well as for deriving histograms, summary statistics, plots and graphs, regression models, and partition trees.



	A	B	na	nb	pa	pb	ny	nz	py	pz	u	v	sdu	sdv	B_Loss	B_Loss_sd	A_Loss
1	16	15	3	3	0.605	0.768	2	2	0.829	0.873	0.486	0.45	0.105	0.188	1.506631045	1.4451705532	3.9001499572
2	10	16	3	3	0.579	0.874	1	2	0.661	0.708	0.294	0.34	0.122	0.153	0.0460482622	0.2248372801	9.7408307123
3	12	10	3	2	0.736	0.954	1	2	0.83	0.781	0.31	0.268	0.168	0.135	2.9245804927	0.9688715023	2.8454348527
4	16	10	4	2	0.685	0.717	1	1	0.618	0.902	0.448	0.423	0.124	0.136	9.9974640354	0.0534528709	2.0330066415
5	12	16	3	3	0.975	0.729	2	1	0.535	0.61	0.281	0.322	0.146	0.166	8.1990842194	1.0771787182	6.2646190292
6	10	16	3	2	0.726	0.589	2	1	0.614	0.536	0.491	0.381	0.132	0.139	5.024315948	1.3084871211	3.2295769544
7	16	11	3	4	0.664	0.515	2	2	0.659	0.958	0.265	0.411	0.174	0.168	4.4340522746	1.5067428853	0.6968642883
8	16	11	2	3	0.987	0.679	2	2	0.695	0.533	0.301	0.328	0.139	0.172	6.5431131031	1.0895484952	0.4697197867
9	11	11	2	2	0.522	0.625	1	1	0.572	0.992	0.46	0.49	0.173	0.17	0.608434406	0.7924510751	3.4196222361
10	12	16	2	4	0.841	0.55	2	1	0.724	0.935	0.476	0.355	0.156	0.107	1.8576244181	0.7583111509	8.4166284107
11	16	16	4	2	0.779	0.977	2	2	0.908	0.986	0.301	0.437	0.182	0.129	7.9874367379	1.5745712043	0.6961321593
12	12	16	2	2	0.748	0.882	1	2	0.806	0.887	0.488	0.436	0.112	0.134	0.0000540145	0.0063325075	9.0559563594
13	11	10	2	4	0.735	0.869	2	1	0.985	0.785	0.447	0.463	0.119	0.162	3.848576461	1.2193834465	5.8502844733
14	18	11	2	3	0.734	0.715	2	2	0.803	0.911	0.463	0.285	0.155	0.141	1.8256088291	0.902799553	0.0455010142
15	18	10	3	4	0.872	0.612	2	2	0.649	0.916	0.324	0.36	0.111	0.169	9.6373380898	0.6430884719	0.7419872335
16	18	9	4	2	0.5	0.734	1	2	0.666	0.516	0.296	0.313	0.192	0.172	7.969110181	1.1811617136	0.5449304869
17	15	11	3	4	0.709	0.726	1	1	0.616	0.767	0.257	0.409	0.127	0.138	9.4416519794	1.295684399	5.8499996433
18	18	16	4	4	0.604	0.977	1	1	0.922	0.713	0.411	0.4	0.199	0.169	12.786160217	1.9290209204	17.765535475
19	18	10	4	4	0.535	0.984	2	2	0.568	0.568	0.377	0.252	0.177	0.102	6.8445959258	1.3043772194	7.1542298572
20	11	10	2	4	0.969	0.815	2	2	0.633	0.888	0.268	0.306	0.168	0.165	1.097075689	0.5831719953	5.0793949351
21	16	16	2	4	0.725	0.95	1	2	0.756	0.96	0.487	0.282	0.136	0.138	0.0004750463	0.0158308166	16
22	11	10	4	4	0.995	0.983	1	1	0.875	0.848	0.434	0.418	0.127	0.11	10	0	10.993814696
23	10	16	2	3	0.672	0.538	1	1	0.59	0.634	0.293	0.475	0.174	0.17	1.6432747756	1.2482965258	5.899848909
24	18	16	4	2	0.744	0.702	2	1	0.582	0.52	0.495	0.5	0.108	0.139	15.998852185	0.034911947	1.2054777112
25	10	11	4	2	0.659	0.866	2	2	0.836	0.534	0.398	0.347	0.107	0.163	5.0760571135	1.4604946744	1.0015219209
26	11	10	2	4	0.575	0.523	2	1	0.729	0.776	0.252	0.444	0.1	0.129	2.1870209793	1.1906983351	1.2787676578
27	11	16	4	3	0.836	0.721	2	1	0.941	0.675	0.419	0.47	0.139	0.176	12.211573653	1.6896384024	5.823031911
28	10	16	3	2	0.53	0.788	2	1	0.675	0.83	0.314	0.357	0.154	0.161	1.065463057	0.9789024757	3.6990375038
29	11	16	3	2	0.72	0.666	2	1	0.984	0.732	0.445	0.443	0.149	0.167	5.3385922437	1.5004090694	4.0037587254
30	18	10	2	3	0.543	0.877	2	1	0.935	0.99	0.459	0.424	0.185	0.135	4.0924786414	1.3422262404	0.0002069994
31	15	16	4	3	0.961	0.659	2	2	0.521	0.897	0.284	0.272	0.13	0.199	8.1023169697	1.1820086489	4.5579949172
32	11	10	3	3	0.897	0.617	2	2	0.698	0.515	0.395	0.399	0.164	0.101	7.6882720262	1.185360321	1.3802302161
33	18	16	4	4	0.793	0.616	2	2	0.678	0.93	0.465	0.306	0.163	0.111	8.3652925613	1.2751532839	6.973587881
34	11	16	3	2	0.666	0.87	1	1	0.826	0.876	0.295	0.492	0.143	0.125	3.9187586144	1.5043589853	5.5487282278
35	10	16	2	2	0.576	0.643	1	1	0.994	0.861	0.358	0.296	0.192	0.119	0.0830923548	0.2437972397	3.8263929689
36	18	10	3	4	0.853	0.902	1	2	0.536	0.978	0.277	0.449	0.131	0.146	9.9364394446	0.2699298282	7.343461237

Figure 11. Screenshot of JMP® Pro 14

2. R

R, supported by the R Foundation for Statistical Computing, is a free-source programming language used for data mining and statistical analysis. R and its libraries have a vast domain of applicability, including classic statistical models, time series, linear or non-linear regression modeling, clustering, and classification. It can be used without writing pages of code. In addition to that, with its high-powered graphics the user can easily produce publication-quality graphs. Beyond the static graphics, interactive graphics are also available with additional libraries. In this thesis, R programming is used with RStudio (Figure 12), which is a development environment for R.

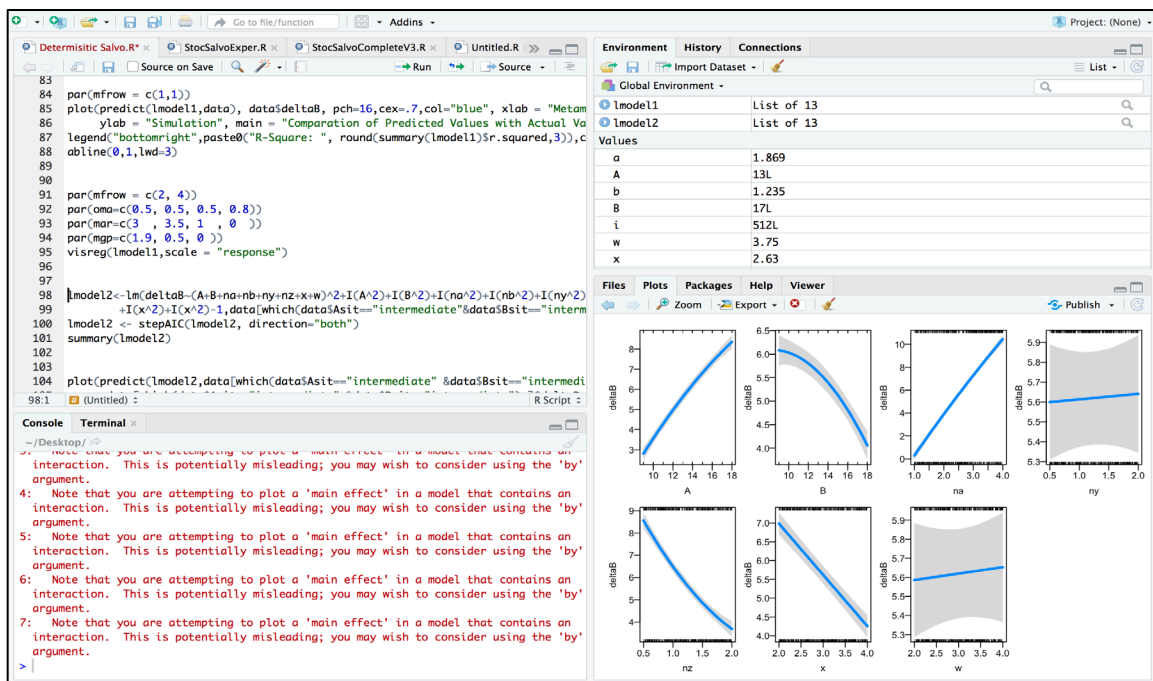


Figure 12. Screenshot of RStudio

3. NOB Mixed Design Worksheet

NOB Mixed Design Worksheet is a Microsoft® Excel spreadsheet (Figure 13) that contains 512 DPs for a total of up to 300 factors (200 discrete with between 2 and 11 levels and 100 continuous factors). The maximum absolute value of the correlation between any pair of columns in the design is 3.56%.

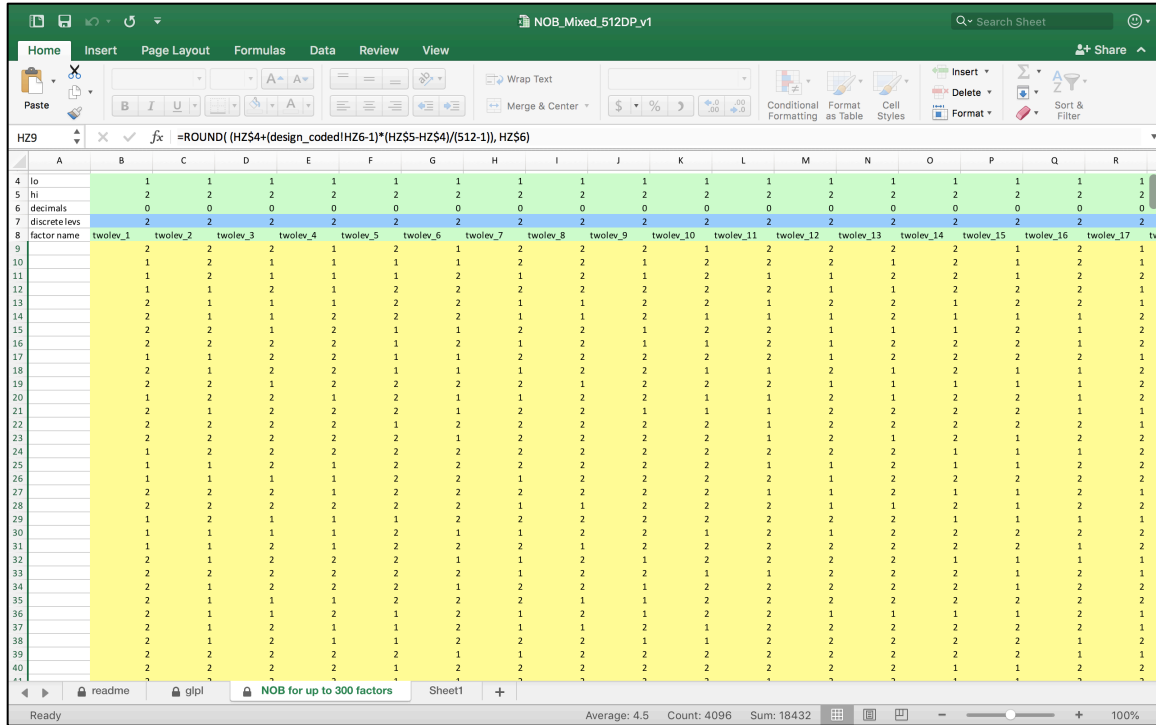


Figure 13. Screenshot of NOB Mixed Design Worksheet

D. CHAPTER SUMMARY

This chapter has provided a basic introduction to the experimental design methods that are used in this thesis. In addition, regression models that are utilized to interpret the results have also been presented. In the next two chapters, NOBs are used to determine the input setting of factors for data farming, and linear models and partition trees are utilized to build metamodels to facilitate understanding of both the deterministic and stochastic salvo models.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. ANALYSIS PART 1: DETERMINISTIC SALVO MODEL

The first three chapters explained the goals of this research, provided an overview of the two salvo models (deterministic and the stochastic), and introduced some of the analysis methodologies that are used. This knowledge is now blended to experiment on the salvo models with data farming. Our analysis part consists of three main sections. These are *designing the experiment*, *summary output analysis*, and *regression and partition tree metamodel analysis*.

A. DESIGNING THE EXPERIMENT

The design of the experiments is a crucial part of this study. When the results gathered from the experiments are analyzed, any fault made in this part may result in misleading conclusions. Therefore, tools like scatterplots and correlation matrixes are utilized to confirm that the experiment is conducted correctly.

In designing the experiment, there are three subparts: *Experimental Flow* gives information about how the experiments are conducted, *Data Farming* explains the design used in creating the data, and *Analysis of the Design* examines the DPs.

1. Experimental Flow

The computation is done in the R environment and it evaluates Hughes's salvo model over 5,120 DPs. A flowchart of the computation is presented in Figure 14, and the code is available in Appendix A.

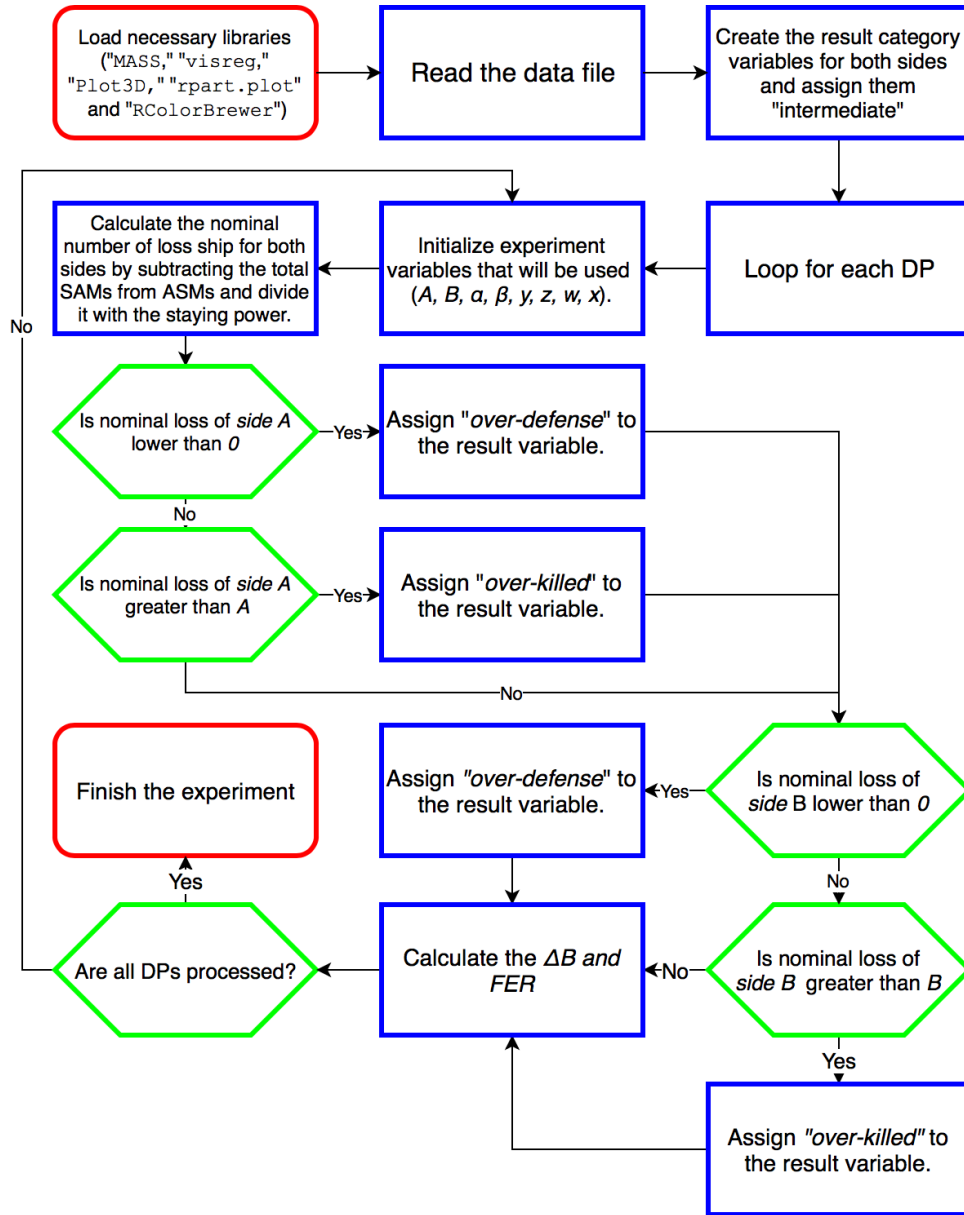


Figure 14. Flow Chart of the Deterministic Experiments

2. Data Farming

We need to determine the ranges of the variables to explore. In this experiment, DPs are selected with the idea of avoiding excessive instability (related to Hughes's salvo model, as explained in Chapter II) and over ranges similar to those of Armstrong (2011).

For the deterministic salvo model, there are eight predictors that are discrete in real life applications (though staying power is more complicated). However, in this study, since

the aggregate mathematical model allows, all of the factors are treated as continuous except for the ship number of each side. With this flexibility, the filling-space property of the mixed NOB design will be observable and the results will be of higher resolution.

As provided in Table 4, ship size for both sides is determined as an integer between 9 and 18, inclusive, which makes for 100 different combinations. The number of ASMs and SAMs per ship per salvo are respectively set to 3001 and 1501 different levels. After adding 2001 different values for the staying power, a crossed design would yield 8.12×10^{21} possible DPs. Out of that vast set of potential DPs, the “mixed NOB design” spreadsheet obtained from the SEED Center website (<https://my.nps.edu/web/seed/software-downloads>) utilizes 512 carefully chosen DPs. However, we appropriately rotate and stack the design nine times to get a total of 5120 DPs. This provides for more space-filling of the designated experimental region and increases the degrees of freedom available in fitting metamodels. Hughes salvo model is evaluated at each of these 5120 DPs using R.

Table 4. Mixed NOB Design for Deterministic Salvo Model

#	Factor	Explanation	Low	Inc.	High	Levels
1	A	Number of ship that individual side has in the beginning	9	1	18	10
2	B					
3	α	Maximum # of ASMs per ship per salvo	1.000	0.001	4.000	3001
4	β					
5	y	Maximum # of SAMs per ship per salvo	0.500	0.001	2.000	1501
6	z					
7	w	Staying power per unit	2.000	0.001	4.000	2001
8	x					
8.12x10 ²¹ DPs						

3. Analysis of the Design

Scatter plots and correlation matrixes evaluate the design in visual and mathematical ways. For that reason, a pairwise analysis is conducted for each pair of factors by generating a scatter plot and correlation matrix. The results for the stacked mixed NOB design are presented in Figures 15 and 16.

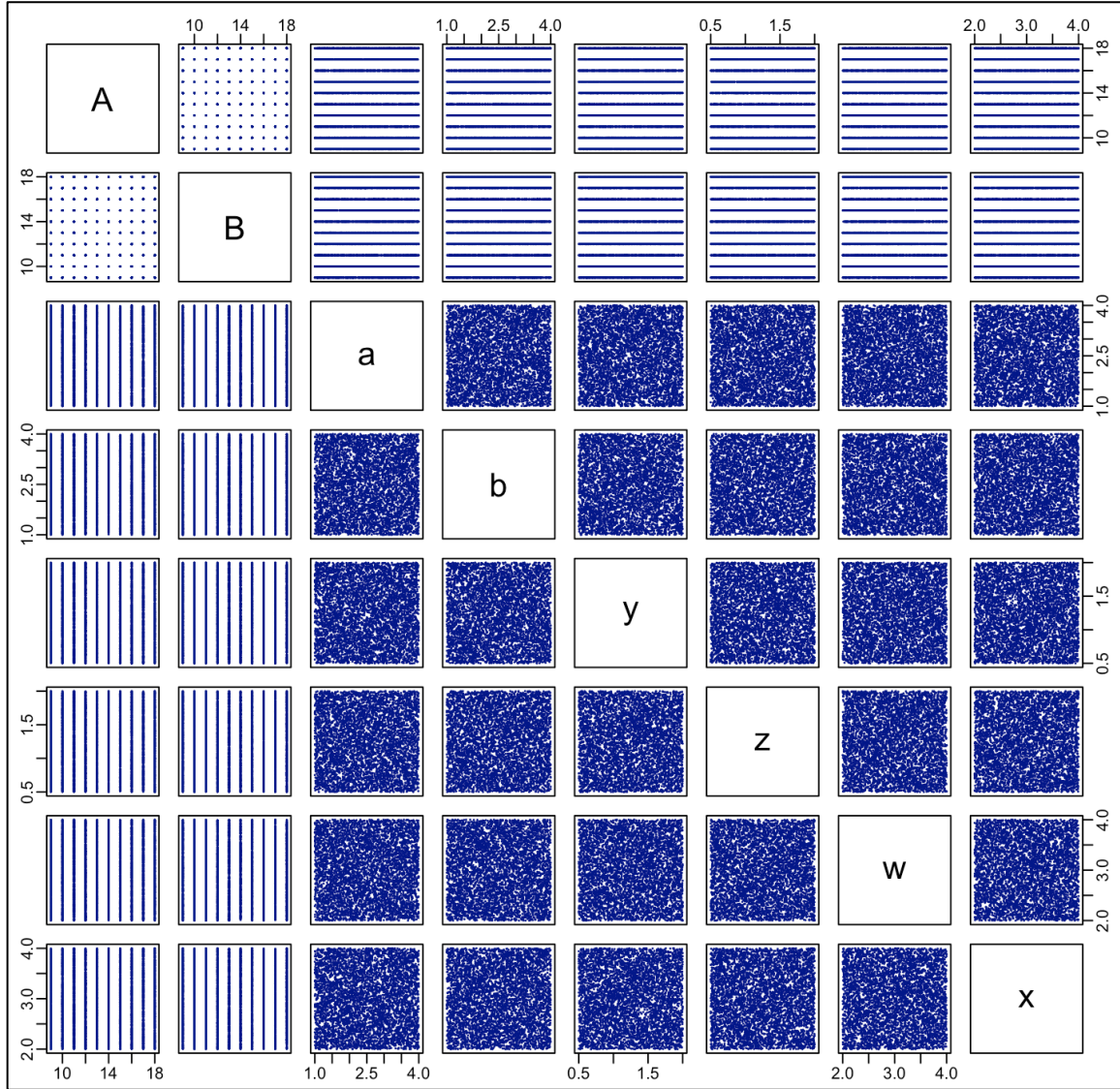


Figure 15. Pairwise Projections of the DPs for the Deterministic Salvo Experiment

The space-filling property of the mixed NOB design is illustrated in Figure 15. As can easily be understood from the scatter plot, the mixed NOB design generates sets of DPs that let the analyst investigate the full range of the experimental region. This provides flexibility to build metamodels involving higher order terms if needed. In this study, we focus on quadratic and interaction terms in fitting a regression metamodel that aims to closely match the response.

	A	B	a	b	y	z	w	x
A	1.000	-0.005	-0.005	-0.004	0.000	0.000	-0.002	0.000
B	-0.005	1.000	-0.003	0.000	0.000	0.001	0.002	-0.001
a	-0.005	-0.003	1.000	-0.002	-0.004	-0.003	0.004	0.002
b	-0.004	0.000	-0.002	1.000	0.000	0.001	0.003	0.001
y	0.000	0.000	-0.004	0.000	1.000	-0.001	0.002	0.003
z	0.000	0.001	-0.003	0.001	-0.001	1.000	-0.003	0.001
w	-0.002	0.002	0.004	0.003	0.002	-0.003	1.000	-0.003
x	0.000	-0.001	0.002	0.001	0.003	0.001	-0.003	1.000

Figure 16. Correlation Matrix for the Factors in the Deterministic Salvo Experiment

The correlation matrix provides information about the factors' linear relationships for all pairs of factors. To avoid confounding, it is important that the predictors should not have excessive multicollinearity. Figure 16 shows that among all pairs, the lower and upper boundaries of the pairwise correlations are between -0.005 and 0.004 . This ensures that all of the main effect estimates will be nearly independent of each other.

B. SUMMARY OUTPUT ANALYSIS OF HUGHES'S DETERMINISTIC SALVO EXPERIMENTS

In this analysis, each DP was run in R just a single time because of the deterministic nature of the model. Data and the detailed descriptions will be available at the author's GitHub profile (<https://github.com/gokhankesler/salvo-equations>) after the publication of this thesis.

The next step is to investigate four outputs (i.e., responses) of the model. These are the *result category* (*intermediate*, *over-destroyed* and *over-defense*), the *actual number of*

ship loss (ΔB), ship loss ratio ($\Delta B/B$), and FER ($[\Delta B/B]/[\Delta A/A]$). Since the deterministic salvo equations and the experimental design space for each side are symmetric, only the outputs of *side B* are examined. By using JMP[®] Pro 14, summary statistics of the experimental results and the histograms of the outputs to be studied are presented in Figure 17.



Figure 17. Summary Output Distributions from the Deterministic Salvo Model

Figure 17 shows the histograms and statistical results of our four outputs from the 5120 experiments. There is one qualitative response: the result category in the first row. The three quantitative responses are: the actual number of ship loss in the second row, the loss ratio in the third row, and *FER* in the fourth row. The number of observations is 5120 for each response except for *FER*. 700 observations of *FER* resulted in positive infinity (the ratio of any positive number to zero). For that reason, the number of observations is stated as 4420 in the summary statistics for *FER*.

For the result category, it can be seen that 3716 salvo exchanges resulted in the partial loss of *side B*'s force, while 717 and 687 salvo exchanges resulted in no loss and complete destruction, respectively. This information is highly utilized during the study of this thesis. That is the reason why this information is provided.

The histogram of the actual number of ship loss (ΔB) reveals that the distribution is right-skewed. A Shapiro-Wilk normality test (Wackerly, Mendenhall, and Scheaffer, 2007) confirms that the distribution is not normal, with a p-value less than 1×10^{-15} . The actual number of ship losses ranges from zero to 18, with a mean of 5.52 and a standard deviation of 4.25.

The loss ratio is derived from the ratio of ship loss to the starting ship size at the beginning of the battle (i.e., loss ratio = $\Delta B/B$). Loss ratio is the proportion of the force lost in the exchange, and ranges from zero to one. It has a mean and standard deviation of 0.44 and 0.34, respectively. Loss ratio is also not normally distributed, as confirmed by the Shapiro-Wilk normality test having a p-value equaling less than 1×10^{-15} .

For *FER*, it also has a right-skewed distribution. It is important to emphasize that although the maximum value of *FER* (excluding the observations where *FER* = infinity) is 3086, the histogram is intentionally limited to between zero and 10 to better see the main portion of the distribution. There are 4420 feasible (non-infinite) observations whose 95th percentile is about 10.54. In the regression modeling phase, there is an additional study on *FER* to determine the range that is used for metamodeling.

C. REGRESSION AND PARTITION MODEL ANALYSIS

In our regression model analysis, the input factors for the deterministic salvo equation are used as predictors of the outputs (responses) of the experiments. Regression metamodels are fit for three different responses: the *actual number of ship loss on side B*, *loss ratio of B*, and the *fractional exchange ratio (FER)*. Partition trees are also fit on these responses.

1. The Actual Number of Ship Loss

The regression metamodel was constructed using stepwise regression with all main effects, quadratics, and first order interactions considered as potential candidates. The final model was chosen as the subset that minimized the Akaike Information Criterion (AIC), see Akaike (1974). It is essential to keep in mind that the residuals are not normally distributed. Nonetheless, the metamodel is the best least squares fit and can be used for understanding the underlying relationships. However, the p-values associated with the regression coefficients won't be precise. This thesis heavily uses R^2 to measure the goodness of the metamodel fits. R^2 is the proportion of the variance of the response explained by the fitted model, it ranges from zero to one. An R^2 of one means the metamodel perfectly fits the data.

The chosen regression metamodel includes 27 terms and has an R^2 of 0.971. That is, the second order polynomial metamodels explains over 97% of the variance in ship loss. See Appendix C for R's regression output. A comparison of the metamodel's predicted values versus the actual values of ΔB is presented in Figure 18.

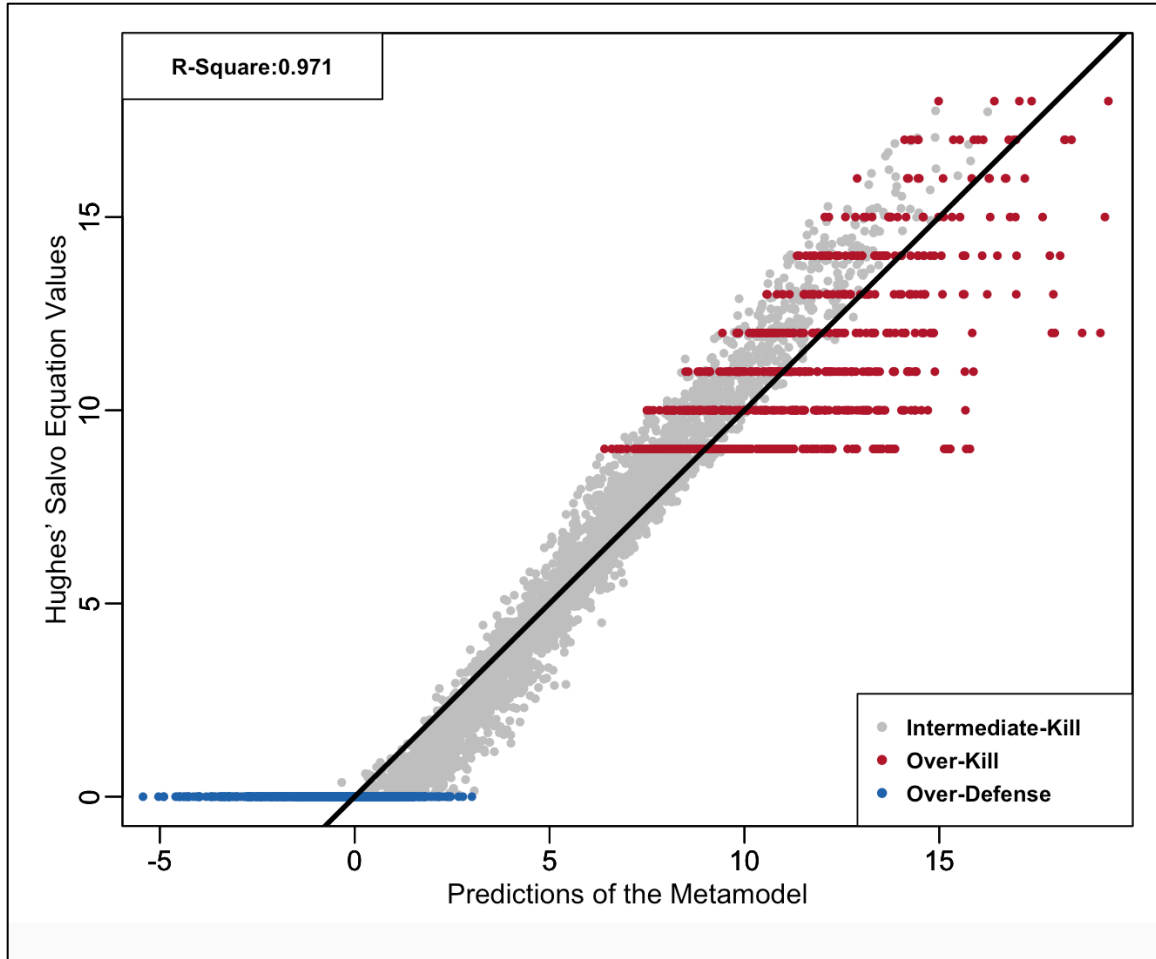


Figure 18. Predicted ΔB Values from the Metamodel versus the Actual Values Obtained by the Experiments.

Figure 18 displays the comparison of the results obtained from the salvo model evaluations and the predictions of the first regression modeling attempt. The results are plotted as three different categories: no loss (*over-defense*) in blue, total destruction (*over-kill*) in red, and partial kill (*intermediate-kill*) in grey. The figure has an equality line to show the ideal location where each point should be placed in a perfect fit in order to help the reader visually evaluate how well the regression metamodel matches the experimental results. It is seen (in Figure 18) that the metamodel performs worse in *over-defense* and *over-kill* situations. In addition, this causes the *intermediate* results to be underestimated for high ΔB values while overestimating for low ΔB values.

To improve the regression model fit and confirm the assumptions made above, the next step is to refit a regression model to the data in which both sides have *intermediate-kill* and limit the results to between zero and B . There are a total of 2792 DPs were this occurs.

After refitting a stepwise regression model, it is observed that the second model matches the actual deterministic salvo model much better than the first model, with an R^2 equaling 0.999. Figure 19 shows how the model predicts all 5120 DPs, with a much better fit. The inaccuracies caused by *over-defense* and *over-kill* results are eliminated when they are not used in the fit.

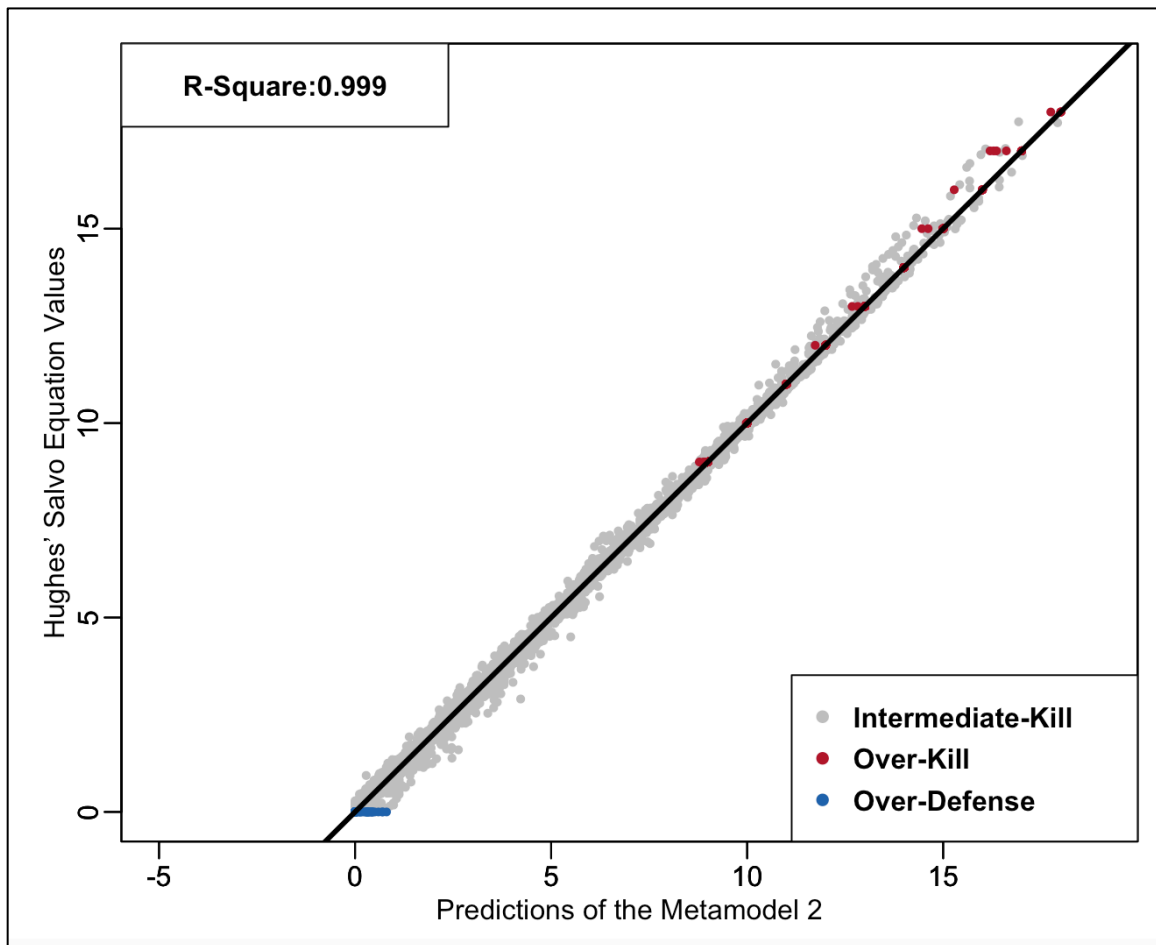


Figure 19. Predicted ΔB Values from the Metamodel Versus the Actual Values Obtained by the Experiments for the Intermediate Cases

For that reason, the second model is the final regression metamodel on ΔB developed in this research. Equation 17 displays the calculation of the *actual number of ship loss on side B* from our second metamodel. It is important to emphasize that these linear, quadratic, and interaction terms are readily interpretable. Moreover, Equation 17 can easily be coded into a spreadsheet and applies exceedingly well over our experimental region.

$$\begin{aligned} \Delta B = & A * (0.854 - 0.003A - 0.265x) \quad (17) \\ & + B * (-0.412 + 0.003A + 0.007\alpha + -0.335z + 0.117x) \\ & + \alpha * (4.671 + 0.325A - 0.032\alpha - 1.466x) \\ & + \beta * (0.021 + 0.016y - 0.012x) \\ & + y * (-0.040) \\ & + z * (-4.881 + 0.018A + 0.061\alpha + 1.452x) \\ & + w * (-0.056 + 0.019x) \\ & + x * (-1.966 + 0.642x) \end{aligned}$$

$$\text{where, } 0 \leq \Delta B \leq B$$

Equation 17 provides the coefficients of the main effect terms, their second order quadratic terms, and the interactions between them from our experimental region. When only the coefficients of the main effect terms are individually considered: α and A increase *side B's* ship loss. That is, more *side A* ships and *offensive power* tends to result in more *side B* ships taken out of action. This is an intuitive result. Conversely, z , x , and B all have a decreasing effect. That is, fewer *side B* ships tend to be lost when there are more of them and they have greater *defensive power* and *staying power*. Again, this finding is what one would expect. A single unit increase in α causes 4.67 more ships to be lost on average (the linear term by itself). On the other hand, a single unit increase in z prevents 4.88 ships from being destroyed on average (again, the linear term by itself). Other predictors, such as “ β ,” “ y ,” and “ w ” also have effects on ΔB , although they are not as significant as the previous

parameters. To confirm the interpretations made from Equation 17, we plot the prediction profiler, it shows the relationships of the individual factors on the response ΔB .

The prediction profiler shown in Figure 20 gives better insight on the effects of the independent variables on the number of ship losses for *side B*. According to the figure, predictors ‘ y ,’ ‘ β ’ and ‘ w ’ have little effect and therefore can be discounted when considering ship loss on *side B*, which is compatible with the mathematical equation above. Ship number A and the number of ASMs α that can be used from each ship on *side A* have a strong nearly linear increasing effect on the number of ships *side B* losses. Conversely, the number of SAMs β , staying power x , and ship size B , all have a strong decreasing effect on *side B*’s losses. The prediction profilers hold all except the graphed variable constant, and thus do not show or account for interactions.

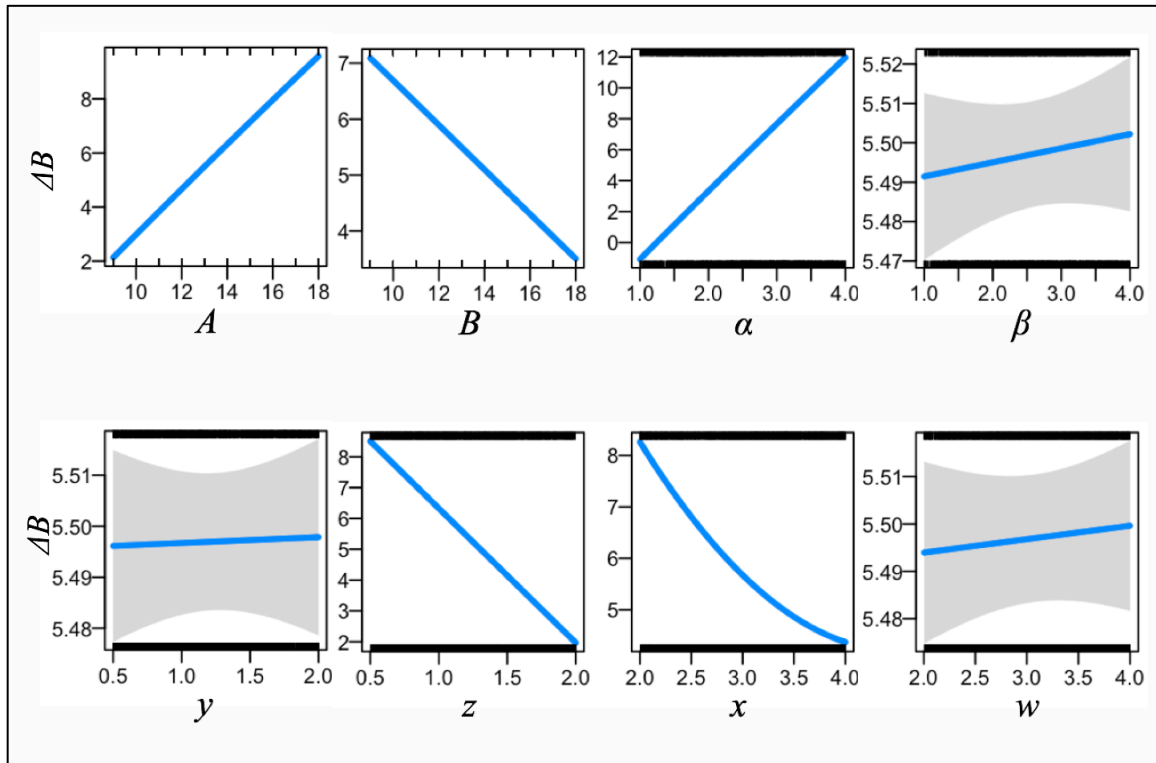


Figure 20. Prediction Profiler of ΔB (Deterministic)

Heat maps are a visual device that show the effects of predictors' interactions. Figure 21 displays a heat map of ΔB for the pairwise combinations of the five most influential factors. The coloration is limited to between zero (blue) and 18 (red). The stationary points (i.e., the values that are set fixed while two other factors are varied) are stated below the name of the variables ($A, B = 11.5, \alpha, \beta = 2.4 = 2.4, y, z = 1.25$, and $w, x = 3$). Red areas show a high loss for *side B*. To give a specific example, an increase in both predictors A and α causes a dramatic increasing in ship loss for *side B*.

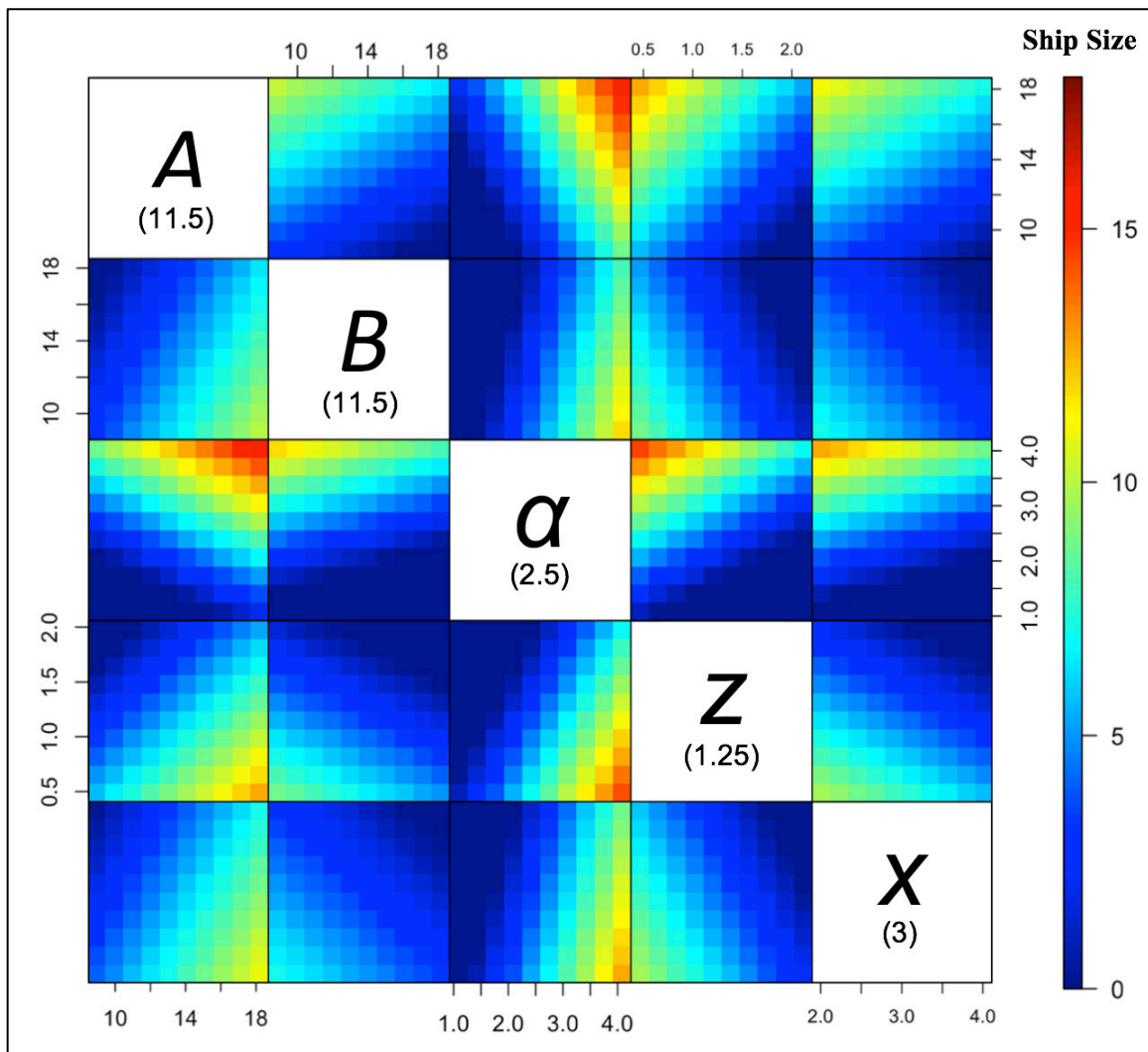


Figure 21. Heat Map of ΔB (Deterministic)

Another useful tool for analyzing the data is a partition tree diagram (Rokach and Maimon, 2008). Figure 22 shows a tree diagram of ship loss on *side B* (i.e., ΔB). It gives the possible outcomes as a proportion of the considered observations as some factors are classified. To give an example from the graph, in the designed experiment space, the expected number of ship losses for *side B* is 5.5 overall. If the number of ASMs per ship per salvo (α) is less than 2.6, then the expected number of ship losses for *side B* is 2.9, and this includes 52% of the data. If, in addition, α is less than 1.7 and z is greater than .97, the mean ship loss for *side B* is 0.77, this occurs in 17% of the data. At the other extreme, if $A \geq 13$ and $\alpha \geq 3.2$ and $x < 2.8$ and $B \geq 12$, then the mean ship loss for *side B* is 13. This combination of inputs occurs in 5% of our data.

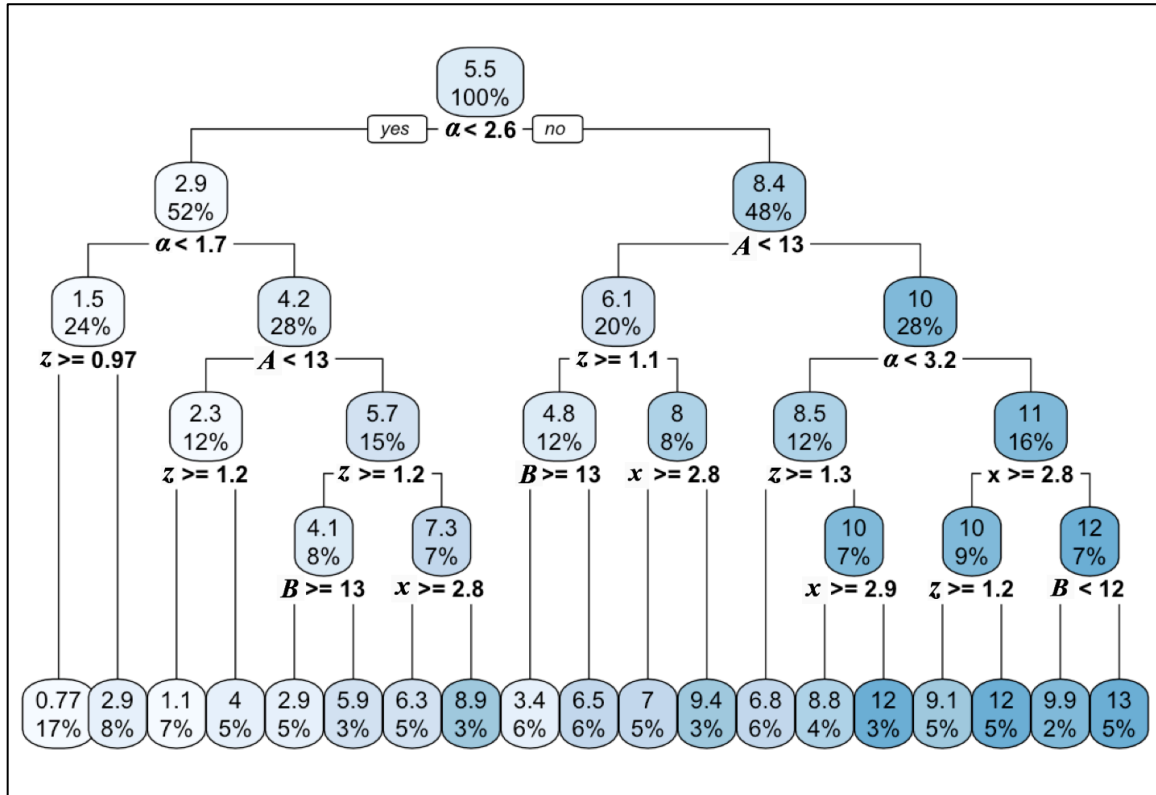


Figure 22. Tree Diagram of ΔB (Deterministic)

2. Loss Ratio

For the loss ratio analysis, the same steps as above are applied. A stepwise regression model of loss ratio with its parameters and their quadratic terms results in an R^2 of 0.974. The R summary of the model is available in Appendix C. The comparison of predicted values versus the actual values of $\Delta B/B$ is presented in Figure 23. This graph shows the same behavior as the previous regression metamodel on the actual number of ship loss on *side B*. The feasible region is limited between zero and one. However, the metamodel sometimes estimates values more than one and lower than zero. This is a property of linear models. For that reason, in the next step, the model is refit with only the results that are labeled as *intermediate*. After fitting a stepwise regression model and limiting the results to between zero and one, the model predicts much more accurately. A scatter plot of the comparison, which is plotted in Figure 24, confirms these findings, as the R^2 value increases to .998.

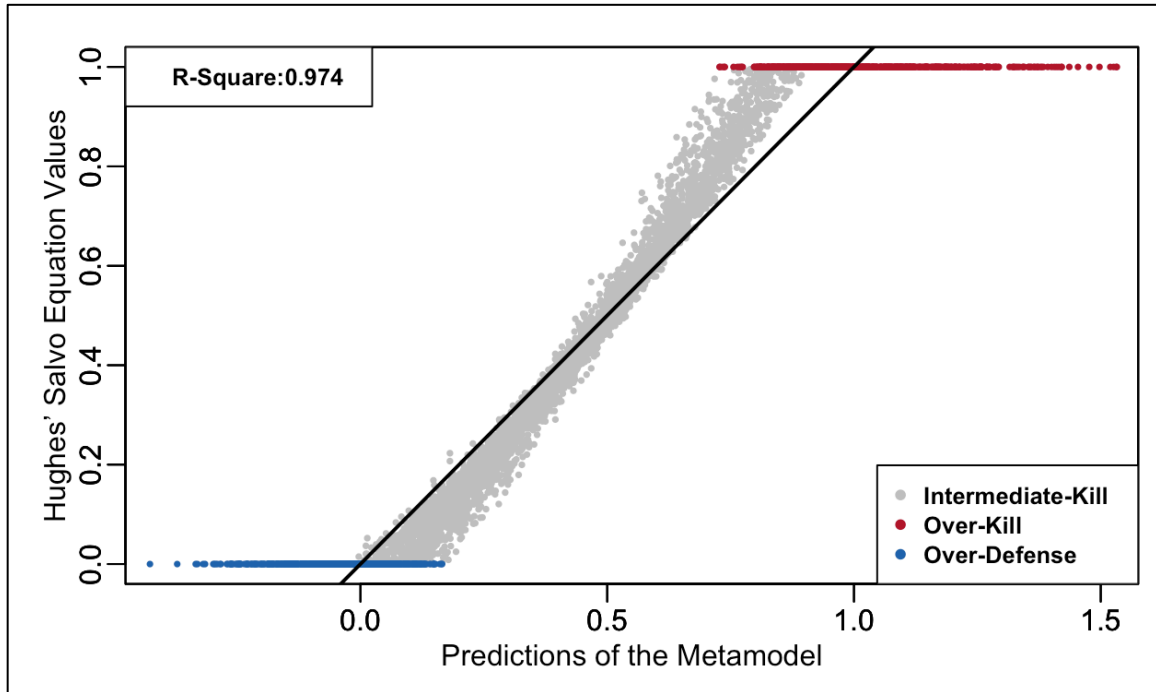


Figure 23. Predicted $\Delta B/B$ Values Versus the Actual Values Obtained by the Experiments

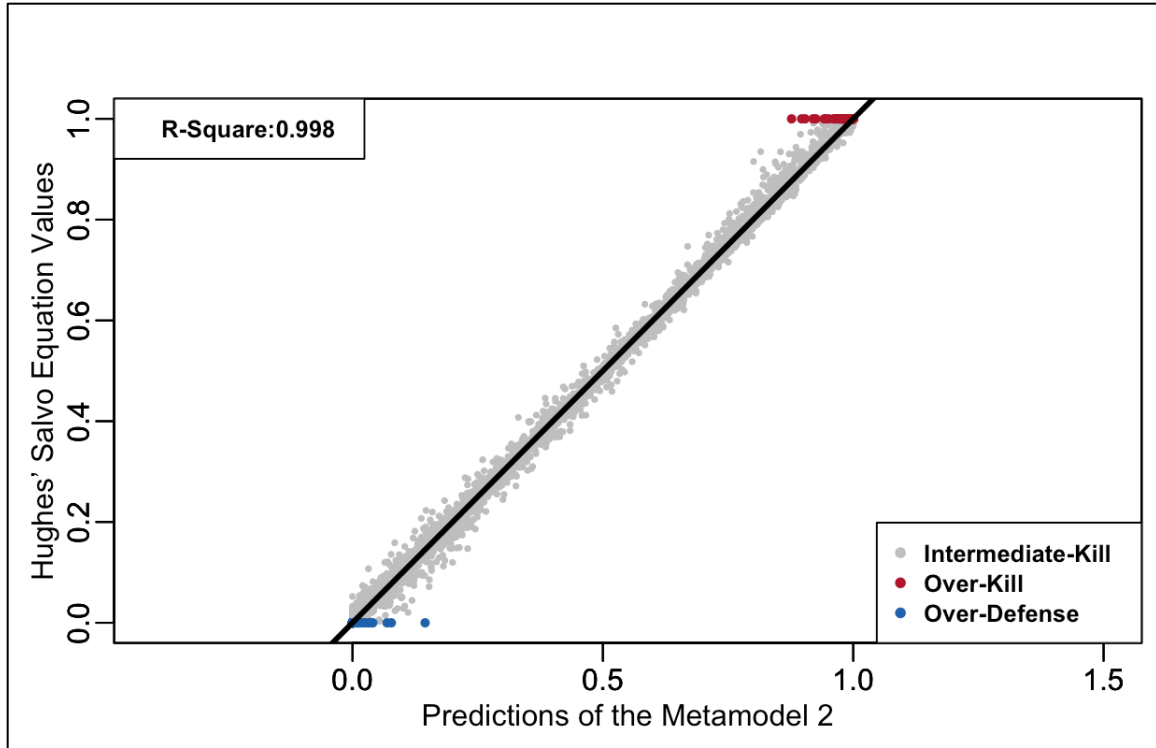


Figure 24. Predicted $\Delta B/B$ Values Versus the Actual Values Obtained by the Experiments for the Intermediate Cases.

After refitting the second regression metamodel, it is observed that it fits much better, with an R^2 of 0.998. Figure 24 shows the goodness of the predictions when the *over-defense* and *over-kill* results are compensated for. Note: all 5120 DPs are displayed. For that reason, the second model is the predictor model for loss ratio. The next phase is to investigate the response behavior as the factors vary.

The prediction profiler of loss ratio is shown in Figure 25. As with ship loss, it shows that the predictors related to the *offensive power* of *side A* and the predictors related to *defensive power* of *side B* have increasing and decreasing effects on loss ratio, respectively. The other parameters, y , β and w , have minimal effects and can be discounted with respect to loss ratio on *side B*. This finding is compatible with the mathematical equation.

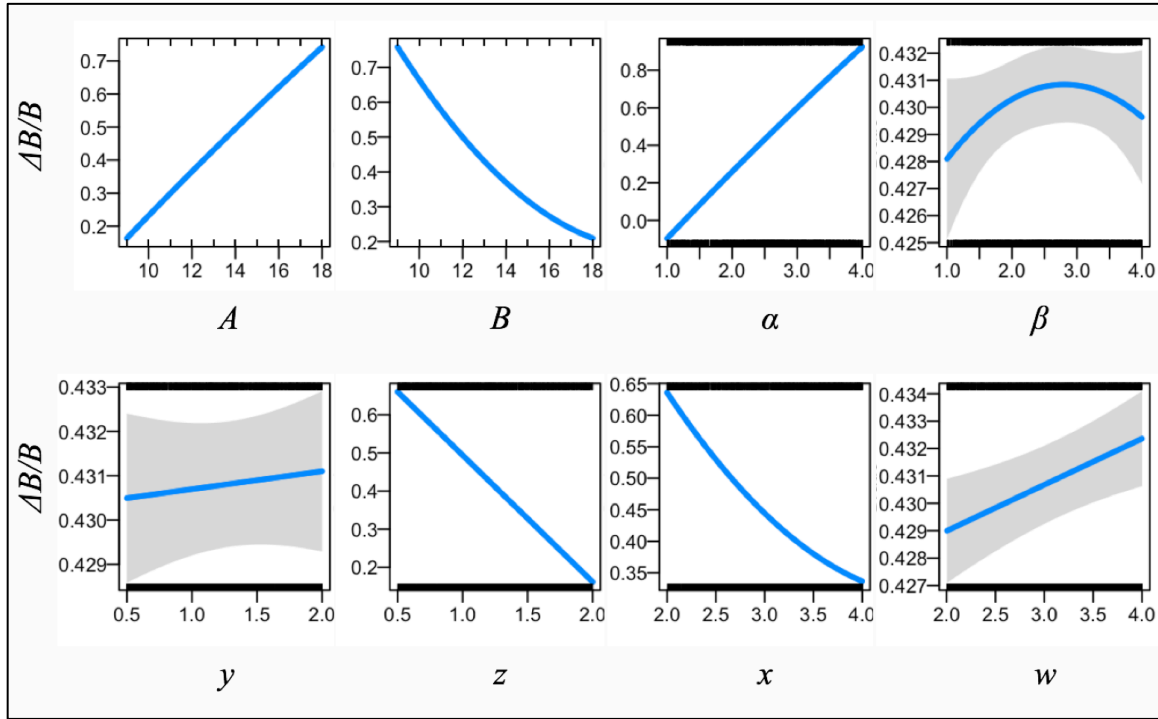


Figure 25. Prediction Profiler of the Loss Ratio (Deterministic)

A tree diagram (Figure 26) gives similar results to the previous regression model on loss ratio. It tells us that in the experiment space, the expected loss ratio is 0.44 overall. If the number of ASMs for a single ship per salvo (α) is less than 2.5, the expected loss ratio decreases to 0.24. This including 52% of the data. In addition, if α is less than 1.7 and z is greater than or equal to .97, then *side B*'s losses are just 6.9% of its force. This occurs in 17% of our data. In the other extreme, if $\alpha \geq 2.5$ and $B < 14$ and $A \geq 14$, then *side B* loses 93% of its force. This happens in 12% of our data.

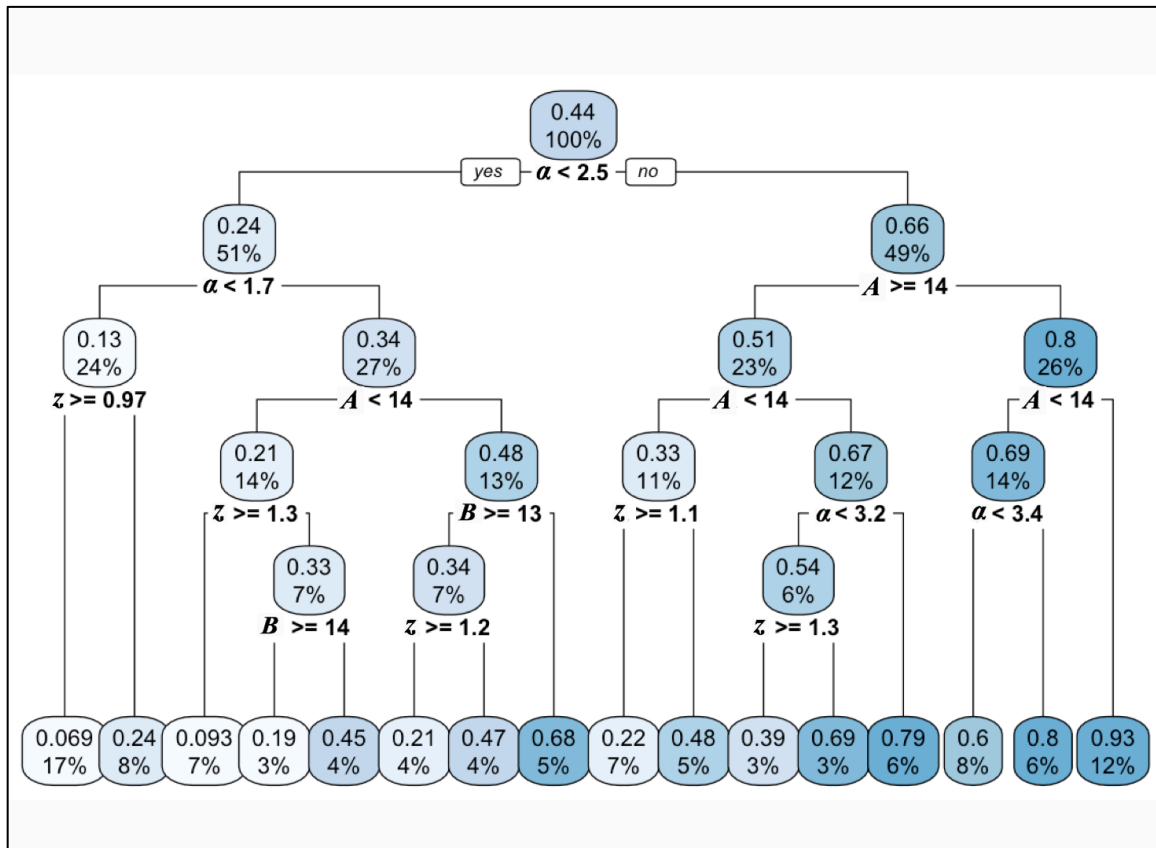


Figure 26. Tree Model of the Loss Ratio (Deterministic)

3. Fractional Exchange Ratio (FER)

Since *FER* is the loss ratio of *side B* divided by the loss ratio of *side A*, it needs special attention to be fitted by a regression model. As discussed in Figure 17, distribution

of FER is not normal. For these types of responses, an alternative solution may be a log transformation. Figure 27 shows the histogram of logarithmically transformed FER .

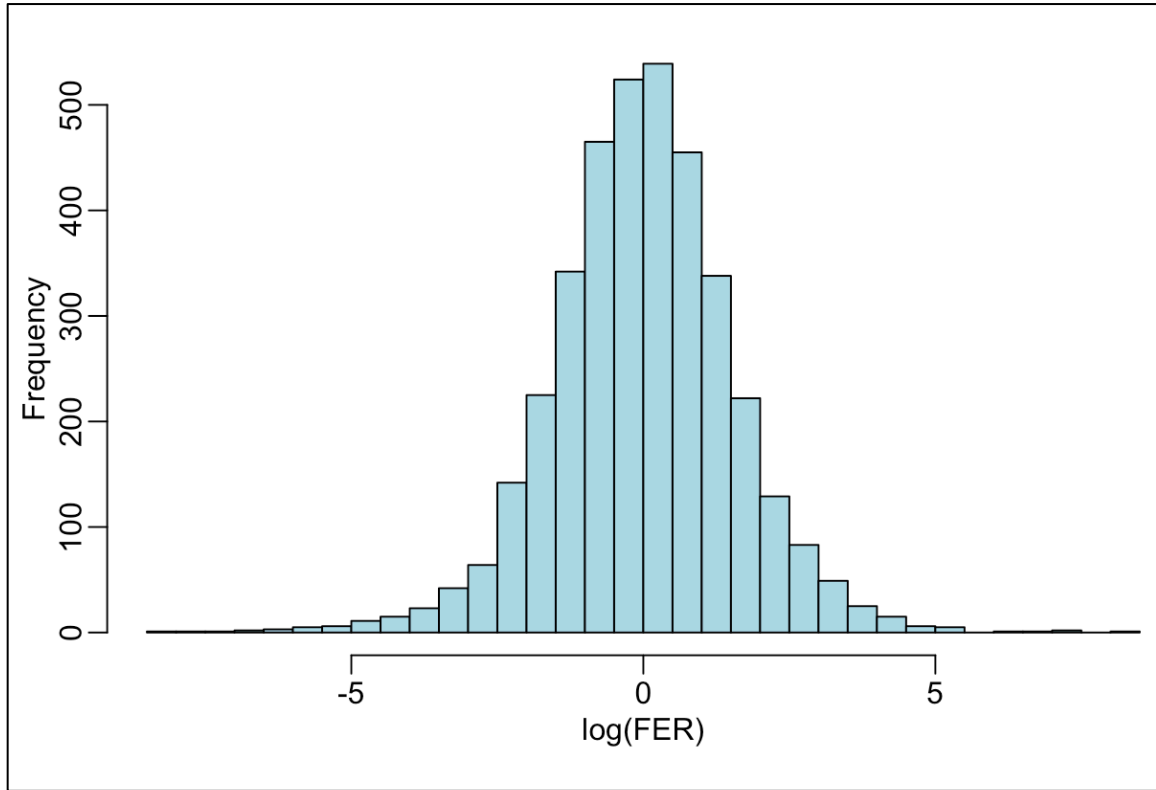


Figure 27. Histogram of the Logarithmically Transformed FER (Deterministic Model)

The logarithmically transformed FER appears to be a symmetric distribution. However, the distribution is still not normal, as verified with a *Shapiro-Wilk normality test*. For that reason, “Generalized Linear Model” (Faraway, J.J., 2015b) is an alternative approach since FER is a non-negative response. R programming has the `glm()` function for modeling these types of responses. `glm()` enables analyzers to fit regression models with distribution families other than the normal distribution. `glm()` has the distribution families of *Binomial*, *Gaussian*, *Gamma*, *Poisson*, and their inverses. In this study, the distribution family is determined as *Gaussian* linking with a logarithmic transformation.

However, before fitting the regression model, we need to determine the range of FER that will be used in the analysis. As FER is transformed logarithmically, values close to zero (near *over-defense* for *side B*) and values close to positive infinity (near *over-defense* for *side A*) are not applicable to use. So, the best way is to truncate the data symmetrically. To illustrate the idea, consider that the upper bound is truncated at ten while the lower bound is truncated at 0.1. Then, $\log(FER)$ will be bounded between $\log(0.1) = -2.3$ and $\log(10) = 2.3$. In order to determine the truncated area, a sensitivity analysis is done by deriving R^2 and sample size of the regression model as the truncation size increases from two to 50.

Figure 28 provides information on the sensitivity analysis made for deriving the truncation bounds. FER is varied between two and 50 in increments of one. For each value, say g , a regression model is fit using only the values of FER between $1/g$ and g , and R^2 and the number of observations is recorded. The thumbnail in the upper part of the figure shows how the regression model fit in terms of R^2 for the entire range. It is seen that as FER increases, the performance of model prediction increases between 2 and 15. After that point, the goodness of fit decreases as FER increases. Between two and 50, R^2 ranges from 0.92 and 0.96. The fitted model for the values of FER between four and 15 has the best performance. So, the red boxed area on the thumbnail is reanalyzed with a higher resolution by increasing FER steps by 0.5 in the reduced range.

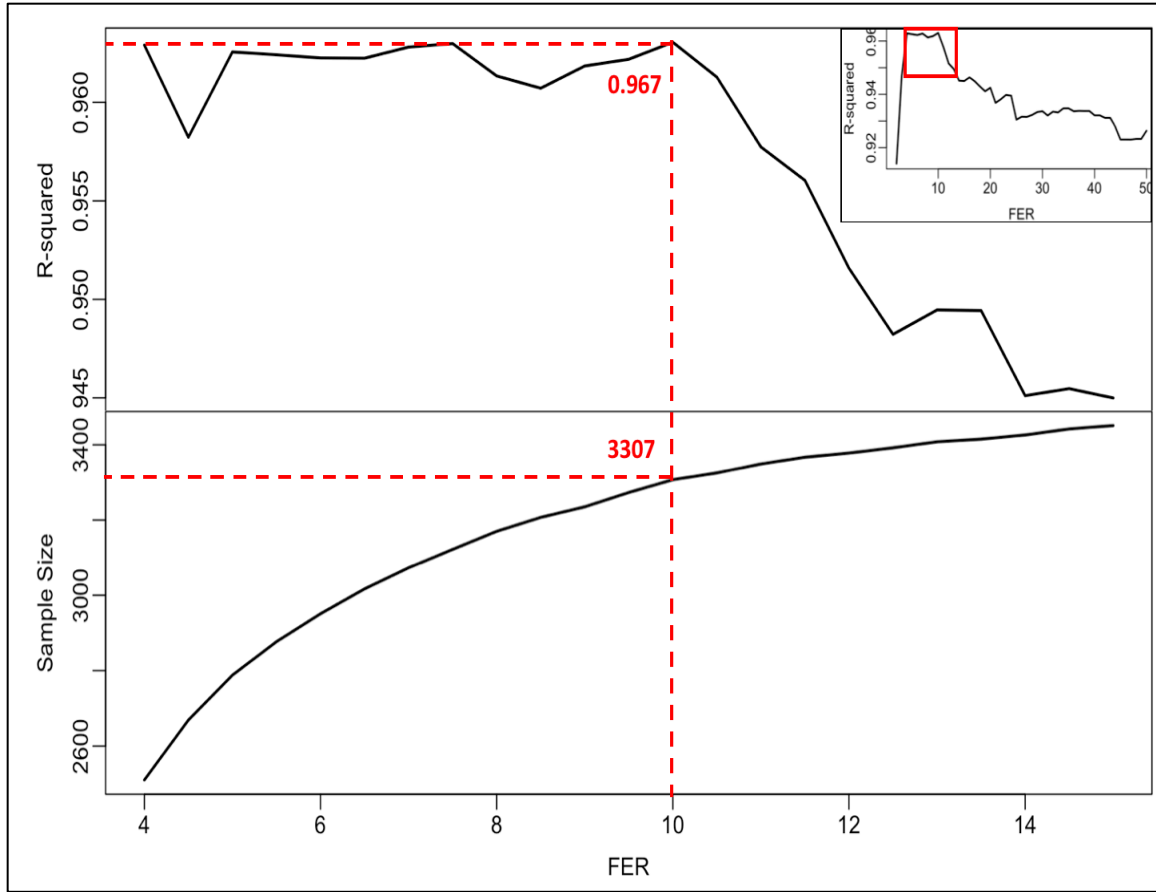


Figure 28. Sensitivity Analysis on R-squared for Given Range of *FER* (Deterministic)

The second sensitivity analysis draws the big picture in Figure 28. The upper part of the figure provides model fit performance using metric R^2 , while the lower part gives the number of available observations in this range. As these two parts are inspected, there is a trade-off between model performance and the available sample size to analyze. In this analysis, the best performance is provided when *FER* is limited to between 1/10 and 10, with a R^2 of .963 and 3307 observations. So, the next step is to refit a stepwise regression for *FER* values of between 0.1 and 10.

The new stepwise regression metamodel results in an R^2 of 0.963. The pairwise comparison of the actual values with the experimental results (for all 4420 DPs) is shown in Figure 29. R's summary output can be seen in Appendix C.

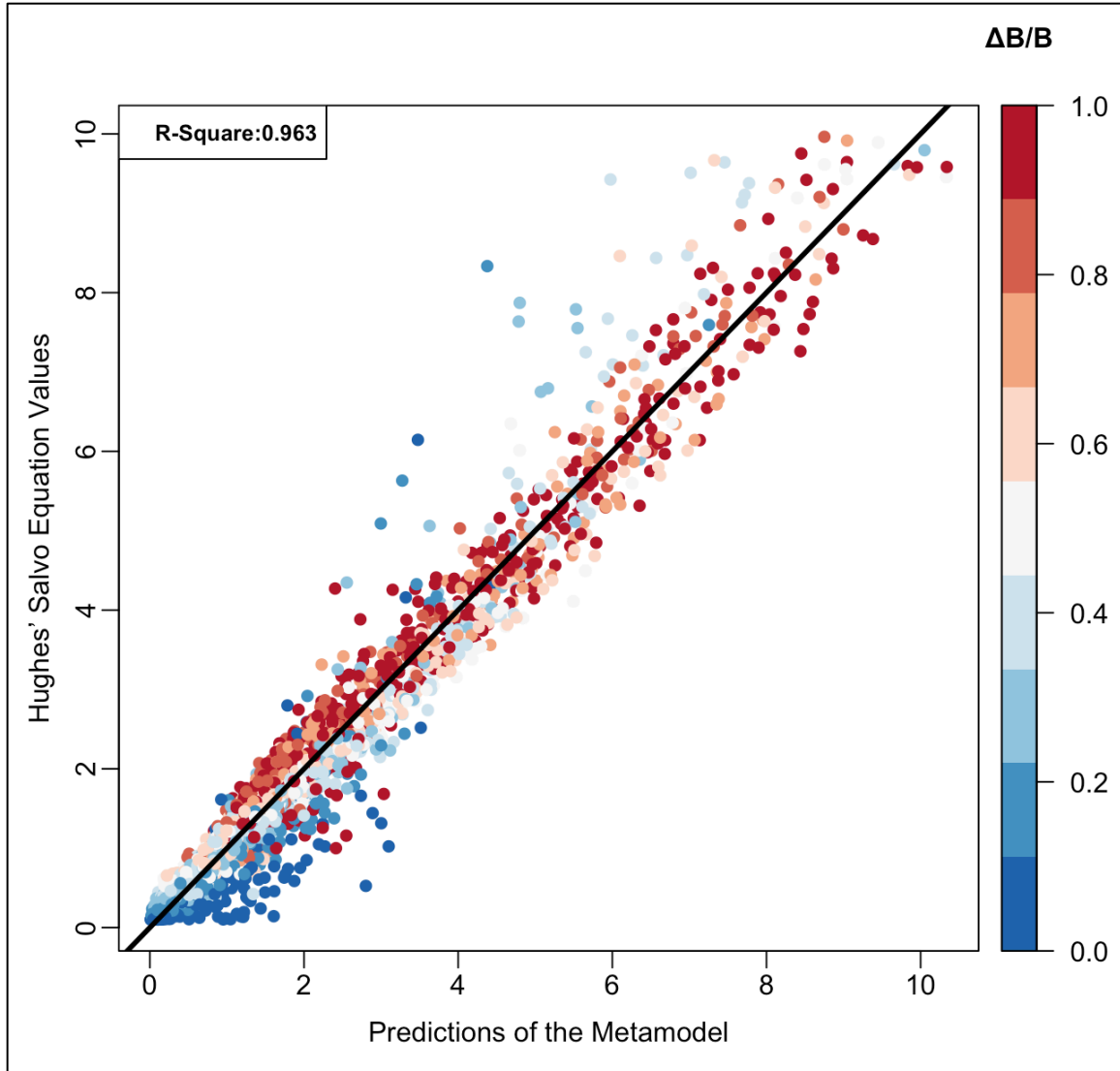


Figure 29. Comparison of Predicted *FER* Values with Actual Values (Deterministic Model)

In Figure 29, points are colored according to the loss ratio that was analyzed in the previous part. Dark red indicates that nearly all of the ships on *side B* are destroyed, while dark blue symbolizes the no loss cases. The figure reveals that the predictions of this regression metamodel are more scattered than the regression models belonging to the previous outputs. This is especially true for the DPs close to *over-defense*. In these cases, the metamodel tends to overestimates *FER*. Conversely, in high lethality cases, the

metamodel tends to underestimate FER . Overall, the metamodel performs reasonably well, capturing nearly 97% of the variance, considering the extreme variability of FER .

The prediction profiler, shown in Figure 30, tells us that as the number of ships in *side A* and its attributes (offensive, defensive and staying power) increase, FER increases. On the other hand, as the number of ships that *side B* has and its attributes increase, FER decreases towards zero. A notable feature is the nonlinear relationships produced by *glm*.

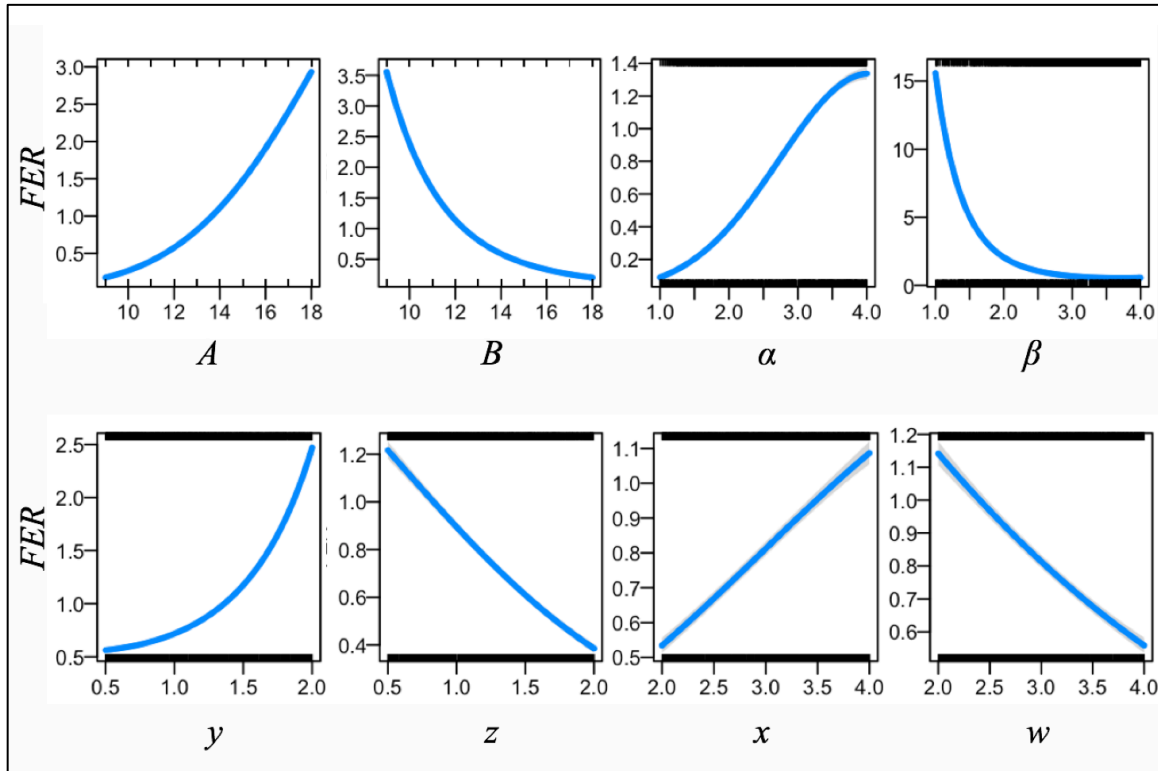


Figure 30. Prediction Profiler for FER (Deterministic Model)

In Figure 31, the tree diagram shows that B is at the top of a hierarchy. That is, B is the factor that most impacts FER . One might suspect that the overall expected value FER should be equal to one, since the experiment is symmetrically designed for each side. However, due to infinities, truncation, and a skewed distribution, the tree diagram reveals this is not so, with an overall expected FER value of 1.5. We also see that the number of ships on *side A* appears multiple times in the tree. This reinforces Hughes’s (1995) finding that “numerical superiority is the force attribute that is consistently most advantageous.”

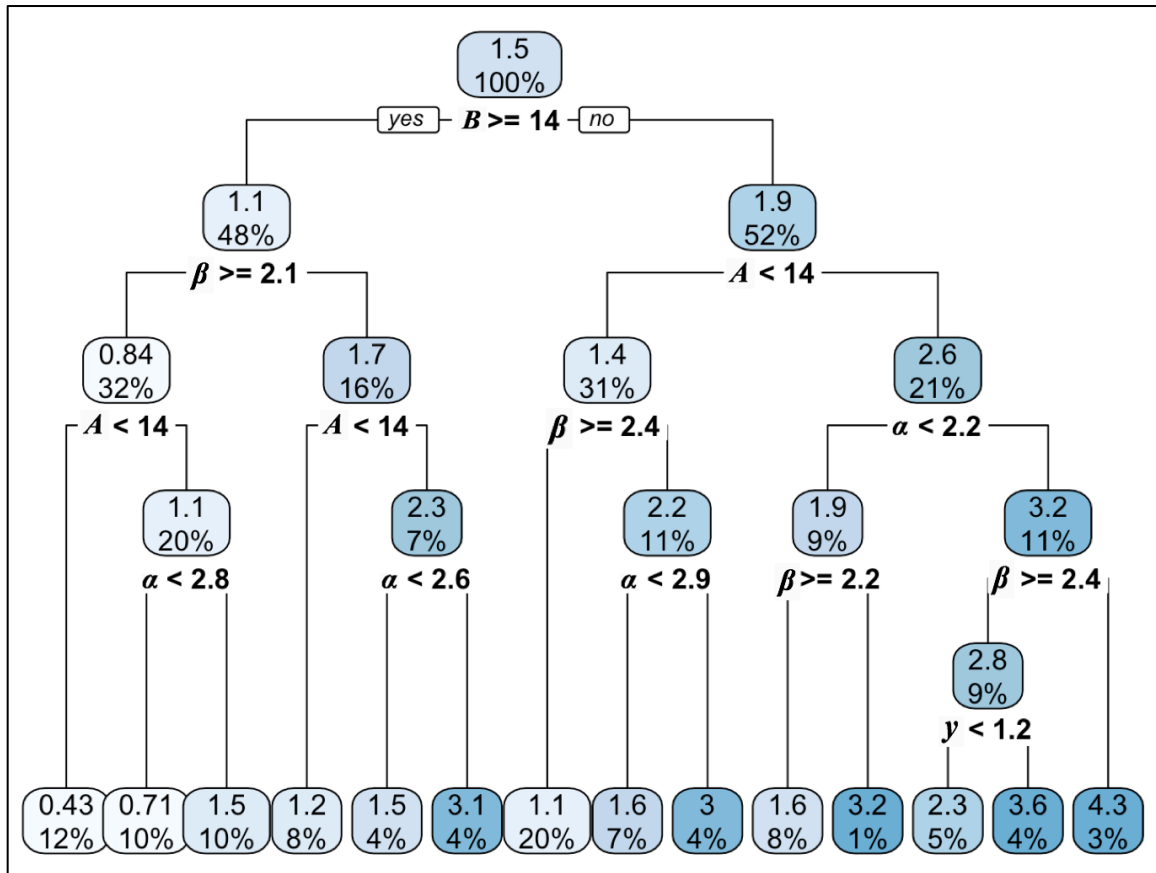


Figure 31. Tree Diagram of FER (Deterministic)

D. CHAPTER SUMMARY

In this chapter, three different outputs of Hughes’s deterministic salvo equations are reformulated as metamodels based on evaluations of 5,120 different input combinations. Each metamodel is analyzed using statistical tools to obtain insight. In this

part, many techniques were utilized, like stepwise regression and partition tree modeling. In addition, graphical methods like scatter plots, prediction profilers, and heat maps were used. Neither metamodeling nor these visual methods were applied in Hughes's original study. Our experiments reinforced the key findings from Hughes (1995) on the importance of numbers and regions of instability.

The experiments also confirmed what we knew to be true, that because of a lack of randomness in the model, the deterministic salvo equations always result in the same outputs for a fixed set of inputs. In addition to that, we observed that the response killing category (*intermediate*, *over-kill*, and *over-defense*), defined as a qualitative variable, is highly informative. Moreover, we found that over our experimental region, an easy-to-understand second order model explains 97% or more of the variability in the three outputs investigated.

In the stochastic salvo model analysis part (in the next chapter), the input variables will vary as random variables and generate a distribution of outcomes. Thus, there will be a chance to derive new findings, such as the probability of winning for an individual side. To see this, similar steps are taken to analyze Armstrong's (2005) closed-form stochastic salvo model. The number of ships lost and *FER* are common responses that are analyzed for both models. Furthermore, the probability of winning for *side B* is an additional response that the deterministic model does not provide (in any meaningful sense).

THIS PAGE INTENTIONALLY LEFT BLANK

V. ANALYSIS PART 2: STOCHASTIC SALVO MODEL (SOLVED BY SIMULATION)

Chapter IV analyzed Hughes’s deterministic salvo model using data farming. Experimental outputs were inspected statistically and reformulated as metamodels of the inputs. This chapter implements the same techniques on a simulation implementation of Armstrong’s (2005) stochastic salvo model. The simulation solves the conceptual stochastic salvo model exactly as specified in his paper, whereas his closed-form solutions presented in Chapter II are derived from approximations. Since one of the goals of this thesis is to bring out the possible differences in behavior of responses that are common for both models, the mean ship loss on *side B* and *FER* are also analyzed for the stochastic salvo simulation. In addition, we will use the variability feature of the stochastic model to derive additional responses, such as the mean probability of winning for a particular side—which has no practical meaning in the deterministic salvo model.

A. DESIGNING THE EXPERIMENT

The method for designing the experiment for the stochastic salvo model is similar to that of the deterministic salvo model. What is different is that random variables are added to induce randomness. The flow of this chapter is similar to that of the previous one.

1. Experimental Flow

A simulation of Armstrong’s stochastic salvo model was developed in the R programming environment. This model was implemented based on the conceptual model specified in Armstrong (2005) and implemented in Armstrong (2011). A flowchart of the simulation experiments is presented in Figure 32, and the code is available in Appendix B. Note the explicit use of random variables in the execution.

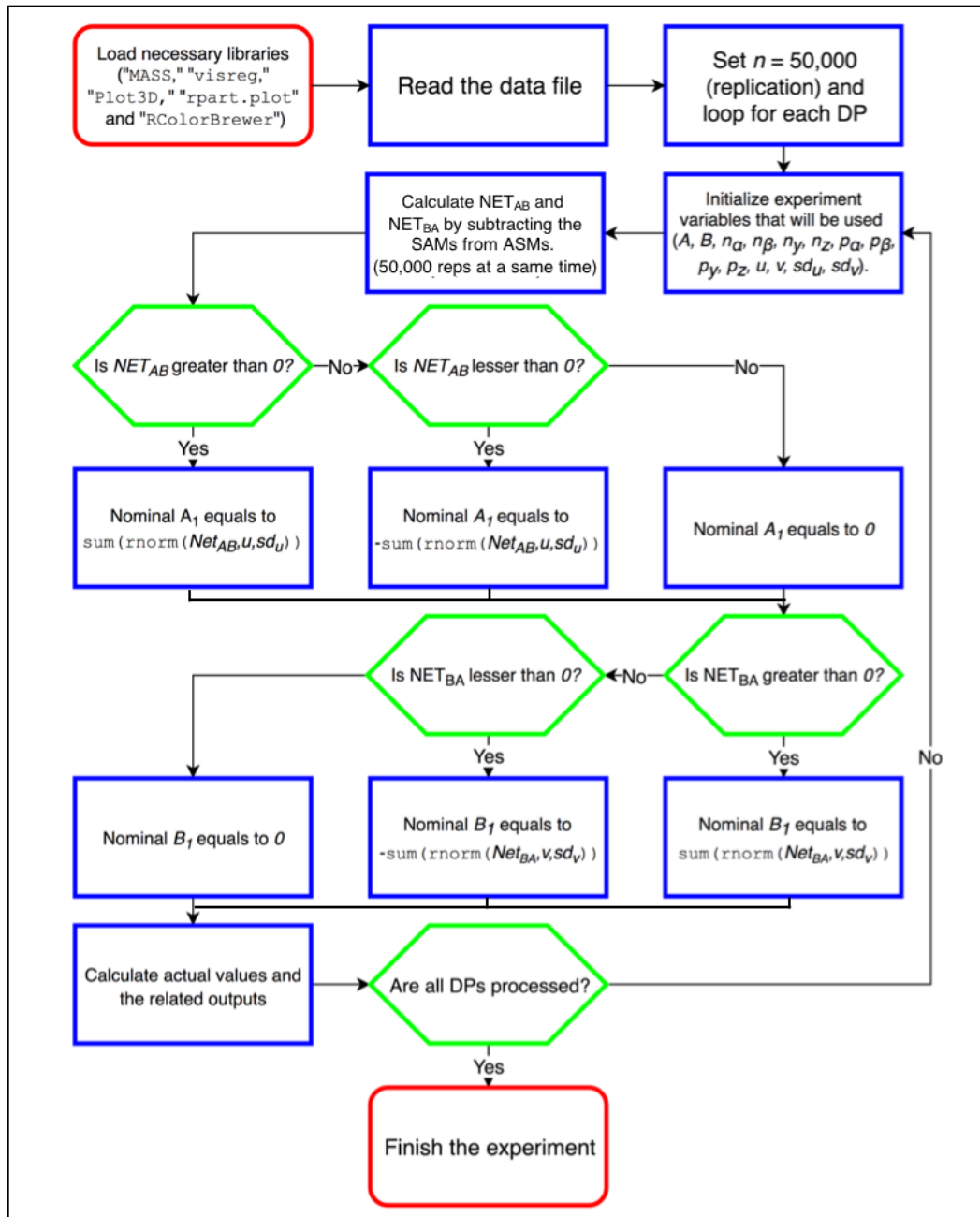


Figure 32. Flow Chart of the Stochastic Simulation Experiments

2. Data Farming

For Armstrong's stochastic salvo model, fourteen factors are explored over ranges similar to those above and in Armstrong (2011). Since we apply a binomial distribution for the number of offensive and defensive missiles, discrete values are assigned for ship size

and the maximum number of offensive and defensive missiles. With all of the levels we chose for the factors, a gridded design would have 1.45×10^{17} feasible DPs. Using a NOB, 512 DPs are selected and run. Since the simulation draws random numbers, 50,000 independent replications are made for each DP. This is the same number of replications Armstrong (2011) used. In total, 25,600,000 stochastic salvo battles are simulated. The factors, their ranges, and their number of levels are presented in Table 5.

Table 5. NOB Design for Stochastic Salvo Model Experiments

#	Factor	Explanation	Low	Inc.	High	# of Levels
1	A	Beginning force strength	9	1	18	10
2	B					
3	n_α	Maximum # of ASMs per ship per salvo.	2	1	4	3
4	n_β					
5	p_α	Prob. of successful launch of single ASM	0.500	0.001	1.000	501
6	p_β					
7	n_y	Maximum # of SAMs per ship per salvo	1	1	2	2
8	n_z					
9	p_y	Prob. of successful launch of single SAM	0.500	0.001	1.000	501
10	p_z					
11	u	Mean losses per hit	0.250	0.001	0.500	251
12	v					
13	sd_u	Sd of mean losses per hit	0.100	0.001	0.200	101
14	sd_v					
1.45×10 ¹⁷ DPs						

3. Analysis of Design Point Sets

Before starting the simulation, it is important to be sure that the selected DPs for the experiment are well chosen. This is why a scatter plot of the experiment's factors and a matrix of their pairwise correlations are generated. The DPs derived by the mixed NOB design are put into R and Figures 33 and 34 were generated to show the pairwise scatterplot and correlation matrix.

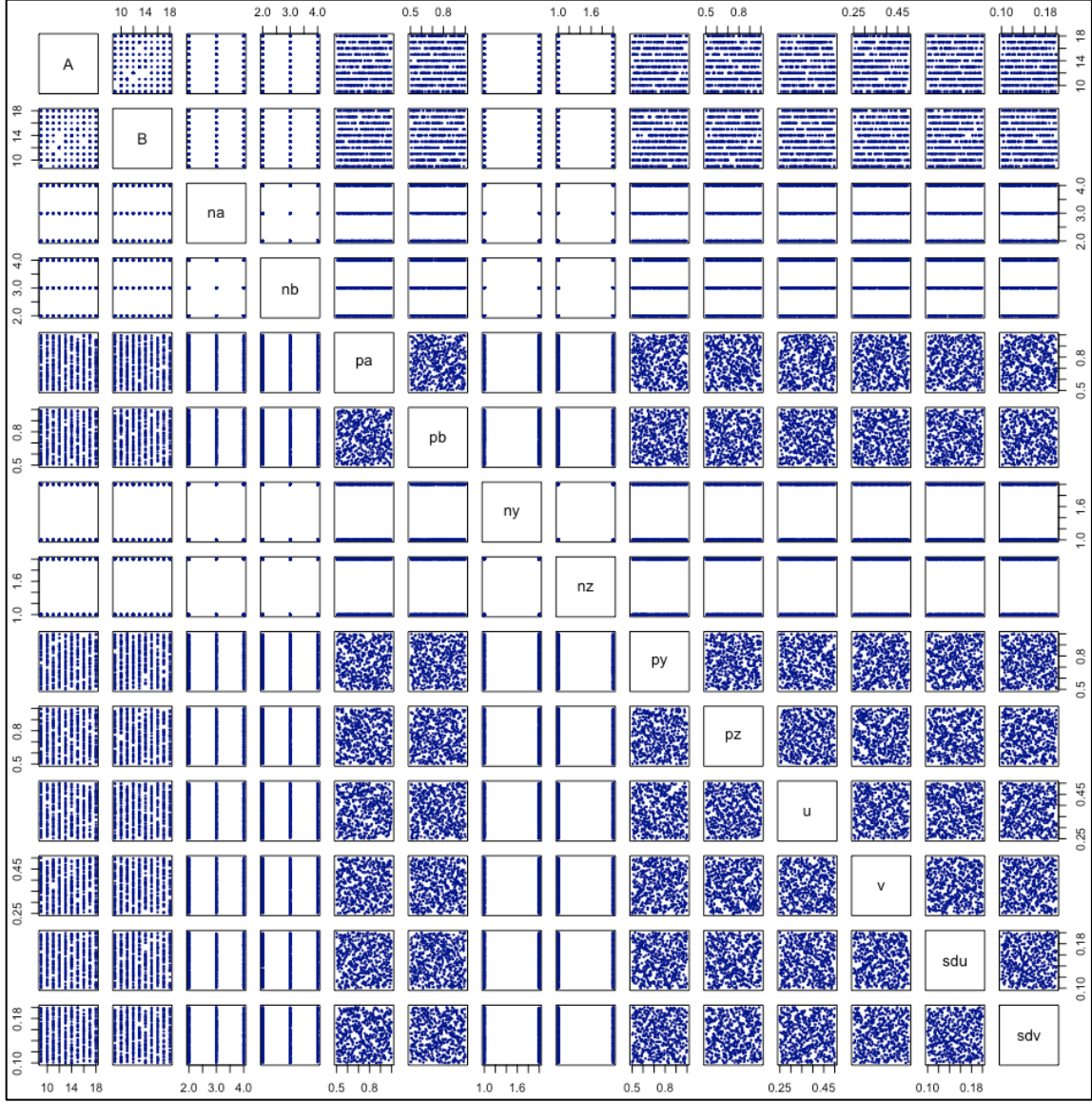


Figure 33. Pairwise Plots from the Design Matrix of the Stochastic Salvo Experiment

The design space is shown in Figure 33 as a pairwise scatterplot. We see that the mixed NOB design fills the interior for the continuous factors of the stochastic salvo simulation. For the discrete predictors having a low number of levels, the design fills much like a full factorial, at least in the pairwise projections. The scatter plots of the parameters n_a , n_b , n_y , and n_z may appear to have poor space-filling, but that is due to their discreteness. In addition, the levels for these predictors are kept low as their effectiveness is really high.

By all these considerations, this design satisfies the aim of this study and is enough to estimate high order terms in regression metamodels.

The correlation matrix shown in Figure 34 confirms that the pairwise correlations are within the limits stated by Vieira (2012), with all pairwise correlations between -0.02 and 0.02 . Thus, there is minimal multicollinearity among the factors.

	A	B	na	nb	pa	pb	ny	nz	py	pz	u	v	sdu	sdv
A	1.00	0.00	-0.02	0.00	0.02	0.01	-0.01	0.02	-0.01	-0.01	0	0.00	-0.02	-0.01
B	0.00	1.00	-0.01	-0.01	0.00	0.00	0.00	0.00	0.00	-0.01	0	0.00	-0.01	0.00
na	-0.02	-0.01	1.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0	-0.01	0.00	0.00
nb	0.00	-0.01	0.00	1.00	-0.01	-0.02	-0.02	-0.01	-0.01	0.01	0	-0.01	0.01	0.00
pa	0.02	0.00	0.01	-0.01	1.00	0.00	0.01	-0.01	0.00	0.00	0	-0.01	0.00	0.01
pb	0.01	0.00	0.00	-0.02	0.00	1.00	-0.01	0.00	0.00	0.00	0	0.01	0.00	0.00
ny	-0.01	0.00	0.00	-0.02	0.01	-0.01	1.00	-0.01	-0.01	-0.02	0	0.01	-0.01	0.01
nz	0.02	0.00	0.00	-0.01	-0.01	0.00	-0.01	1.00	0.02	-0.01	0	-0.01	0.00	0.01
py	-0.01	0.00	0.00	-0.01	0.00	0.00	-0.01	0.02	1.00	0.00	0	0.00	0.00	0.00
pz	-0.01	-0.01	0.00	0.01	0.00	0.00	-0.02	-0.01	0.00	1.00	0	-0.01	0.00	0.00
u	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1	0.00	0.00	0.00
v	0.00	0.00	-0.01	-0.01	-0.01	0.01	0.01	-0.01	0.00	-0.01	0	1.00	0.01	-0.01
sdu	-0.02	-0.01	0.00	0.01	0.00	0.00	-0.01	0.00	0.00	0.00	0	0.01	1.00	0.00
sdv	-0.01	0.00	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0	-0.01	0.00	1.00

Figure 34. Pairwise Correlation Between Factors (Stochastic)

B. OUTPUT ANALYSIS OF STOCHASTIC SALVO SIMULATION EXPERIMENTS

The first step in our analysis is to review the summarized information for the outputs of the simulation of the stochastic salvo model. There are three outputs that this thesis focuses on, they are: *number of ship loss (ΔB)*, *fractional exchange ratio ($[\Delta B/B]/[\Delta A/A]$)*, and *the probability of victory*. Figure 35 provides a basic statistical summary of the results. By saying summarized results, we mean that all statistical metrics are derived by averaging the outputs for each design point that was simulated 50,000 times. The summaries are over the sample means of the 512 design points. As was done previously, only outputs belonging to *side B* are investigated, since the sides are essentially symmetric in both the model and the design.

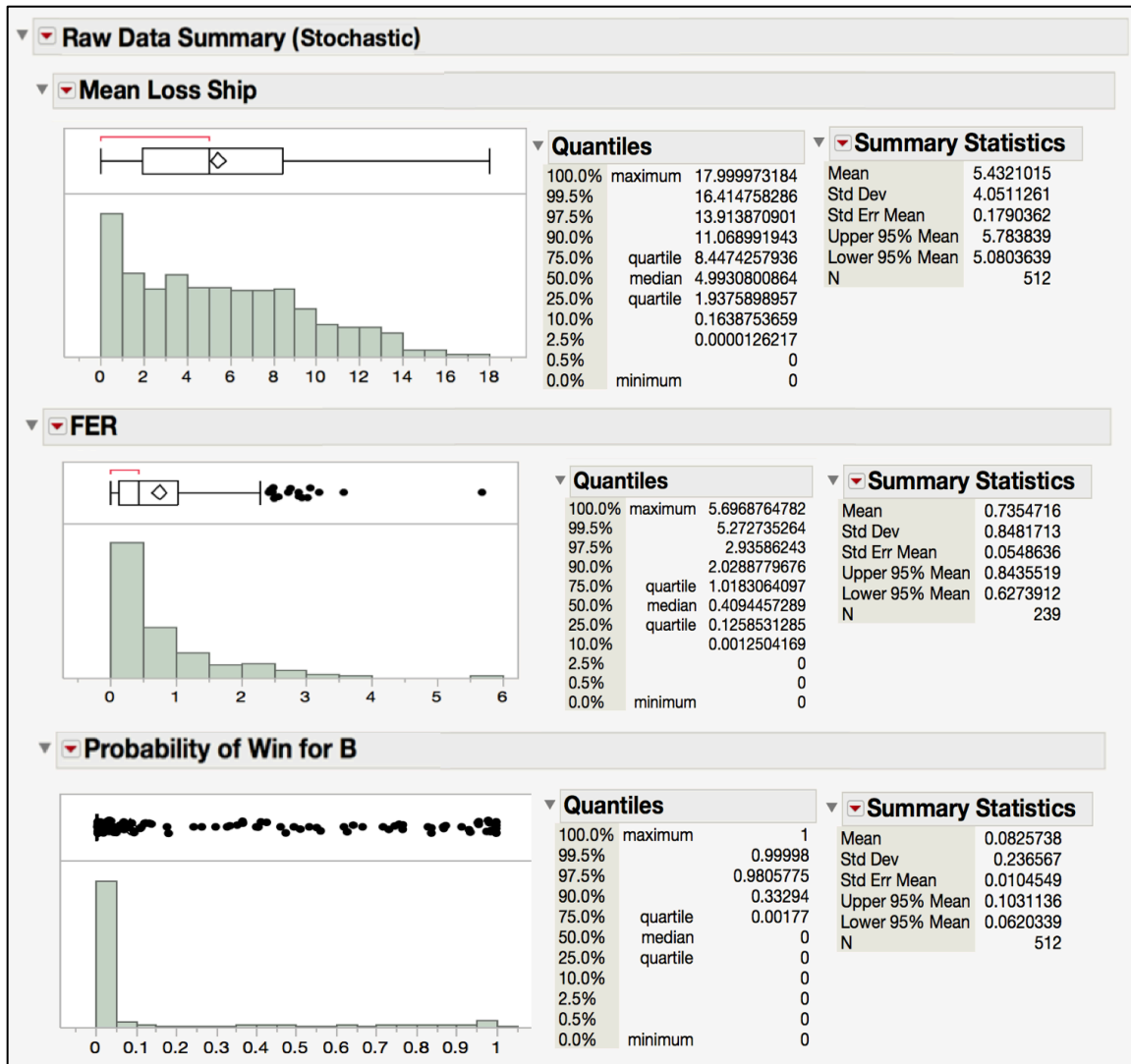


Figure 35. Summarized Output Data Distributions for the Stochastic Model Simulation

The first row in Figure 35 summarizes the mean ship loss, which ranges from zero to almost 18, with mean and standard deviation of 5.43 and 4.05, respectively. It is noteworthy that these values are similar to those obtained in our deterministic experiments. For the second row, *FER* has a wide range and is right-skewed between zero and positive infinity. The analysis program, JMP, disregards the infinity values belonging to *FER* and scales to use a range from 0 to 6. Over these 239 DPs, *FER* has a mean and standard deviation of 0.73 and 0.84, respectively. It is interesting that the available range is narrow when compared to the deterministic salvo model's results. In addition to that, there are only

239 valid (finite) observations, which means that over half of the experimental region does not meet the requirements for our analysis of *FER*. For the deterministic simulation results, this proportion was about 14% (700 out of 5120). An explanation of the situation lies within the simulation algorithm. For each DP, if even a single one of the 50,000 calculations of *FER* results in infinity (i.e., $\Delta A = 0$), the average *FER* for that DP is infinity. This is a byproduct of the variability inherent in the stochastic salvo model. It also suggests that perhaps looking at medians is more appropriate. In the regression modeling phase, additional analysis is done to derive the best model to predict *FER*.

Analysis of the result category in the deterministic salvo model is rigid and always concludes with the same outcome (*intermediate*, *over-defense*, or *over-kill*) for a given DP. However, for the stochastic salvo model, the result can vary due to its probabilistic nature. This gives the motivation for formulating the probability of winning for *side B* by using data farming, design of experiments, and regression modeling techniques. Simulation results for the winning probability are distributed between zero and one, which is the maximum feasible range for any probabilistic event. Over the 512 DPs, the probability of *side B* winning ranges from zero to one, with a mean of 0.08 and a standard deviation of 0.24. Winning is defined as *side A* being totally wiped out while *side B* has some forces remaining.

C. REGRESSION MODEL ANALYSIS

This section builds metamodels of the simulation outputs using stepwise regression. A difference from the regression metamodels of the deterministic salvo equations is that the responses of the stochastic simulation are averages of 50,000 replications.

1. Actual Number of Ship Loss

The next step is to fit a stepwise regression model for up to a full second order model with the input factors as candidates. The regression model that results has 57 terms and an R^2 of 0.978. See Appendix C for R's summary of the regression. While the fit is quite good, the residuals are not normally distributed.

A comparison of the predicted values and actual values of ΔB is presented in Figure 36. Coloration in the figure shows the mean kill ratio for *side B*, which was also looked at in our deterministic model analysis. Color is distributed between dark blue and dark red, and show *over-defense* and *over-kill*, respectively. As seen in the figure, the metamodel has poorer predictions in *over-defense* and *over-kill* conditions.

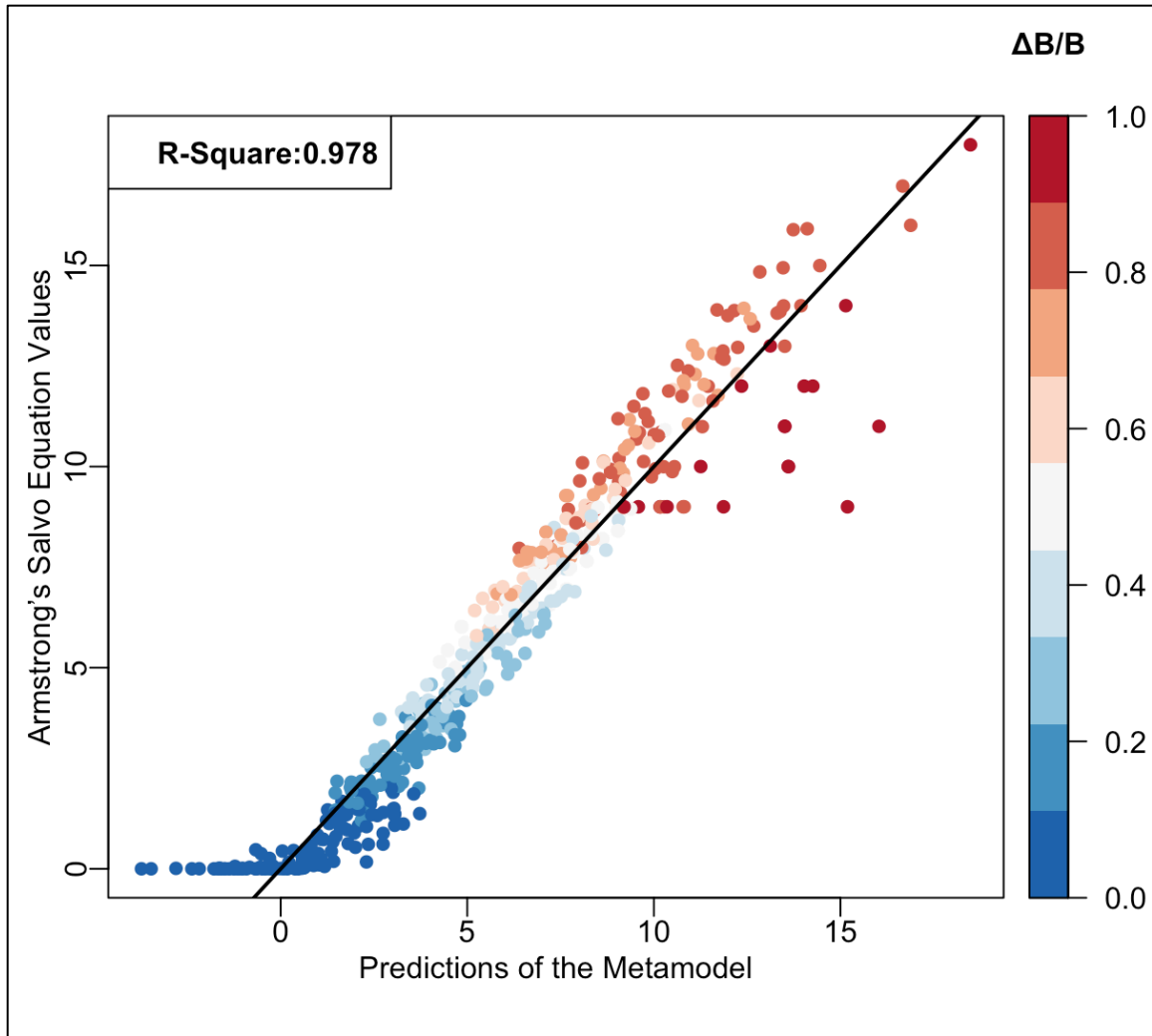


Figure 36. Comparison of Predicted ΔB Values with the Actual Values (Stochastic)

Figure 37 provides the fit results of mean loss for the three result categories. A single DP can result in multiple categories due to the stochastic nature. For that reason,

categorization for each DP is made by applying the dominant result over the 50,000 replications. To give an example, if 50,000 replications of the simulation for a single DP results in 15,000 *intermediate*, 15,000 *over-defense*, and 20,000 *over-kill* outcomes, that DP is categorized as an *over-kill* result.

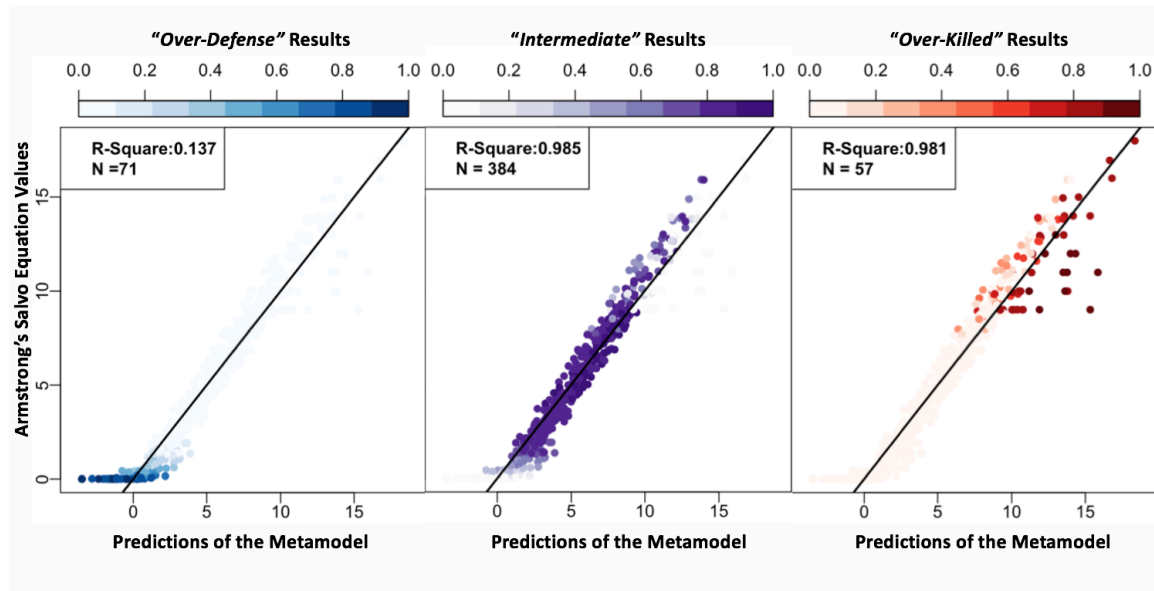


Figure 37. Comparison of Predicted ΔB Values with the Actual Values by Category (Stochastic)

With this initial knowledge, according to Figure 37, 71 DPs are dominated with *over-defense*, while 384, and 58 DPs are categorized as predominantly *intermediate* and *over-kill*, respectively. R^2 metrics for these three categories show how well they fit with the overall simulation results. For example, salvo fires that resulted in *over-defense* have an R^2 of 0.137, implying that these results limit the performance of the overall regression model. On the other hand, for the *intermediate* and *over-kill* results, the regression model shows much better performance, with R^2 values of 0.985 and 0.981, respectively. So, the next step is to refit a regression model on only the *intermediate* dominated results.

The linear model on the *intermediate* outcomes has a much better fit, with an R^2 of 0.999. The full R regression summary is in Appendix C. As shown in Figure 38 the simulation results and model predictions are gathered around the line of equality. In addition, although the R^2 of fitted *over-defense* results is still only 0.627, a scatter plot of a comparison of categorized result of the simulation with the model's prediction (Figure 39) improved in all categories. Specifically, the *intermediate* and *over-kill* categories have R^2 values of 0.999 and 1, respectively.

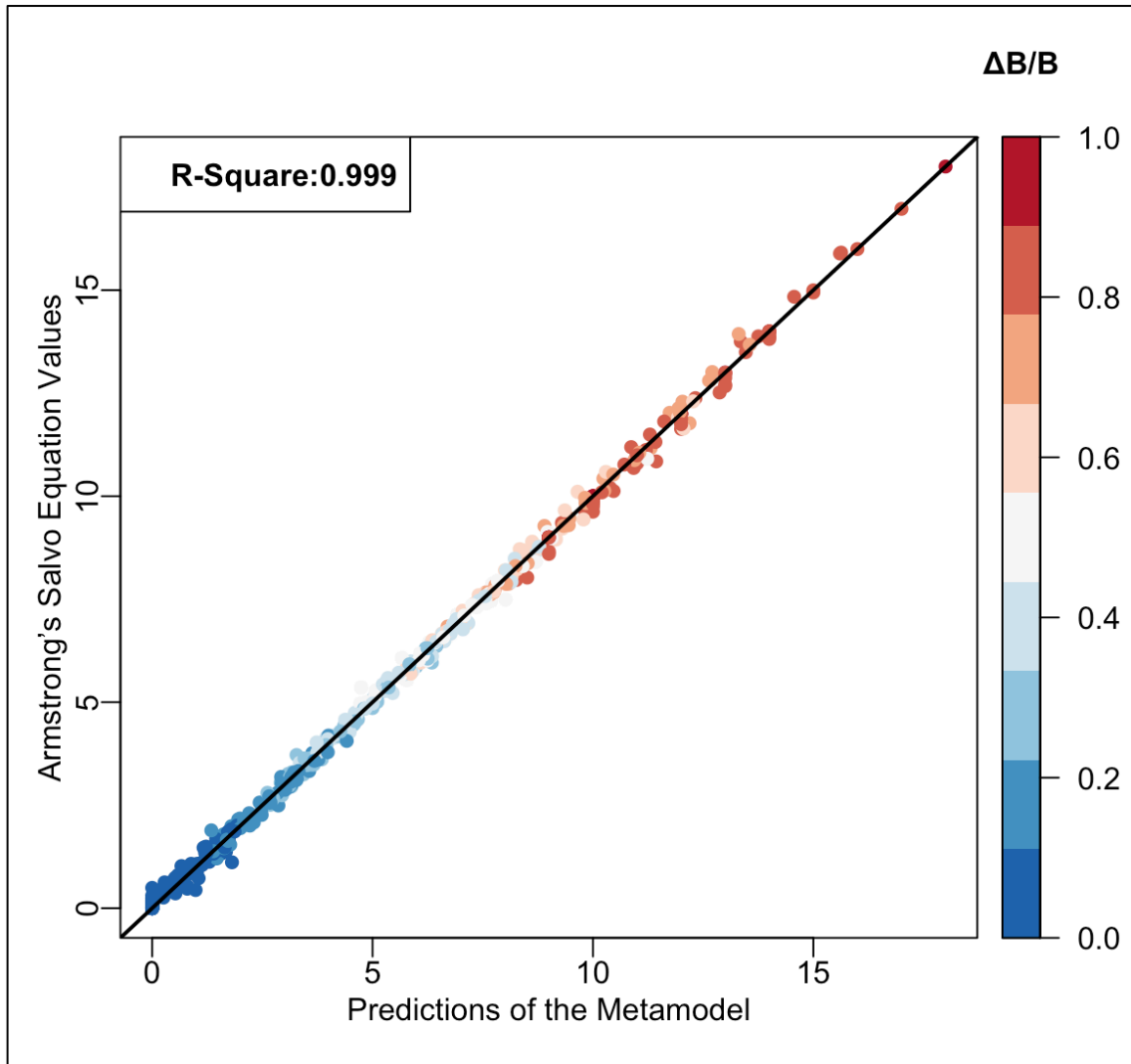


Figure 38. Predicted ΔB Values versus Simulation Values for the Metamodel Based on Intermediate Outcomes

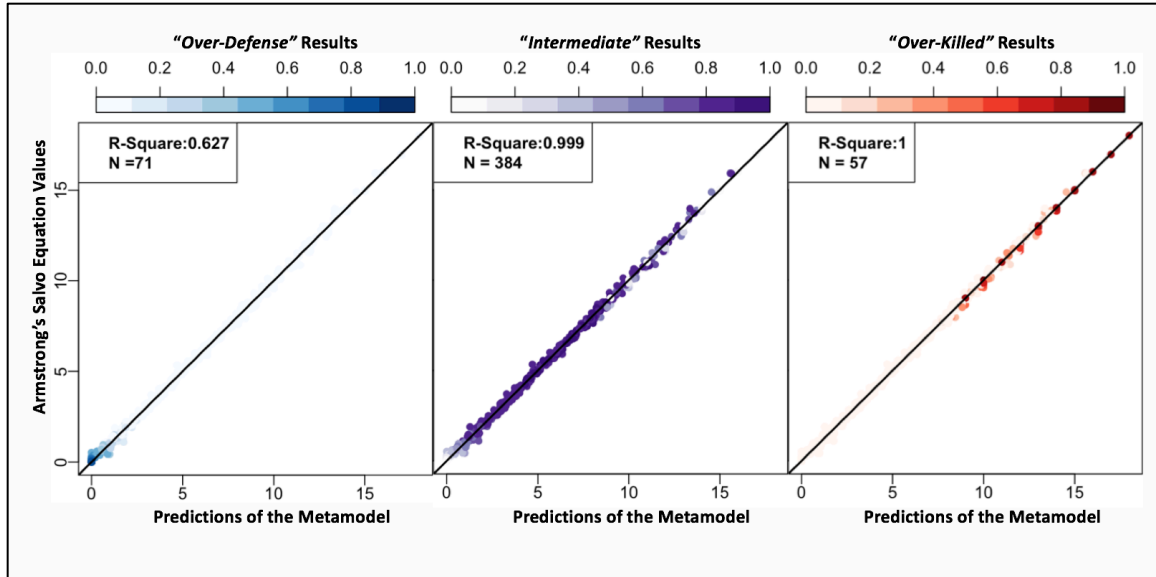


Figure 39. Comparison of Predicted ΔB Values with Actual Values by Category

The prediction profiler is provided in Figure 40. A , n_α , p_α , and v have an increasing effect on ship loss for *side B* (i.e., ΔB), while B , n_z , and p_z have a decreasing effect. This is similar to what we saw with the deterministic salvo model. Moreover, all of these are intuitive. The rest of the factors are not as influential. Hence, our heat map, which reveals two-way interactions, focuses on the factors just listed.

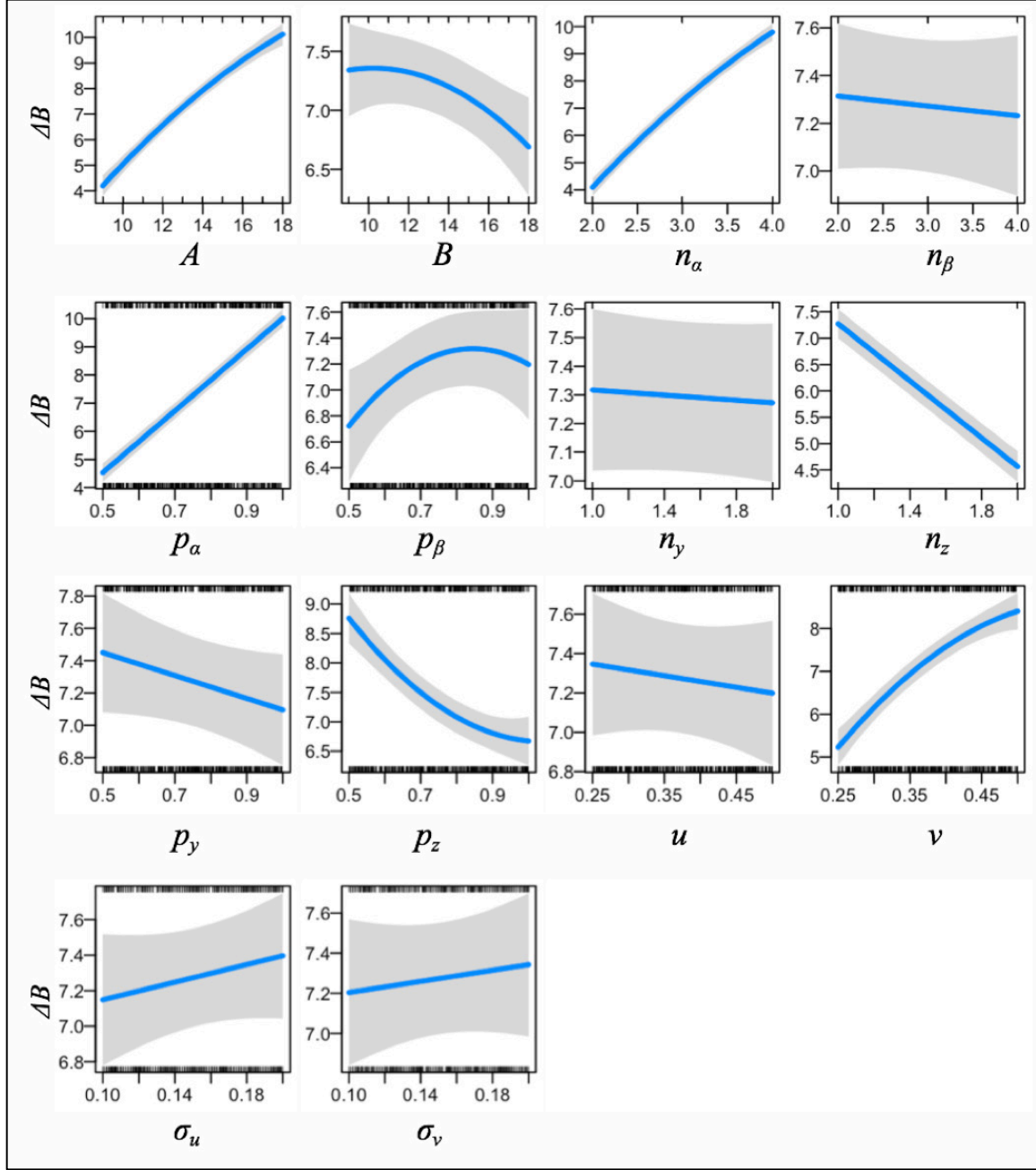


Figure 40. Prediction Profiler for ΔB (Stochastic)

A heat map is shown of ship loss for *side B* in Figure 41. It has similarities with the deterministic model. As before, interactions between factors related to the offensive power of *side A* can dramatically increase losses for *side B* (i.e., ΔB), while interactions between factors related to the defensive power of *side B* have the opposite effect. Interactions between a mixture of offensive and defensive power has balancing effects. One can conclude that the number of ships, available maximum number of both ASMs and SAMs, and their success probabilities have a substantial impact on losses.

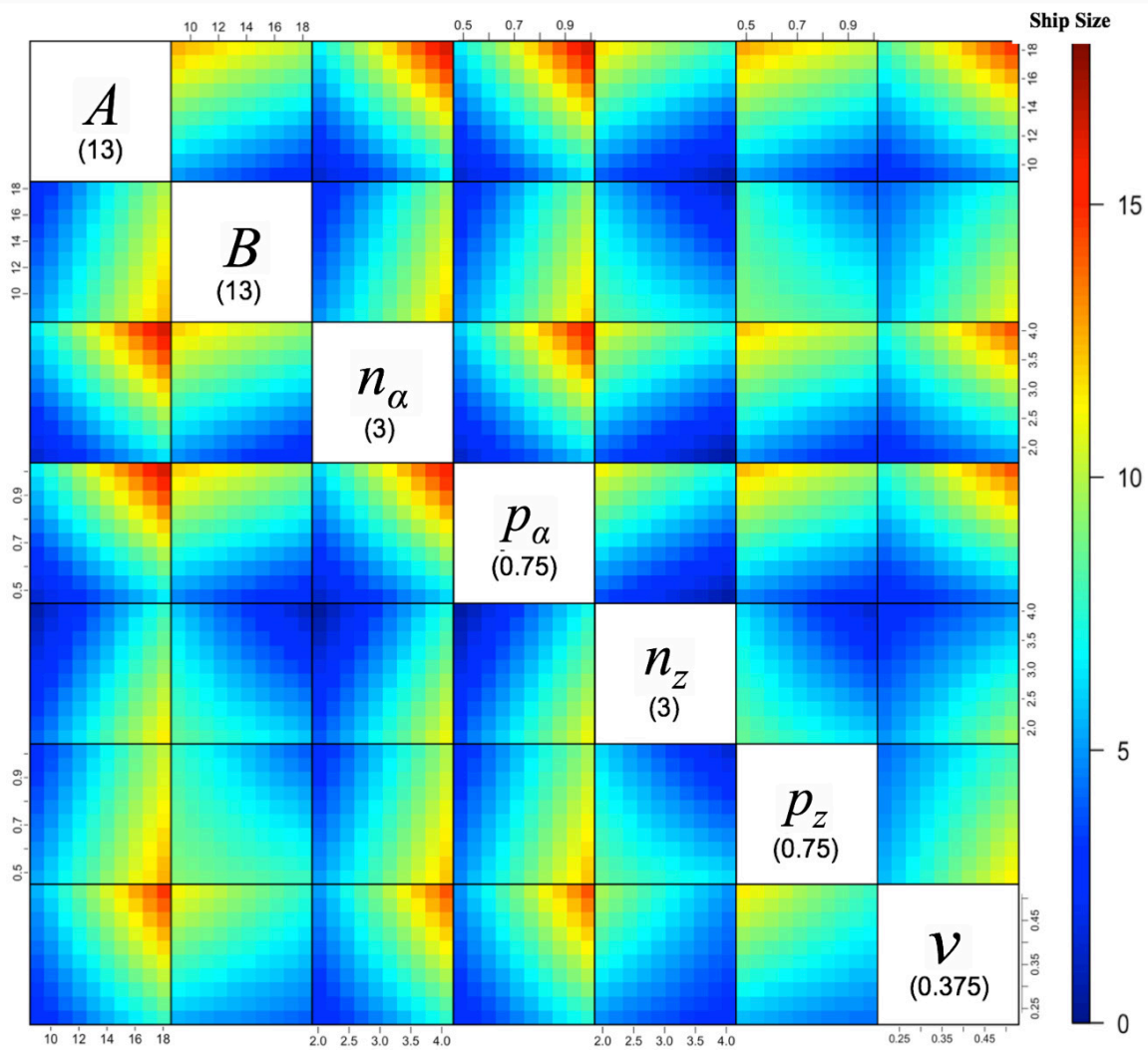


Figure 41. Heat Map of ΔB (Stochastic)

The tree model diagram in Figure 42 shows that the factors A , n_α , and p_α are important in determining ΔB —as they are the top branches in the tree. All of these are predominantly under the control of *side A*. With this provided information, *side B* minimizes its losses by increasing its defensive and staying powers (i.e., n_z , p_z , and v).

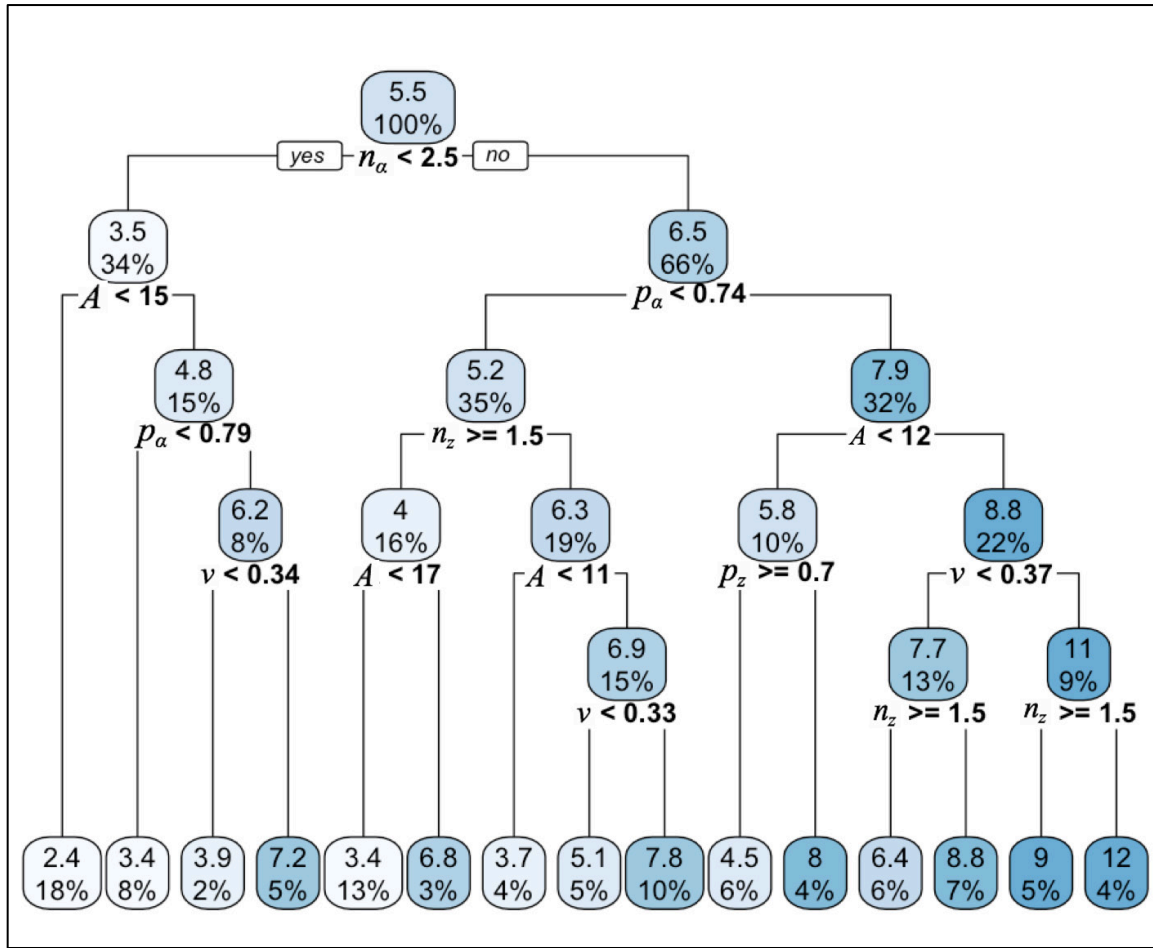


Figure 42. Tree Diagram of ΔB (Stochastic)

2. Fractional Exchange Ratio (*FER*)

In his article, Armstrong (2005) did not analyze *FER*. Since *FER* is central to Hughes's (1995) study, this thesis investigates whether there are important differences in *FER* between the deterministic and stochastic salvo models.

With this motivation, in each replication of each design point, FER is calculated according to Equation 2. After replicating 50,000 times, the mean is calculated to estimate FER for each DP. As mentioned in the summary statistics analysis, there are 239 DPs that did not end up as infinity. For the following steps, the process flows similarly to what was done in the deterministic salvo model analysis using generalized linear models. First, let us draw the histogram of FER after a logarithmic transform.

The histogram of the logarithmic transformed FER presented in Figure 43 is non-symmetric, with a longer left tail. Many of the DPs favoring *side A* have been trimmed. This is due to averaging 50,000 FER values, some of which include infinity (when ΔA is zero). R disregards these observations.

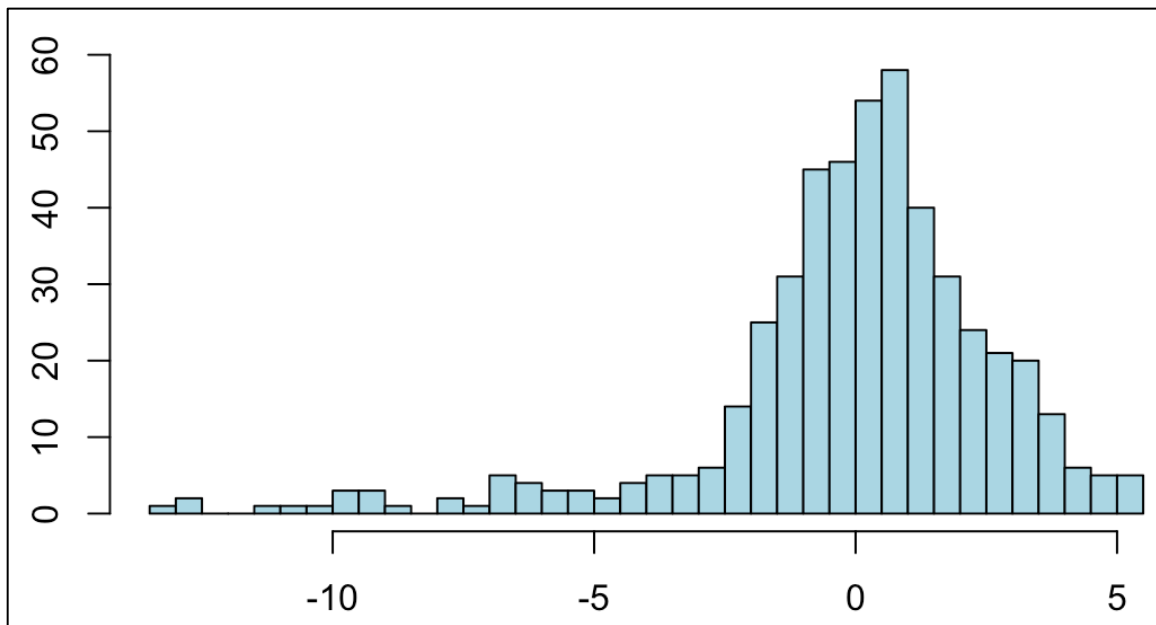


Figure 43. Histogram of the Logarithmic Transformed FER (Stochastic)

By trimming each side of these observations, as was done in the deterministic salvo model analysis, this asymmetric distribution will be mitigated. So, the first part is a sensitivity analysis with a trim value varied from two to 10 (and the inverse). After that, the data for FER values in the range that gives higher R^2 values will be reanalyzed. In the

end, the trimmed area is determined with a consideration of the sample size. The result is visualized in Figure 44.

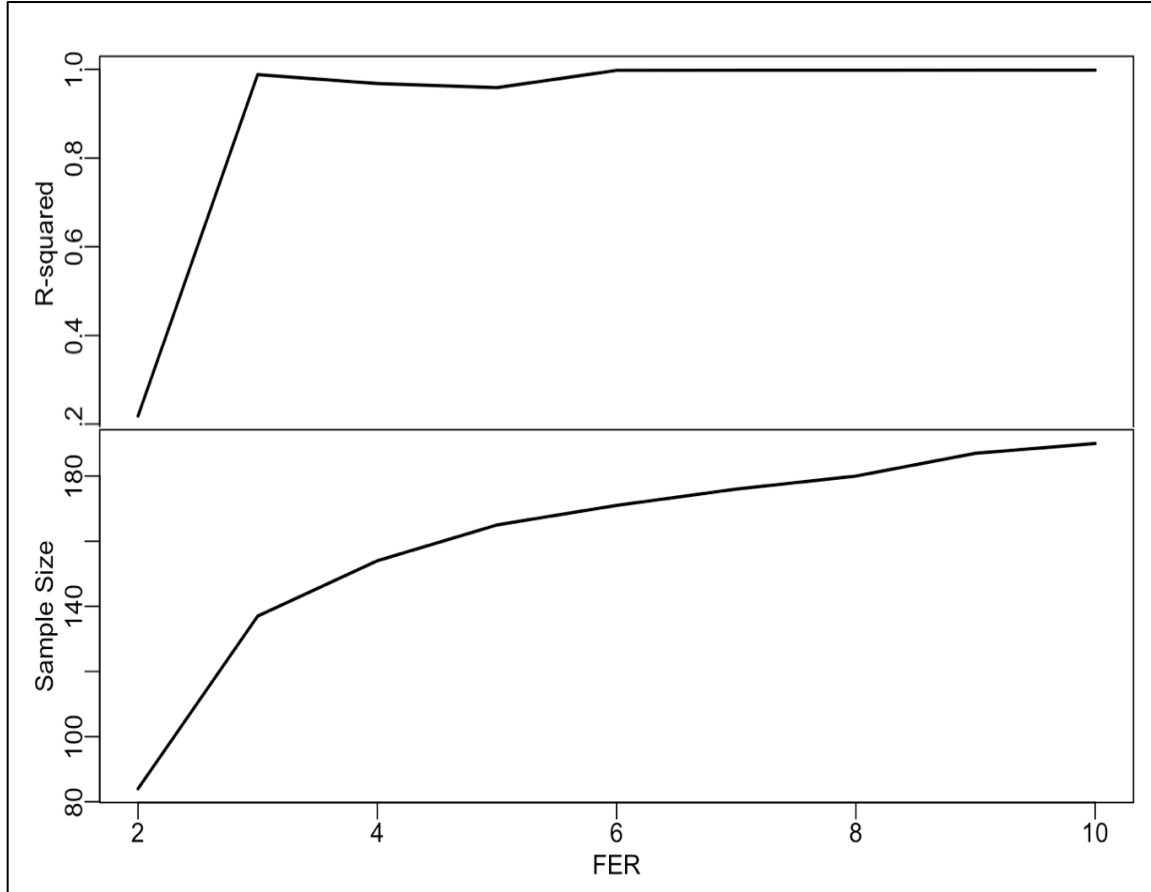


Figure 44. Sensitivity Analysis on R-squared for Given Range of *FER* (Stochastic)

Figure 44 shows that logarithmic transformed *FER* does not need to be trimmed. According to the figure, as the range of *FER* increases, both the number of eligible observations and R^2 are increasing. So, we fit a stepwise regression model by including all remaining observations except those with $FER = 0$, since its logarithmic transformation is negative infinity (i.e., the opposite of the infinities that were already excluded).

After fitting the generalized lineal model using stepwise AIC, it is observed that the result is quite satisfying, with an R^2 value of 0.998. R's summary glm output is in Appendix

C. Even the single discordant point is successfully fit. Figure 45 provides a scatter plot of the actual values versus the metamodel's predictions.

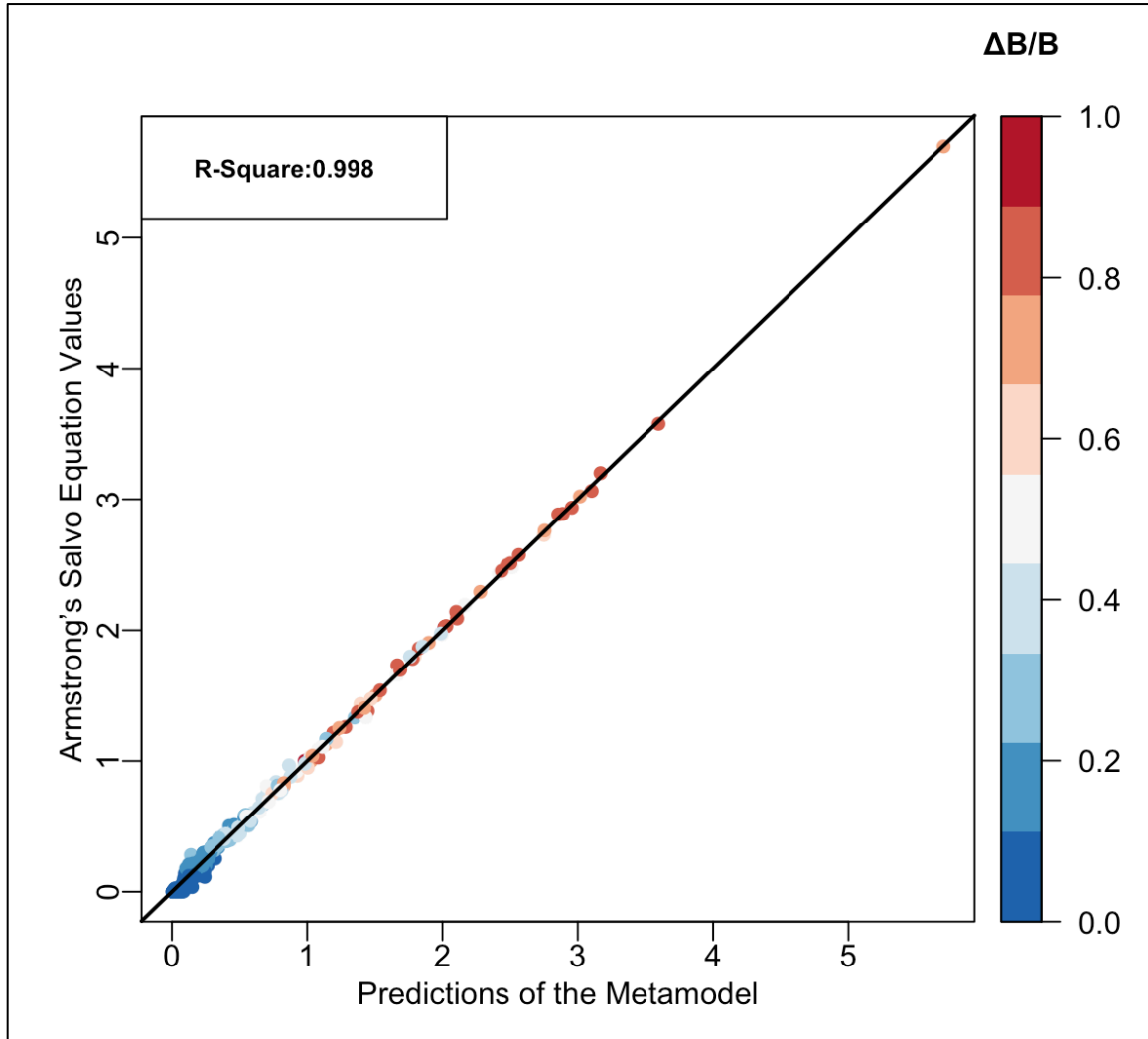


Figure 45. Comparison of Predicted *FER* Values with Actual Values (Stochastic)

The prediction profiler, shown in Figure 46, shows that as the number of ships on *side A* and its offensive, defensive and staying powers increase, *FER* increases. On the other hand, as the number of ships that *side B* has, as well as its offensive, defensive and staying powers increase, *FER* decreases towards zero. The profiles are decidedly nonlinear.

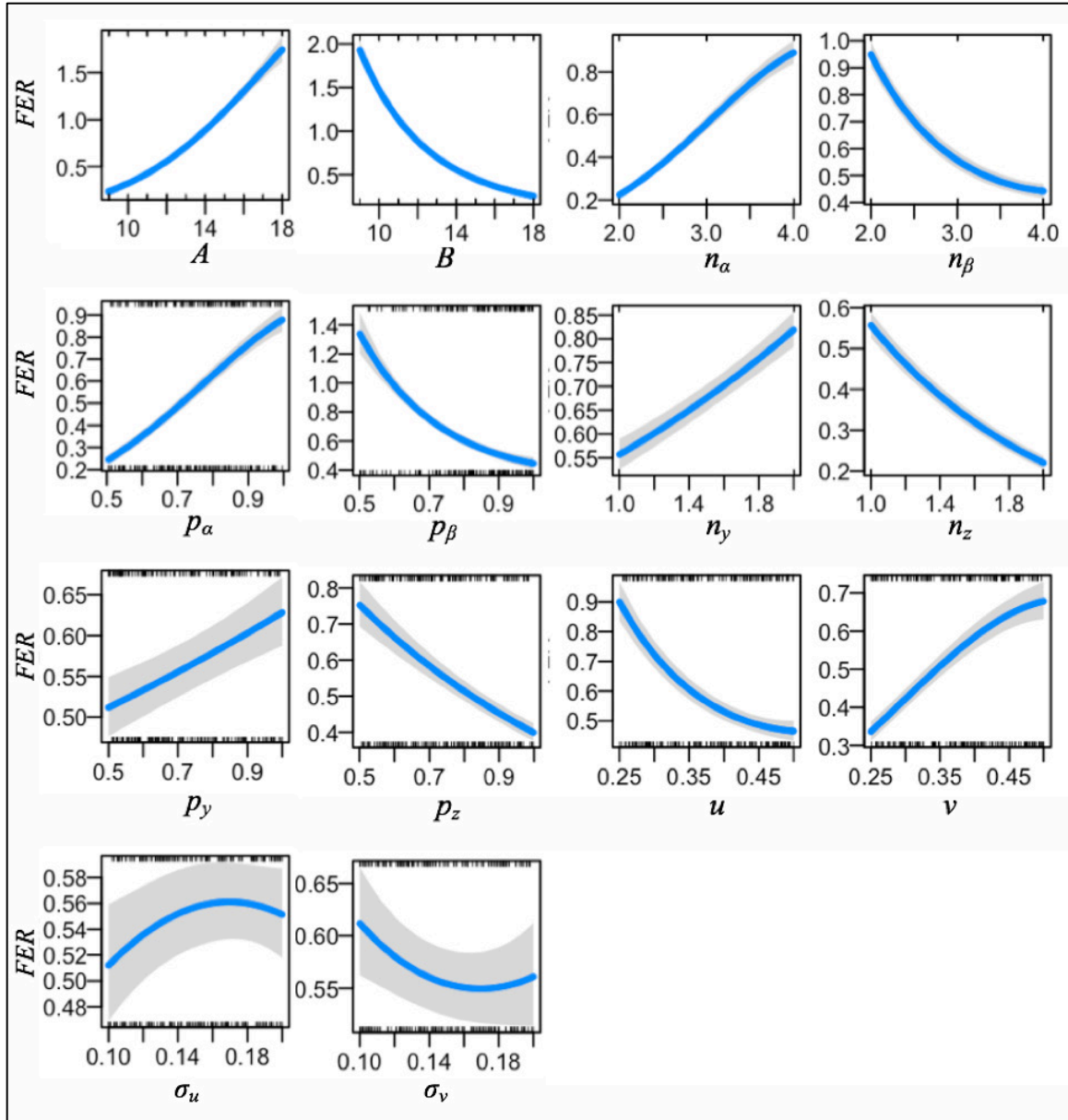


Figure 46. Prediction Profiler for FER (Stochastic)

The tree diagram in Figure 47 tells that the force size factors B and A have the most effect on FER . This also matches what Hughes's (1995) concluded. It follows that *side B* should strive for numerical superiority above all else, if possible. In addition, unlike with the deterministic salvo model, the expected value of FER overall is one. This is because the sensitivity analysis resulted in a truncation so that the data analyzed was nearly symmetrically for both sides.

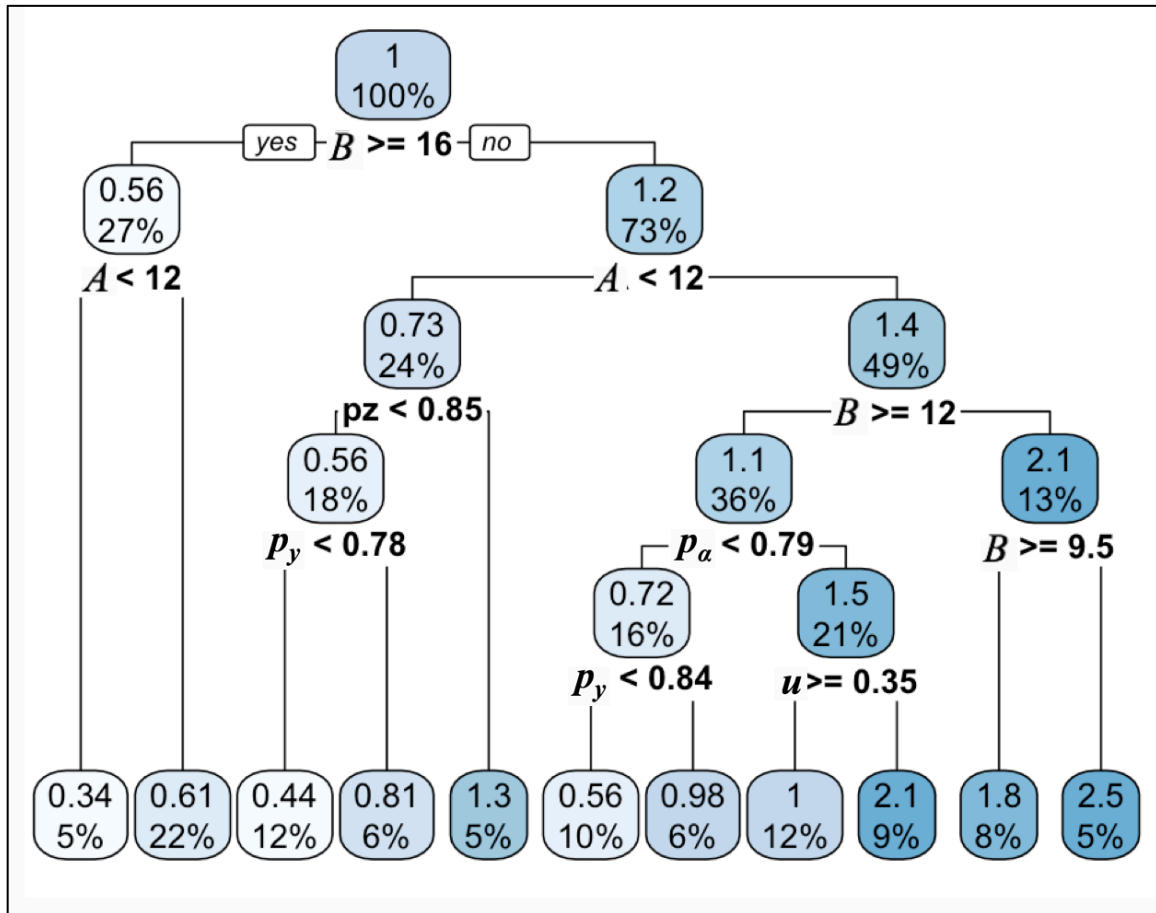


Figure 47. Tree Graph of FER (Stochastic)

3. Probability of Victory

As was done in building previous metamodels, we use stepwise regression to identify the best fitting metamodel allowing all main, quadratic, and pairwise interactions as candidate terms.

Figure 48 shows that prediction of the model is not satisfying. As is seen, the polynomial metamodel has predictions bigger than one and less than zero, which, of course, are infeasible. In addition, values that are closer to one show that most of the winning occurs when *side B* has few losses. This result reveals that our experimental region and definition of win results in few close wins. In the next step, another method, which is known as “logistic regression”, is applied. Logistic regression is a type of regression specifically designed to address binary responses, such as win and lose, see Cox (1958).

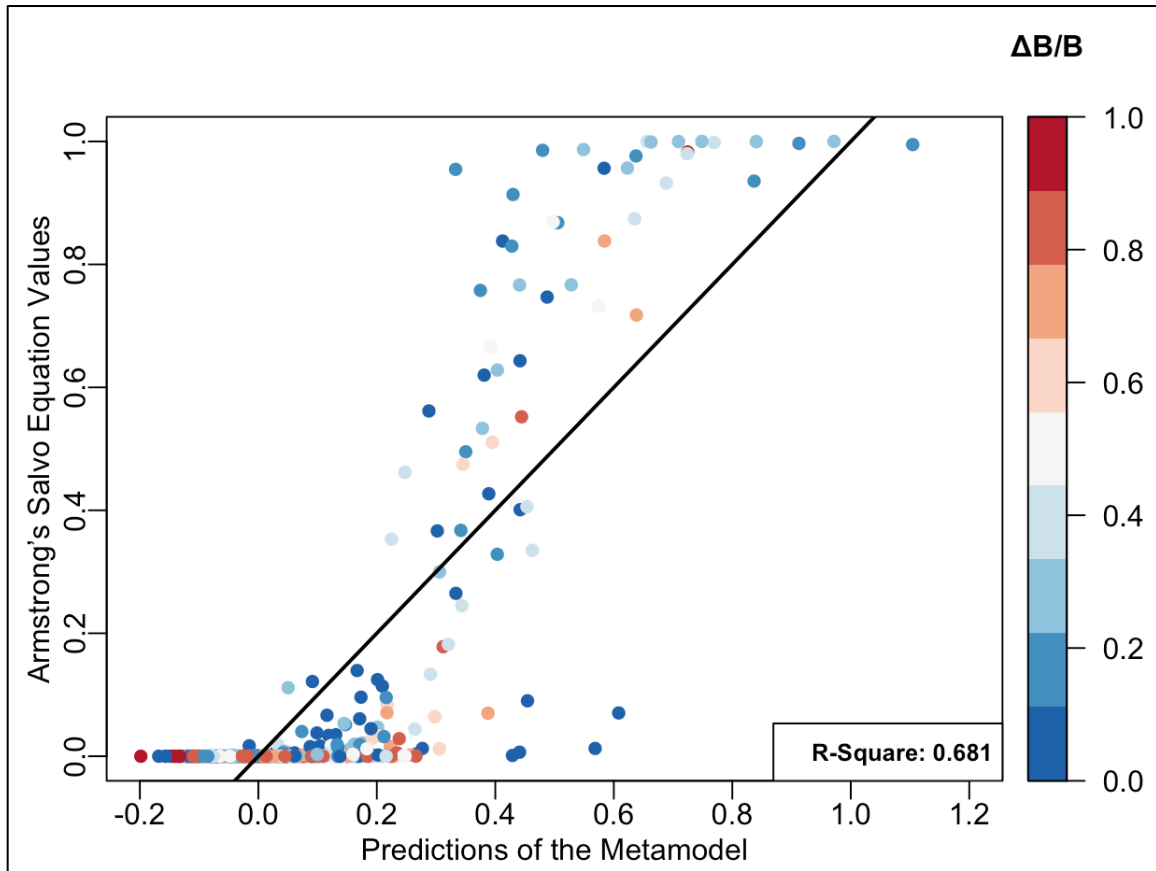


Figure 48. Comparison of Predicted Probability of Win for *Side B* Values with Actual Values (Stochastic)

After applying logistic regression on the probability of winning for *side B*, the model fits almost perfectly (Figure 49), with an R^2 close to one. See Appendix C for R's summary output.

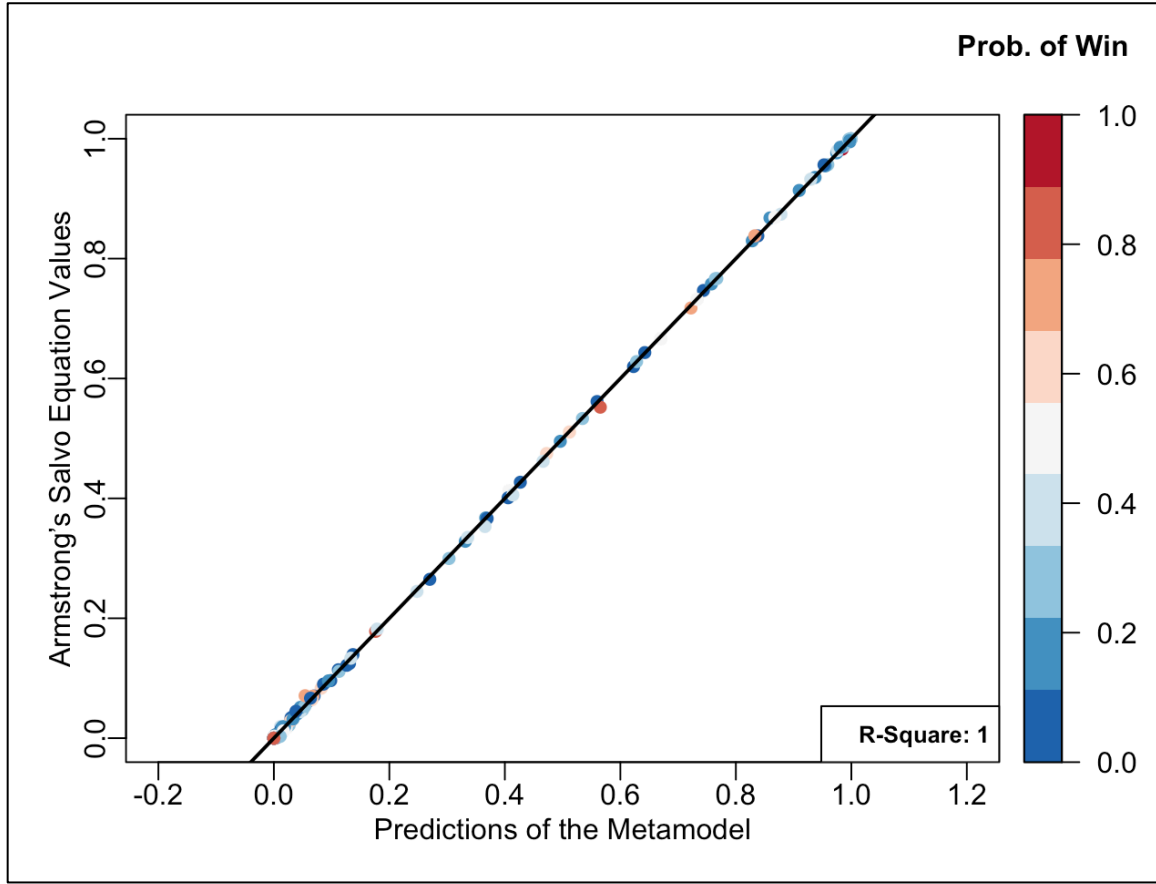


Figure 49. Predicted Winning Probability Values from Logistic Regression versus Simulation Values (Stochastic)

The prediction profiler shown in Figure 50 shows how the probability of *side B* winning changes as the factors are varied one at a time. When not being varied, the factors have fixed values of $A=14$, $B=14$, $n_\alpha=4$, $n_\beta=4$, $p_\alpha=0.9$, $p_\beta=0.9$, $n_y=1$, $n_z=1$, $p_y=0.677$, $p_z=1$, $u=0.4$, $v=0.4$, $sd_u=0.2$, and $sd_v=0.2$. The trends are similar to what we observed before, as *side A*'s numbers and capabilities increase (i.e., A , n_α , p_α , n_y , p_y , and $v \uparrow$) the probability that *side B* wins trends downward. The trends are highly nonlinear. The opposite holds true when *side B*'s numbers and capabilities increase. The grey shaded areas

indicate that changes in other factors cause the probability of winning to vary between zero and one for all of these factors. The next step is fitting a tree model and plotting it.

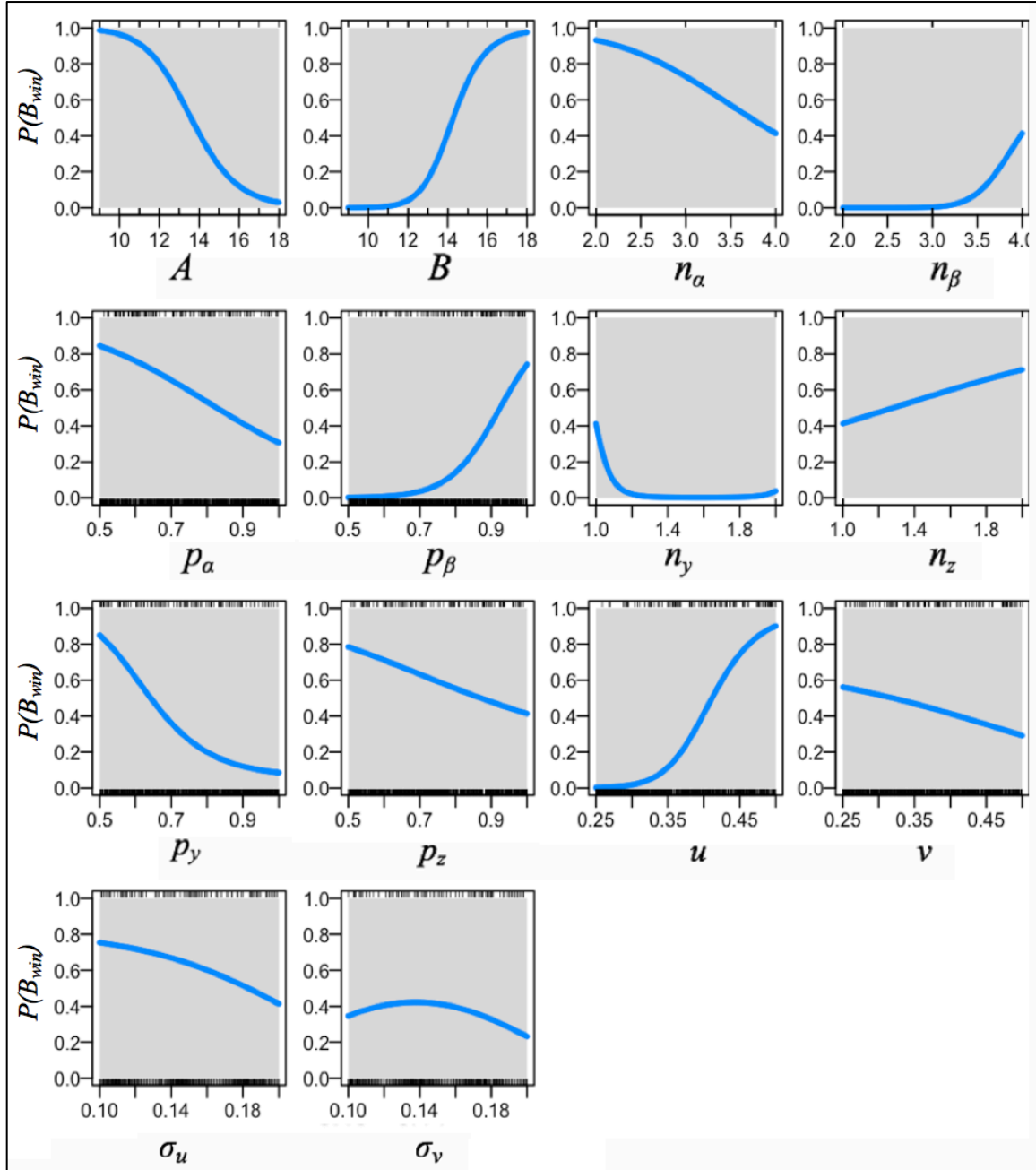


Figure 50. Prediction Profiler for *Probability of Win* (Stochastic)

The tree diagram shown in Figure 51 is based on the 165 DPs in which FER is between .2 and five. That is, the battles that are most competitive. In this region of the design space, the expected winning probability for *side B* is 0.18. The mean damage that *side A* suffers per missile (u) is the most influential factor, followed by the number of ships for *side B* (B) and the number of ASM per salvo *side B*'s ships fires (n_b).

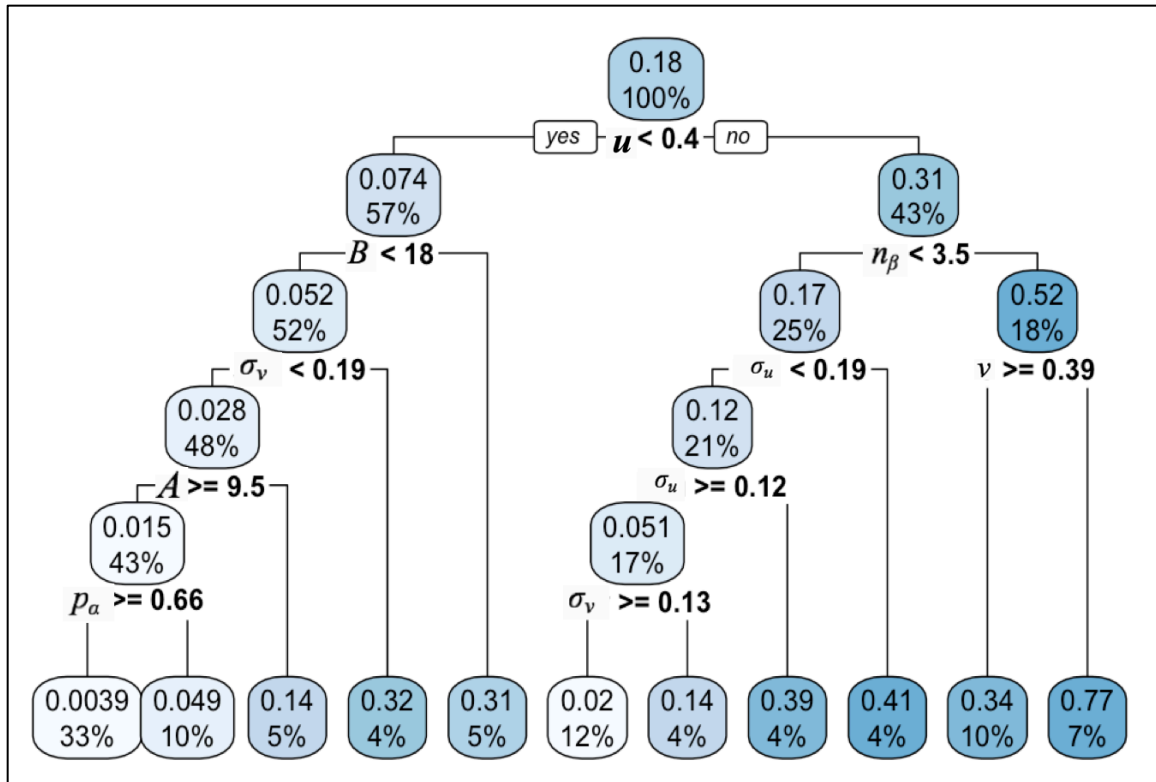


Figure 51. Tree Diagram of *Winning Probability* (Stochastic)

D. CHAPTER SUMMARY

This chapter used data farming to explore a simulation of Armstrong's stochastic salvo model. Regression and tree metamodels were fit to ΔB , FER , and the probability of *side B* winning. Insights gleaned from plots of the prediction profilers and tree models show which factors are most important and how they affect the responses. As a whole, the general findings are similar to those obtained in Hughes's original study.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. SUMMARY, CONCLUSION, AND IDEAS FOR FUTURE RESEARCH

Models are important elements in modern scientific research. They are simplified representations of the actual problem that facilitate research when physical experimentation is infeasible or severely limited. This is particularly relevant in the military domain, where systems can cost billions of dollars and field experimentation is highly constrained and extremely expensive. This thesis studies two models of naval missile warfare. The thesis consists of two main parts. The first part, Chapters I, II, and III, presents a literature review explaining in detail Hughes's deterministic salvo model and Armstrong's stochastic salvo model. The second part, Chapters IV and V, uses data farming to design, run, and analyze experiments over thousands of input combinations.

In the literature review part, we explained that the deterministic salvo model gives fixed results and the stochastic salvo model provides probabilistic information. Both models contribute to naval warfare studies from an academic perspective. The first part also introduces the reader to modern design of experiments and linear models. These tools are used in the data farming process to comprehensively explore the salvo models and build metamodels that map the inputs to outputs in understandable forms.

In the analysis part, we examined the two naval salvo models separately using data farming with sophisticated experimental designs. For both models, common responses are reformulated as readily interpretable metamodels of the inputs. The metamodels are found using stepwise regression and partition tree models. The metamodels generally explain over 97% of the variance of the experimental results. With such a good fit, the metamodels could be embedded in excel spreadsheets for ready use by sailors and analysts. The results reinforce Hughes's basic findings that modern naval warfare can have regions of instability and that numerical superiority provides a consistent advantage.

Future Researchers:

- Can model the battle for multiple salvos until one side is destroyed and determine the number of exchanges needed for neutralizing one side totally.

- Model the reinforcements needed for the losing side to have an advantage.
- Explore a broader range of the experimental space.
- Fit many different forms of metamodels using different representations of the factors and additional responses.

These are four ideas that may be modelled using the approaches developed in this research. In fact, there are infinite number of possible outcomes depending on the researchers' creativity using simulation and data farming.

APPENDIX A. CODE FOR CHAPTER IV

```
#####
#
#           CHAPTER IV. ANALYSIS PART 2: DETERMINISTIC SALVO MODEL (CLOSED-FORM)
#
#
#####

#-----INSTALLING AND LOADING REQUAIED LIBRARIES-----#
if(!require("MASS"          ))install.packages("MASS"          );library("MASS"          )
if(!require("visreg"         ))install.packages("visreg"         );library("visreg"         )
if(!require("plot3D"         ))install.packages("plot3D"         );library("plot3D"         )
if(!require("rpart.plot"     ))install.packages("rpart.plot"     );library("rpart.plot"     )
if(!require("RColorBrewer"))install.packages("RColorBrewer");library("RColorBrewer")

#####
#-----LOADING CSV DATA-----#
#####

data = read.csv( file.choose() , header=T )

#####
#-----PAIR-WISE PLOTTING AND PRINTING CORRALTATION TABLE-----#
#####
plot(data , pch = 16 , cex = 0.3 , col = "darkblue")      # Page:40 , Figure 15
round(cor(data) , digits = 3)                             # Page:41 , Figure 16

#####
#-----DEFINING NULL VARIABLE FOR A AND B-----#
#-----TO DETERMINE"Intermediate" , "Over-Defense" AND "Over-Kill"-----#
#####
data$AS = rep("intermediate" , dim(data)[1])
data$BS = rep("intermediate" , dim(data)[1])

#####
#-----START OF THE SIMULATION-----#
#####
for(i in 1:dim(data)[1])      #Loop for each DP
{
  A = data[i:i , 1]           # Number of Ships on Side A
  B = data[i:i , 2]           # Number of Ships on Side B
  a = data[i:i , 3]           # Offensive power of A
  b = data[i:i , 4]           # Offensive power of B
  y = data[i:i , 5]           # Defensive power of A
  z = data[i:i , 6]           # Defensive power of B
  w = data[i:i , 7]           # Staying power of A
  x = data[i:i , 8]           # Staying power of B

  data$deltaA[i] = ( B * b - A * y ) / w      # Calculation of The Nominal
  data$deltaB[i] = ( A * a - B * z ) / x      # Value of ΔA and ΔB.

  if(data$deltaA[i] < 0)          #####
  {                               #
    data$AS[i] = "Over-defense"  #
  }                               #
  # Determination of whether Side A
  else if(data$deltaA[i] > data$A[i]) # has an Over-defense or Over-killed. #
  {                               #
    data$AS[i] = "Over-destroyed" #
  }                               #
  #####
}
```

```

if(data$deltaB[i] < 0) #####
{
  data$BS[i] = "Over-defense" #####
}
else if(data$deltaB[i] > data$B[i]) #####
{
  data$BS[i] = "Over-destroyed" #####
}
#####

data$deltaA[i] = min(A , max(0 , data$deltaA[i])) ##### # Calculation for #
data$deltaB[i] = min(B , max(0 , data$deltaB[i])) ##### # the Actual value #
data$FER[i]=((data$deltaB[i] / B)/((data$deltaA[i] / A) ##### # of ΔA , ΔB and FER #
data$Bratio[i] = ((data$deltaB[i] / B) #####
}
#####
#-----END OF THE SIMULATION-----#
#####
data$BS = factor(data$BS) ##### # Converting the simulation destroy #
data$AS = factor(data$AS) ##### # results from string to the factor #
summary(data) ##### # Obtaining simulation statistics. #
#####

#-----NORMALITY TEST-----#
#####
shapiro.test(sample(data$deltaB , 3000))
shapiro.test(sample(data$Bratio , 3000))
shapiro.test(sample(data$FER , 3000))

write.csv(data , file = "RawDetResult.csv" , row.names = F , quote = F)

#####
#-----THE FIRST LINEAR MODEL OF THE SIMULATION BELONGING ACTUAL LOSS ON SIDE B-----#
#####
lm1=lm(deltaB~(A+B+a+b+y+z+x+w)^2+I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+I(x^2)+I
(x^2) -1 , data)
lm1 = stepAIC(lm1 , direction = "both") ##### # Stepwise Regression.
summary(lm1)
#####
#-----PLOTTING PAIRWISE SCATTER PLOT THAT COMPARES-----#
#-----THE SIMULATION'S OUTPUT WITH THE LM'S PREDICTION-----#
#####
# Page:45 , Figure 18
{
  par(mfrow = c( 1 , 1))
  par(mgp = c( 1.5 , 0.5 , 0 ))
  plot(predict(lm1,data[which(data$BS=="intermediate"),],data[which(data$BS=="intermediate"),]$deltaB,pch=16,cex=.7,col="grey",xlab="Predictions of the Metamodel",ylab="Hughes' Salvo Equation Values",xlim=c(-5,19),ylim= c(0 , 19))
  par(new=TRUE)
  plot(predict(lm1,data[which(data$BS=="Over-defense"),],data[which(data$BS=="Over-defense"),]$deltaB,pch=16,cex=.7,col="#2166AC",xlab="",ylab="",xlim= c(-5 , 19) , ylim = c(0 , 19))
  par(new=TRUE)
  plot(predict(lm1,data[which(data$BS=="Over-destroyed"),],data[which(data$BS=="Over-destroyed"),]$deltaB,pch=16,cex=.7,col="#B2182B",xlab="",ylab="",xlim=c(-5 , 19) , ylim = c(0 , 19))
  legend("bottomright" , pch = c(16 , 16 , 16) , col=c("grey" , "#B2182B" , "#2166AC") , c("Intermediate-Kill" , "Over-Kill" , "Over-Defense") , cex=.8 , text.font = 2)
  legend("topleft" , paste0("R-Square:" , round(summary(lm1)$r.squared,3)) , cex=.8 , text.font = 2)
  abline(0 , 1 , lwd=3)
}

```

```

#####
#-----APPLYING THE SECOND LINEAR MODEL-----#
#-----FOR THE RESULTS THAT END WITH INTERMEDIATE KILL-----#
#####
lm2=lm(deltaB~(A+B+a+b+y+z+x+w)^2 + I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+ I(x^2)
+ I(x^2) -1, data[which(data$AS == "intermediate"&data$BS == "intermediate") , ])
lm2 = stepAIC(lm2 , direction = "both") # Stepwise Regression.
summary(lm2)

#####
#-----SCATTER PLOT THAT COMPARES THE RESULTS OF THE SIMULATION-----#
#-----WITH THE SECOND MODEL'S PREDICTION-----#
#####
# Page:46 , Figure 19
{
  par(mfrow = c(1 , 1))
  plot(pmin(pmax(predict(lm2
    ,data[which(data$BS=="intermediate"),],type="response"),0),data[which(data$BS
    == "intermediate"),]$B),data[which(data$BS=="intermediate"),]$deltaB,pch=16,cex=
    .7,col="grey",xlab="Predictions of the Metamodel 2 ",xlim=c(-
    5,19),ylim=c(0,19),ylab="Hughes' Salvo Equation Values")
  par(new=TRUE)
  plot(pmin(pmax(predict(lm2,data[which(data$BS=="Over-defense"),],type
    ="response"),0),data[which(data$BS=="Over-defense"),]$B),data[which(data$BS
    == "Over-defense"),]$deltaB,pch=16,cex=.7,col="#2166AC",xlab="",ylab=""
    ,xlim=c(-5,19),ylim=c(0,19))
  par(new=TRUE)
  plot(pmin(pmax(predict(lm2,data[which(data$BS=="Over-destroyed"),],type=
    "response"),0),data[which(data$BS=="Over-destroyed"),]$B),data[which(data$BS
    == "Over-destroyed"),]$deltaB,pch=16,cex=.7,col="#B2182B",xlab="",ylab=""
    ,xlim=c(-5,19),ylim=c(0,19)) legend("bottomright",pch=c(16,16,16),col=c("grey"
    ,"#B2182B", "#2166AC"),c("Intermediate-Kill", "Over-Kill", "Over-Defense")
    ,text.font=2)
  legend("topleft",paste0("R-Square:",round(summary(lm(pmin(pmax(predict(lm2
    ,data,type="response"),0),data$B)~data$deltaB-1))$r.squared,3)),text.font=2)
  abline(0,1,lwd=3)
}

#####
#-----PREDICTION PROFILER OF ΔB-----#
#####
# Page:48 , Figure 20
par(mfrow = c( 2 , 4 ))
visreg(lm2 , scale = "response") # The code that executes the prediction profiler

#####
#-----TREE DIAGRAM-----#
#####
# Page:50 , Figure 22
par(mfrow=c(1 , 1))
rpart.plot(rpart(lm(deltaB~A+B+a+b+y+z+w+x,data),control=rpart.control(cp=0.006))
, cex=.7)

#####
#-----BUILDING LINEAR MODEL FOR LOSS RATIO-----#
#####
lm3=lm(Bratio~(A+B+a+b+y+z+x+w)^2+I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+I(x^2)
+I(x^2)-1,data)
lm3 = stepAIC(lm3 , direction = "both") # Stepwise Regression.
summary(lm3)

```

```
#####
#-----SCATTER PLOT THAT COMPARES THE RESULTS OF THE SIMULATION-----#
#-----WITH THE MODEL'S PREDICTION-----#
#####
# Page:51 Figure 23
par(mfrow = c(1 , 1))
plot(predict(lm3,data[which(data$BS=="intermediate"),],type="response"),data[which(d
ata$BS=="intermediate"),]$Bratio,pch=16,cex=.5,col="grey",xlab="Predictions of
the Metamodel",ylab="Hughes' Salvo Equation Values",xlim=c(-0.4,1.5),ylim=c(0,1))
par(new=TRUE)
plot(predict(lm3,data[which(data$BS=="Over-defense"),],data[which(data$BS=="Over-
defense"),]$Bratio,pch=16,cex=.5,col="#2166AC",xlab="",ylab="",xlim=c(-0.4,1.5)
,ylim=c(0,1))
par(new=TRUE)
plot(predict(lm3,data[which(data$BS=="Over-destroyed"),],data[which(data$BS=="Over-
destroyed"),]$Bratio,pch=16,cex=.5,col="#B2182B",xlab="",ylab="",xlim=c(-0.4,1.5)
,ylim=c(0,1))
legend("bottomright",pch=c(16,16,16),col=c("grey","#B2182B","#2166AC"),c("Intermedia
te-Kill","Over-Kill","Over-Defense"),cex=.8,text.font=2)
legend("topleft",paste0("R-Square:",round(summary(lm3)$r.squared,3)),cex=.8
,text.font=2)
abline(0 , 1 , lwd=2)

#####
#-----APPLYING THE SECOND LINEAR MODEL-----#
#-----FOR THE RESULTS THAT END WITH INTERMEDIATE KILL-----#
#####
lm4=lm(Bratio~(A+B+a+b+y+z+x+w)^2+I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+I(x^2)+I
(x^2)-1,data[which(data$AS=="intermediate"&data$BS=="intermediate"),])
lm4=stepAIC(lm4,direction="both")
summary(lm4)

#####
#-----SCATTER PLOT THAT COMPARES THE RESULTS OF THE SIMULATION-----#
#-----WITH THE SECOND MODEL'S PREDICTION-----#
#####
# Page:52 Figure 24
par(mfrow = c(1 , 1))
plot(pmin(predict(lm4,data[which(data$BS=="intermediate"),],type="response")
,0),1),data[which(data$BS=="intermediate"),]$Bratio,pch=16,cex=.5,col="grey",xlab
="Predictions of the Metamodel 2",ylab="Hughes' Salvo Equation Values",xlim=c(-
0.4,1.5),ylim=c(0,1))
par(new=TRUE)
plot(pmin(pmax(predict(lm4,data[which(data$BS=="Over-defense"),],type="response"),0)
,1),data[which(data$BS=="Over-defense"),]$Bratio,pch=16,cex=.7,col="#2166AC"
,xlab="",ylab="",xlim=c(-0.4,1.5),ylim=c(0,1))
par(new=TRUE)
plot(pmin(pmax(predict(lm4,data[which(data$BS=="Over-destroyed"),],type="response")
,0),1),data[which(data$BS=="Over-destroyed"),]$Bratio,pch=16,cex=.7,col="#B2182B"
,xlab="",ylab="",xlim=c(-0.4,1.5),ylim=c(0,1))
legend("bottomright",pch=c(16,16,16),col=c("grey","#B2182B","#2166AC")
,c("Intermediate-Kill","Over-Kill","Over-Defense"),cex=.8,text.font=2)
legend("topleft",paste0("R-Square:",round(summary(lm(pmin(pmax(predict(lm4,data
,type="response"),0),1)~data$Bratio))$r.squared,3)),cex=.8,text.font=2)
abline(0,1,lwd=3)

#####
#-----DRAWING THE PREDICTION PROFILER OF FIRST MODEL-----#
#-----THAT PREDICTS THE NUMBER OF LOSS SHIP ON SIDE B-----#
#####
# Page:53 Figure 25
par(mfrow = c(2 , 4))
par(oma = c(0.5 , 0.5 , 0.5 , 0.8)) # Setting the margins for best view.
par(mar = c(3 , 3.5 , 1 , 0))
par(mgp = c(1.5 , 0.5 , 0))
```

```

visreg(lm4 , scale = "response") # The code that executes the prediction profiler

#####
#-----TREE DIAGRAM-----#
#####
# Page:54 Figure 26
par(mfrow=c(1 , 1))
rpart.plot(rpart(lm(Bratio~A+B+a+b+y+z+w+x,data),control=rpart.control(cp=0.009))
,cex=.7)

#####
#-----FER SENSITIVITY ANALYSIS-----#
#####
# Page:55 Figure 27
hist(log(data[which(data$FER>0&data$FER<4000),]$FER),breaks=40,col="lightblue" ,xlab
= "log(FER)" , main = "")
summary(log(data[which(data$FER>0&data$FER<4000) , ]$FER))
shapiro.test(log(data[which(data$FER>0&data$FER<4000) , ]$FER))

# Page:56 Figure 28
t = seq(2 , 50 , 1)
rs = NULL
lg = NULL
for (i in 1:length(t))
{
  lm5=glm(FER~(A+B+a+b+y+z+w+x)^2+I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+I(w^2)+
I(x^2)-1,data[which(data$FER<t[i]&data$FER>1/t[i]),]
,family=gaussian(link="log"))
rs[i]=summary(lm(pmax(predict(lm5,data[which(data$FER<t[i]&data$FER>1/t[i]),]
,type="response"),0)~data[which(data$FER<t[i]&data$FER>1/t[i])
,$FER))$adj.r.squared
lg[i]=length(data[which(data$FER>(1/t[i])&data$FER<t[i]),]$FER)
}
plot(t , rs , type = "l" , lwd=2 , ylab = "R-squared" , xlab="FER")
plot(t , lg , type = "l" , lwd=2 , ylab = "Sample Size" , xlab="FER")

#####
#-----GENERALIZED LINEAR MODEL OF THE SIMULATION BELONGING THE FER.-----#
#####
lm5=glm(FER~(A+B+a+b+y+z+w+x)^2+I(A^2)+I(B^2)+I(a^2)+I(b^2)+I(y^2)+I(z^2)+I(w^2)
+I(x^2)-1,data[which(data$FER<=10&data$FER>(1/10)),],family=gaussian(link="log"))
lm5 = stepAIC(lm5 , direction="both")
summary(lm5)
1-(lm5$deviance/lm5$null.deviance)

summary(lm(predict(lm5,data[which(data$FER<10&data$FER>1/10),],type="response")~data
[which(data$FER<10&data$FER>1/10) , ]$FER))$r.squared

#####
#-----SCATTER PLOT THAT COMPARES THE RESULTS OF THE SIMULATION-----#
#-----WITH THE SECOND MODEL'S PREDICTION.-----#
#####
# Page:58 Figure 29
par(mfrow=c(1 , 1))
scatter2D(predict(lm5,data[which(data$FER<10&data$FER>1/10),],type="response"),data[
which(data$FER<10&data$FER>1/10),]$FER,cex=1,clim=c(0,1),colvar=data[which(data$F
ER<10&data$FER>1/10),]$deltaB/data[which(data$FER<10&data$FER>1/10),]$B,pch=16
,xlab="Predictions of the Metamodel",ylab="Hughes' Salvo Equation
Values",clab="ΔB/B",col=rev(c(brewer.pal(n=9,name="RdBu"))))
legend("topleft",paste0("R-Square:",round((summary(lm(predict(lm5
,data[which(data$FER<10&data$FER>1/10),],type="response")~data[which(data$FER<10&
data$FER>1/10),]$FER))$r.squared,3)),cex=.8, text.font=2)
abline(0,1,lwd=3)

```

```
#####
#-----PREDICTION PROFILER OF FER-----#
#####
# Page:59 Figure 30
par(mfrow=c(2 , 4))
par(oma=c(0.5 , 0.5 , 0.5 , 0.8)) # Setting the margins for best view.
par(mar=c(3 , 3.5 , 1 , 0))
par(mgp=c(1.9 , 0.5 , 0))
visreg(lm5 , scale="response")      # The code that executes the prediction profiler

#####
#-----TREE DIAGRAM OF FER-----#
#####
# Page:60 Figure 31
par(mfrow=c(1 , 1))
rpart.plot(rpart(lm(FER~A+B+a+b+y+z+w+x,data[which(data$FER<=6.7&data$FER>(1/6.7)),]
),control=rpart.control(cp=0.01)) , cex=1)
```

APPENDIX B. CODE FOR CHAPTER V

```
#####
#-----INSTALLING AND LOADING REQUIRED LIBRARIES-----#
#####
if(!require("MASS"          ))install.packages("MASS"          );library("MASS"          )
if(!require("visreg"         ))install.packages("visreg"         );library("visreg"         )
if(!require("plot3D"         ))install.packages("plot3D"         );library("plot3D"         )
if(!require("rpart.plot"     ))install.packages("rpart.plot"     );library("rpart.plot"     )
if(!require("RcolorBrewer"))install.packages("RcolorBrewer");library("RcolorBrewer")

#####
#-----LOADING CSV DATA-----#
#####
sdata = read.csv(file.choose(), header = T)

#####
#-----PAIR-WISE PLOTTING AND PRINTING CORRELATION TABLE-----#
#####
plot(sdata, pch=16, cex=.4, col= "darkblue")      # Page:65, Figure 33
round(cor(sdata ), 2)                             # Page:66, Figure 34

#####
#-----NUMBER OF TRIALS-----#
#####
n = 50000

#####
#-----START OF THE SIMULATION-----#
#####
for (i in 1:dim(sdata)[1])
{
  if(i == 1) print(Sys.time())
  A = sdata$A[i]      # Beginning force strength for side A
  B = sdata$B[i]      # Beginning force strength for side B
  na = sdata$na[i]    # Maximum # of SSMs per ship per salvo for side A
  nb = sdata$nb[i]    # Maximum # of SSMs per ship per salvo for side B
  pa = sdata$pa[i]    # Prob. of successful launch of single SSM for side A
  pb = sdata$pb[i]    # Prob. of successful launch of single SSM for side B
  ny = sdata$ny[i]    # Maximum # of SAMs per ship per salvo for side A
  nz = sdata$nz[i]    # Maximum # of SAMs per ship per salvo for side B
  py = sdata$py[i]    # Prob. of successful launch of single SAM for side A
  pz = sdata$pz[i]    # Prob. of successful launch of single SAM for side B
  u = sdata$u[i]      # Mean losses per hit suffered by side A
  v = sdata$v[i]      # Mean losses per hit suffered by side B
  sdu = sdata$sdu[i]  # Sd of mean losses per hit suffered by side A
  sdv = sdata$sdv[i]  # Sd of mean losses per hit suffered by side B

  A_offensive = rbinom(n, A * na, pa) # Offensive power of A
  B_offensive = rbinom(n, B * nb, pb) # Offensive power of B
  A_defensive = rbinom(n, A * ny, py) # Defensive power of A
  B_defensive = rbinom(n, B * nz, pz) # Defensive power of B

  NetBA = A_offensive - B_defensive # Nominal non-intercepted SSM's from A to B
  NetAB = B_offensive - A_defensive # Nominal non-intercepted SSM's from B to A

  A_damage = NULL      # Temporary variable for calculating ΔA
  B_damage = NULL      # Temporary variable for calculating ΔB
}
```

```

for (j in 1:n)
{
  if(NetAB[j] > 0)
  {
    A_damage[j] = sum(rnorm(NetAB[j], u, sdu))
  }
  else if(NetAB[j] < 0)
  {
    A_damage[j] = -sum(rnorm(-NetAB[j], u, sdu))
  }
  else
  {
    A_damage[j] = 0
  }

  if(NetBA[j] > 0)
  {
    B_damage[j] = sum(rnorm(NetBA[j], v, sdv))
  }
  else if(NetBA[j] < 0)
  {
    B_damage[j] = -sum(rnorm(-NetBA[j], v, sdv))
  }
  else
  {
    B_damage[j] = 0
  }
}

sdata$B_Loss[i] = mean(pmax(pmin(B_damage,B),0))
sdata$B_Loss_sd[i] = sd(pmax(pmin(B_damage,B),0))

sdata$FER[i] = mean((pmax(pmin(B_damage,B),0)/B)/(pmax(pmin(A_damage,A),0)/A))
sdata$FER_sd[i] = mean((pmax(pmin(B_damage,B),0)/B)/(pmax(pmin(A_damage,A),0)/A))

sdata$noLossB[i] = sum(pmax(pmin(B_damage,B),0) == 0)/n
sdata$destroyedB[i] = sum(pmax(pmin(B_damage,B),0) == B)/n
sdata$Bint[i] = sum(pmax(pmin(B_damage,B),0) != 0 & pmax(pmin(B_damage,B),0) != B)/n
sdata$Bwin[i] = sum(pmax(pmin(B_damage,B),0) != B & pmax(pmin(A_damage,A),0) == A)/n

if(i == dim(sdata)[1]) print(Sys.time())
}

#####
#-----END OF THE SIMULATION-----#
#####
write.csv(sdata, file = "RawStoResult.csv", row.names = F, quote = F)
summary(sdata)
shapiro.test(sdata$B_Loss)
shapiro.test(sdata$FER)
shapiro.test(sdata$Bwin)

#####
#-----REGRESSION MODEL OF SHIP LOSS ON SIDE B-----#
#####
# slm: stochastic linear model
slm1 = lm(B_Loss~(A + B + na + nb + pa + pb + ny + nz + py + pz + u + v + sdu +sdv )^2
+ I(A^2) + I(B^2) + I(na^2) + I(nb^2) + I(ny^2) + I(nz^2) +I(u^2) + I(v^2) +
I(pa^2) + I(pb^2) + I(py^2) + I(pz^2) + I(sdu^2) +I(sdv^2) -1, sdata)
summary(slm1)
slm1 = stepAIC(slm1, direction = "both")
summary(slm1)

```

```
#####
#-----PLOTTING PAIRWISE SCATTER PLOT THAT COMPARES-----#
#-----THE SIMULATION'S OUTPUT WITH THE LM'S PREDICTION-----#
#####
# Page:69, Figure 36
par(oma = c(0, 0, 0, 0))
par(mar = c(4, 4, 4, 4))
par(mgp = c(1.4, 0.4, 0))

scatter2D(predict(slm1,sdata,type="response"),sdata$B_Loss,colvar=sdata$B_Loss/sdata
  $B,cex=1,clim=c(0,1),pch=16,xlab="Predictions of the Metamodel",ylab="Armstrong's
  Salvo Equation Values",clab="ΔB/B",col=rev(c(brewer.pal(n = 9, name = "RdBu"))))
legend("topleft", c(paste0("R-Square:", round(summary(lm(predict(slm1, sdata, type =
  "response")~sdata$B_Loss-1))$adj.r.squared, 3))), text.font = 2)
abline(0, 1, lwd = 2)

# Page:70, Figure 37
scatter2D(predict(slm1,sdata,type="response"),sdata$B_Loss,colvar=sdata$noLossB,col=
  c(brewer.pal(n=9,name="Blues")),xlim=c(-4,19),ylim=c(0,18),cex=1,clim=c(0,1)
  ,pch=16,colkey = F)
colkey(col=c(brewer.pal(n=9,name="Blues")),clim=c(0,1),add=TRUE,side=3,length=0.9)
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:", round(summary(lm(predict(slm1
  ,sdata[which(sdata$noLossB>sdata$Bint&sdata$noLossB>sdata$destroyedB),],type="res
  ponse")~sdata[which(sdata$noLossB>sdata$Bint&sdata$noLossB>sdata$destroyedB),]$B_
  Loss-1))$r.squared,3)),paste0("N = ",sum(sdata$noLossB > sdata$Bint&sdata$noLossB
  > sdata$destroyedB))),text.font = 2)
scatter2D(predict(slm1,sdata,type="response"),sdata$B_Loss,colvar=sdata$Bint,col=c(b
  rewer.pal(n=9,name="Purples")),xlim=c(-4,19),ylim=c(0,18),cex=1,clim=c(0,1)
  ,pch=16,colkey = F)
colkey(col=c(brewer.pal(n=9,name="Purples")),clim=c(0,1),add=TRUE,side=3,length=0.9)
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:", round(summary(lm(predict(slm1
  ,sdata[which(sdata$Bint>sdata$noLossB&sdata$Bint > sdata$destroyedB),]
  ,type="response")~sdata[which(sdata$Bint>sdata$noLossB&sdata$Bint>sdata$destroyed
  B),]$B_Loss-1))$r.squared,3)),paste0("N=",sum(sdata$Bint>sdata$noLossB&sdata$Bint
  > sdata$destroyedB))),text.font = 2)
scatter2D(predict(slm1,sdata,type="response"),sdata$B_Loss,colvar=sdata$destroyedB,c
  ol=c(brewer.pal(n=9,name="Reds")),xlim=c(-4,19),ylim=c(0,18),ylab="",cex=1,clim=
  c(0,1),pch=16,colkey = F)
colkey(col= c(brewer.pal(n=9,name="Reds")),clim=c(0,1),add=TRUE,side =3,length= 0.9 )
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:", round(summary(lm(predict(slm1
  ,sdata[which(sdata$destroyedB>sdata$noLossB&sdata$destroyedB > sdata$Bint),],type
  = "response")~sdata[which(sdata$destroyedB > sdata$noLossB&sdata$destroyedB >
  sdata$Bint),]$B_Loss-1))$r.squared,3)),paste0("N = ",sum(sdata$destroyedB >
  sdata$noLossB&sdata$destroyedB > sdata$Bint))),text.font = 2)

#####
#-----REFITTING LM OF SHIP LOSS WHERE EACH SIDE HAS INTERMEDIATE KILL-----#
#####
slm2=lm(B_Loss~(A + B + na + nb + pa + pb + ny + nz + py + pz + u + v + sdu + sdv)^2
  + I(A^2) + I(B^2) + I(na^2) + I(nb^2) + I(ny^2) + I(nz^2) + I(u^2) + I(v^2) +
  I(pa^2) + I(pb^2) + I(py^2) + I(pz^2) + I(sdu^2) + I(sdv^2) -1
  ,sdata[which(sdata$B_Loss/sdata$B < = 0.95 & sdata$B_Loss/sdata$B > = 0.02 ),])
slm2 = stepAIC(slm2, direction = "both")
summary(slm2)
```

```
#####
#-----PLOTTING PAIRWISE SCATTER PLOT THAT COMPARES-----#
#-----THE SIMULATION'S OUTPUT WITH THE SECOND LM'S PREDICTION-----#
#####
# Page:71, Figure 38
par(oma = c(0, 0, 0, 0))
par(mar = c(4,4,4,4))
scatter2D(pmax(pmin(predict(slm2,sdata),sdata$B),0),sdata$B_Loss,colvar=sdata$B_Loss
/sdata$B,cex=1,clim=c(0,1),pch=16,xlab="Predictions of the Metamodel",ylab=
"Armstrong's Salvo Equation Values",clab="ΔB/B",col=rev(c(brewer.pal(n=9
,name= "RdBu"))))
legend("topleft",c(paste0("R-Square:",round(summary(lm(pmax(pmin(predict(slm2,sdata)
,sdata$B),0)~sdata$B_Loss-1))$adj.r.squared,3))),text.font = 2)
abline(0,1,lwd = 2)

# Page:72, Figure 39
scatter2D(pmin(pmax(predict(slm2,sdata,type="response"),0),sdata$B),sdata$B_Loss,col
var=sdata$noLossB,col=c(brewer.pal(n=9,name="Blues")),xlim=c(0,18),ylim=c(0,18),c
ex=1,clim = c(0,1),pch=16,colkey = F)
colkey(col=c(brewer.pal(n =9,name="Blues")),clim=c(0,1),add=TRUE,side=3,length = 0.9)
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:",round(summary(lm(pmin(pmax(predict(slm2,
sdata[which(sdata$noLossB>sdata$Bint&sdata$noLossB>sdata$destroyedB),],type="respon
se"),0),sdata[which(sdata$noLossB>sdata$Bint&sdata$noLossB>sdata$destroyedB),]$
B)~sdata[which(sdata$noLossB>sdata$Bint&sdata$noLossB>sdata$destroyedB),]$B_Loss-
1))$r.squared,3)),paste0("N = ",sum(sdata$noLossB>sdata$Bint&sdata$noLossB>
sdata$destroyedB))),text.font = 2)
scatter2D(pmin(pmax(predict(slm2,sdata,type="response"),0),sdata$B),sdata$B_Loss,col
var=sdata$Bint,col=c(brewer.pal(n=9,name="Purples")),xlim=c(0,18),ylim=c(0,18),ce
x=1,clim=c(0,1),pch=16,colkey = F)
colkey(col=c(brewer.pal(n=9,name="Purples")),clim=c(0,1),add=TRUE,side=3,length=0.9)
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:",round(summary(lm(pmin(pmax(predict(slm2
,sdata[which(sdata$Bint>sdata$noLossB&sdata$Bint>sdata$destroyedB),],type="respon
se"),0),sdata[which(sdata$Bint>sdata$noLossB&sdata$Bint>sdata$destroyedB),]$B)~sd
ata[which(sdata$Bint>sdata$noLossB&sdata$Bint>sdata$destroyedB)
,$B_Loss-1))$r.squared,3)),paste0("N = ",sum(sdata$Bint>sdata$noLossB&sdata$Bint
>sdata$destroyedB))),text.font = 2)
scatter2D(pmin(pmax(predict(slm2,sdata,type="response"),0),sdata$B),sdata$B_Loss
,colvar = sdata$destroyedB,col= c(brewer.pal(n = 9,name = "Reds")),xlim = c(0,18)
,ylim = c(0,18),ylab = "",cex=1,clim = c(0,1),pch=16,colkey = F)
colkey(col= c(brewer.pal(n = 9,name="Reds")),clim=c(0,1),add=TRUE,side= 3,length=0.9)
abline(0,1,lwd = 2)
legend("topleft",c(paste0("R-Square:",round(summary(lm(pmin(pmax(predict(slm2
,sdata[which(sdata$destroyedB>sdata$noLossB&sdata$destroyedB>sdata$Bint),],type="res
ponse"),0),sdata[which(sdata$destroyedB>sdata$noLossB&sdata$destroyedB>sdata$Bint)
,$B)~sdata[which(sdata$destroyedB>sdata$noLossB&sdata$destroyedB>sdata$Bint),]$
B_Loss-1))$r.squared,3)),paste0("N = "
,sum(sdata$destroyedB>sdata$noLossB&sdata$destroyedB>sdata$Bint)))
,text.font = 2)

#####
#-----PREDICTOR PROFILER OF ΔB-----#
#####
# Page:73, Figure 40
par(mfrow = c(4, 4))
par(oma = c(0.5, 0.5, 0.5, 0.8))
par(mar = c(3, 3.5, 1, 0))
par(mgp = c(2.1, 0.5, 0))
visreg(slm2,scale = "response")
par(mfrow = c(1, 1))
```

```

#####
#-----TREE DIAGRAM OF AB-----#
#####
# Page:75, Figure 42
rpart.plot(rpart(lm(B_Loss~ A + B + na + nb + pa + pb + ny + nz + py + pz + u + v +
  sdu + sdv ,sdata[which(sdata$Bint > = 0.42),]),control = rpart.control(cp =
  0.012)),cex=.68)

#####
#-----REGRESSION MODEL OF FER-----#
#####
# Sensitivity Analysis
# Page:76, Figure 43
hist(sdata$FER )
hist(log(sdata$FER),breaks = 40,main = "",col= "lightblue")

t = seq(2,100,1)
rs = NULL
lg = NULL
for (i in 1:length(t))
{
  print(i)
  lm5 = glm(FER~( A + B + na + nb +pa+pb + ny + nz + py + pz + u + v + sdu+ sdv )^2
    +I(A^2)+I(B^2)+I(na^2)+I(nb^2)+I(ny^2)+I(nz^2)+I(u^2)+I(v^2)+I(pa^2)+I(pb^2)
    +I(py^2) +I(pz^2)+I(sdu^2)+I(sdv^2)-1,sdata[which(sdata$FER < t[i]&sdata$FER >
    1/t[i]),],family = gaussian(link = "log"))

  rs[i]=summary(lm(predict(lm5,sdata[which(is.finite(sdata$FER)) ,]
    ,type="response")~sdata[which(is.finite(sdata$FER)) ,]$FER))$adj.r.squared
  lg[i] = length(sdata[which(sdata$FER > (1/t[i])&sdata$FER < t[i]),]$FER)
}
# Page:77, Figure 44
plot(t,rs,type = "l",lwd = 2,ylab = "R-squared",xlab = "FER")
plot(t,lg,type = "l",lwd = 2,ylab = "Sample Size",xlab = "FER")#)

slm3=glm(FER~(A+B+na+nb+pa+pb+ny+nz+py+pz+u+v+sdu+sdv)^2+I(A^2)+I(B^2)+I(na^2)+I(nb^
  2)+I(ny^2)+I(nz^2)+I(u^2)+I(v^2)+I(pa^2)+I(pb^2)+I(py^2)+I(pz^2)+I(sdu^2)+I(sdv^2)
  )-1,sdata[which(sdata$FER < 100&sdata$FER > 0),],family = gaussian(link = "log"))
slm3 = stepAIC(slm3, direction = "both")
1-(slm3$deviance/slm3$null.deviance)

#####
#-----PLOTING PAIRWISE SCATTER PLOT THAT COMPARES-----#
#-----THE SIMULATION'S OUTPUT WITH THE SECOND LM'S PREDICTION-----#
#####
# Page:78, Figure 45
par(mfrow = c(1,1))

scatter2D(predict(slm3,sdata[which(sdata$FER<100 & sdata$FER>0),],type="response")
  ,sdata[which(sdata$FER < 100&sdata$FER > 0),]$FER,colvar = sdata[which(sdata$FER
  < 100 & sdata$FER > 0),]$B_Loss/sdata[which(sdata$FER < 100&sdata$FER >
  0),]$B,cex=1,clim = c(0,1),pch=16,xlab = "Predictions of the Metamodel",ylab =
  "Armstrong's Salvo Equation Values",clab = "AB/B",col= rev(c(brewer.pal(n = 9,name
  = "RdBu"))))
legend("topleft",paste0("R-Square:",round((summary(lm(predict(slm3
  ,sdata[which(sdata$FER<100&sdata$FER>0),],type="response") ~sdata[which(sdata$FER
  < 100&sdata$FER > 0),]$FER))$r.squared),3)),cex=.8,text.font = 2)
abline(0,1,lwd = 2)

#####
#-----PREDICTOR PROFILER OF FER-----#
#####
# Page:79, Figure 46
par(mfrow = c(4, 4))

```

```

par(oma = c(0.5, 0.5, 0.5, 0.8))
par(mar = c(3, 3.5, 1, 0))
par(mgp = c(2.1, 0.5, 0))
visreg(slm3, scale = "response")

#####
#-----TREE DIAGRAM OF AB-----#
#####
# Page:80, Figure 47
par(mfrow = c(1, 1))
rpart.plot(rpart(lm(FER~ A + B + na + nb + pa + pb + ny + nz + py + pz + u + v + sdu
+ sdv, sdata[which(sdata$FER < 5&sdata$FER > 1/5),]), control = rpart.control(cp =
0.005)), cex=.85)

#####
#-----REGRESSION MODEL OF WIN-----#
#####
slm4 = lm(Bwin~( A + B + na + nb + pa + pb + ny + nz + py + pz + u + v + sdu + sdv
)^2+I(A^2)+I(B^2)+I(na^2)+I(nb^2)+I(ny^2)+I(nz^2)+I(u^2)+I(v^2)+I(pa^2)+I(pb^2)+I
(py^2) +I(pz^2)+I(sdu^2)+I(sdv^2)-1, sdata)
slm4 = stepAIC(slm4, direction = "both")
summary(slm4)

#####
#-----PLOTING PAIRWISE SCATTER PLOT-----#
#####
# Page:81, Figure 48

scatter2D(predict(slm4, sdata, type="response"), sdata$Bwin, colvar=sdata$B_Loss/sdata$B
, cex=1, clim=c(0,1), pch=16, xlab="Predictions of the Metamodel", ylab = "Armstrong's
Salvo Equation Values", clab="AB/B", xlim=c(-0.2,1.2), col=rev(c(brewer.pal(n=9, name
= "RdBu"))))
legend("bottomright", paste0("R-Square:", round(summary(slm4)$adj.r.squared,3)), cex=.8
, text.font = 2)
abline(0,1, lwd = 2)

#####
#-----SECOND LINEAR MODEL OF BWIN WITH LOGISTIC REGRESSION-----#
#####
slm5 = glm(Bwin~( A + B + na + nb + pa + pb + ny + nz + py + pz + u + v + sdu + sdv
)^2+I(A^2)+I(B^2)+I(na^2)+I(nb^2)+I(ny^2)+I(nz^2)+I(u^2)+I(v^2)+I(pa^2)+I(pb^2)+I
(py^2) +I(pz^2)+I(sdu^2)+I(sdv^2)-1, sdata, family = binomial)
slm5 = stepAIC(slm5, direction = "backward")
summary(slm5)
1-(slm5$deviance/slm5$null.deviance)

#####
#-----PLOTING PAIRWISE SCATTER PLOT THAT COMPARES-----#
#-----THE SIMULATION'S OUTPUT WITH THE SECOND LM'S PREDICTION-----#
#####
# Page:82, Figure 49
scatter2D(predict(slm5, sdata, type = "response"), sdata$Bwin, colvar =
sdata$B_Loss/sdata$B, cex=1, clim = c(0,1), pch=16, xlab = "Predictions of the
Metamodel", ylab = "Armstrong's Salvo Equation Values", clab = "Prob. of Win", xlim
= c(-0.2,1.2), col= rev(c(brewer.pal(n = 9, name = "RdBu"))))
legend("bottomright", paste0("R-Square:", round(1-(slm5$deviance/slm5$null.deviance)
,3)), cex=.8, text.font = 2)
abline(0,1, lwd = 2)

#####
#-----PREDICTOR PROFILER OF FER-----#
#####
# Page:83, Figure 50
par(mfrow = c(4, 4))
par(oma = c(0.5, 0.5, 0.5, 0.8))

```

```

par(mar = c(3 , 3.5, 1 , 0 ))
par(mgp = c(2.1, 0.5, 0 ))
visreg(slm5,scale = "response",cond = list(A = 14,B = 14, na = 4, nb = 4, pa = 0.9,
      pb = 0.9, ny = 1, nz = 1,py = 0.677, pz = 1, u = 0.4,v= 0.4, sdu = 0.2, sdu = 0.2))

#####
#-----TREE DIAGRAM OF ΔB-----#
#####
# Page:84, Figure 51
par(mfrow = c(1, 1))
rpart.plot(rpart(lm(Bwin~A+B+na+nb+pa+pb+ny+nz+py+pz+u+v+sdu+sdv,sdata[which(sdata$F
      ER < 5&sdata$FER > 1/5),]),control = rpart.control(cp = 0.0005)),cex=.75)

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. REGRESSION RESULTS

Metamodel of ΔB – (Experiment of Deterministic Salvo Model)

```
lm(formula = deltaB ~ A + B + a + b + y + z + x + w + I(A^2) +
    I(B^2) + I(a^2) + I(z^2) + I(x^2) + A:B + A:a + A:b + A:x +
    B:a + B:z + B:x + B:w + a:b + a:y + a:z + a:x + y:w + z:x -
    1, data = data)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-7.1351	-0.6427	-0.1324	0.6695	6.9928

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	0.230201	0.067119	3.430	0.000609	***
B	0.196654	0.068092	2.888	0.003893	**
a	2.278047	0.211645	10.764	< 2e-16	***
b	-0.280286	0.104157	-2.691	0.007147	**
y	-0.620302	0.226061	-2.744	0.006092	**
z	-1.642046	0.369202	-4.448	8.87e-06	***
x	-0.874266	0.319342	-2.738	0.006208	**
w	0.002345	0.146972	0.016	0.987272	
I(A^2)	-0.011289	0.002223	-5.078	3.95e-07	***
I(B^2)	-0.019558	0.002237	-8.742	< 2e-16	***
I(a^2)	-0.146254	0.024785	-5.901	3.85e-09	***
I(z^2)	0.426923	0.099194	4.304	1.71e-05	***
I(x^2)	0.346732	0.051567	6.724	1.96e-11	***
A:B	0.033520	0.001836	18.253	< 2e-16	***
A:a	0.195309	0.006548	29.829	< 2e-16	***
A:b	0.016457	0.006613	2.489	0.012855	*
A:x	-0.093095	0.009567	-9.731	< 2e-16	***
B:a	0.105202	0.006578	15.993	< 2e-16	***
B:z	-0.315690	0.013029	-24.230	< 2e-16	***
B:x	-0.042047	0.009645	-4.360	1.33e-05	***
B:w	-0.016987	0.009437	-1.800	0.071914	.
a:b	0.033735	0.021944	1.537	0.124278	
a:y	0.078243	0.043880	1.783	0.074626	.
a:z	-0.215281	0.044273	-4.863	1.19e-06	***
a:x	-0.644617	0.032702	-19.712	< 2e-16	***
y:w	0.136556	0.065795	2.075	0.037991	*
z:x	0.646420	0.066141	9.773	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.198 on 5093 degrees of freedom

Multiple R-squared: 0.9706, Adjusted R-squared: 0.9705

F-statistic: 6236 on 27 and 5093 DF, p-value: < 2.2e-16

Metamodel of ΔB Based on Intermediate Values – (Experiment of Deterministic Salvo Model)

Call:

```
lm(formula = deltaB ~ A + B + a + b + y + z + x + w + I(A^2) +
    I(a^2) + I(x^2) + A:B + A:a + A:z + A:x + B:a + B:z + B:x +
    a:z + a:x + b:y + b:x + z:x + x:w - 1, data = data[which(data$AS ==
    "intermediate" & data$BS == "intermediate"), ])
```

Residuals:

Min	1Q	Median	3Q	Max
-1.08702	-0.09127	-0.00635	0.09652	0.99471

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	0.8535792	0.0155252	54.980	< 2e-16	***
B	-0.4123537	0.0112432	-36.676	< 2e-16	***
a	4.6710000	0.0484422	96.424	< 2e-16	***
b	0.0212334	0.0291961	0.727	0.46712	
y	-0.0395714	0.0304868	-1.298	0.19440	
z	-4.8813994	0.0769332	-63.450	< 2e-16	***
x	-1.9656355	0.0635351	-30.938	< 2e-16	***
w	-0.0562501	0.0328882	-1.710	0.08731	.
I(A^2)	-0.0027425	0.0005565	-4.929	8.77e-07	***
I(a^2)	-0.0315903	0.0070227	-4.498	7.13e-06	***
I(x^2)	0.6421448	0.0111959	57.355	< 2e-16	***
A:B	0.0027998	0.0005449	5.138	2.97e-07	***
A:a	0.3248243	0.0023299	139.414	< 2e-16	***
A:z	0.0181613	0.0037820	4.802	1.65e-06	***
A:x	-0.2653347	0.0025937	-102.302	< 2e-16	***
B:a	0.0056764	0.0021806	2.603	0.00929	**
B:z	-0.3352830	0.0036794	-91.125	< 2e-16	***
B:x	0.1171102	0.0025943	45.142	< 2e-16	***
a:z	0.0605828	0.0140043	4.326	1.57e-05	***
a:x	-1.4663996	0.0092652	-158.270	< 2e-16	***
b:y	0.0160443	0.0111484	1.439	0.15022	
b:x	-0.0120277	0.0083309	-1.444	0.14892	
z:x	1.4522697	0.0164733	88.159	< 2e-16	***
x:w	0.0190971	0.0105064	1.818	0.06922	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1929 on 2768 degrees of freedom

Multiple R-squared: 0.9992, Adjusted R-squared: 0.9992

F-statistic: 1.41e+05 on 24 and 2768 DF, p-value: < 2.2e-16

Metamodel of $\Delta B/B$ – (Experiment of Deterministic Salvo Model)

```
lm(formula = Bratio ~ A + B + a + b + y + z + x + w + I(A^2) +
    I(B^2) + I(a^2) + I(y^2) + I(z^2) + I(x^2) + A:B + A:a +
    A:b + A:z + A:x + B:a + B:b + B:z + B:x + B:w + a:x + b:x +
    b:w + y:w + z:x - 1, data = data)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.59051	-0.04813	-0.01413	0.05087	0.42646

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	0.0762978	0.0052102	14.644	< 2e-16	***
B	-0.0518952	0.0052994	-9.793	< 2e-16	***
a	0.5046508	0.0148409	34.004	< 2e-16	***
b	0.0262696	0.0136510	1.924	0.054363	.
y	0.0098391	0.0241077	0.408	0.683197	
z	-0.3948010	0.0295698	-13.352	< 2e-16	***
x	-0.1268196	0.0251852	-5.035	4.93e-07	***
w	0.0195417	0.0123722	1.579	0.114287	
I(A^2)	-0.0012616	0.0001701	-7.414	1.42e-13	***
I(B^2)	0.0019470	0.0001714	11.361	< 2e-16	***
I(a^2)	-0.0204953	0.0018909	-10.839	< 2e-16	***
I(y^2)	-0.0139576	0.0075596	-1.846	0.064900	.
I(z^2)	0.0275307	0.0075643	3.640	0.000276	***
I(x^2)	0.0201963	0.0039794	5.075	4.01e-07	***
A:B	-0.0009784	0.0001403	-6.974	3.46e-12	***
A:a	0.0114717	0.0004995	22.964	< 2e-16	***
A:b	0.0009524	0.0005032	1.893	0.058454	.
A:z	0.0033415	0.0010055	3.323	0.000896	***
A:x	-0.0056988	0.0007348	-7.756	1.05e-14	***
B:a	-0.0115795	0.0005016	-23.083	< 2e-16	***
B:b	-0.0009928	0.0005005	-1.984	0.047359	*
B:z	-0.0063925	0.0009950	-6.424	1.45e-10	***
B:x	0.0031817	0.0007396	4.302	1.72e-05	***
B:w	-0.0017325	0.0007220	-2.400	0.016453	*
a:x	-0.0426515	0.0024977	-17.076	< 2e-16	***
b:x	-0.0042354	0.0025248	-1.677	0.093511	.
b:w	-0.0036272	0.0024803	-1.462	0.143692	
y:w	0.0079564	0.0050243	1.584	0.113354	
z:x	0.0391228	0.0050541	7.741	1.18e-14	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09127 on 5091 degrees of freedom

Multiple R-squared: 0.974, Adjusted R-squared: 0.9738

F-statistic: 6565 on 29 and 5091 DF, p-value: < 2.2e-16

Metamodel of $\Delta B/B$ Based on Intermediate Values – (Experiment of Deterministic Salvo Model)

Call:

```
lm(formula = Bratio ~ A + B + a + b + y + z + x + w + I(A^2) +
    I(B^2) + I(a^2) + I(b^2) + I(x^2) + A:B + A:a + A:z + A:x +
    A:w + B:a + B:b + B:y + B:z + B:x + B:w + a:b + a:z + a:x +
    a:w + b:y + b:x + b:w + y:z + y:x + z:x + z:w + x:w - 1,
    data = data[which(data$AS == "intermediate" & data$BS ==
        "intermediate"), ])
```

Residuals:

	Min	1Q	Median	3Q	Max
	-0.090380	-0.009605	-0.001904	0.008447	0.119705

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	1.314e-01	1.586e-03	82.853	< 2e-16	***
B	-1.105e-01	1.570e-03	-70.411	< 2e-16	***
a	7.046e-01	5.052e-03	139.454	< 2e-16	***
b	2.452e-02	4.728e-03	5.187	2.29e-07	***
y	3.031e-02	7.105e-03	4.267	2.05e-05	***
z	-6.531e-01	8.046e-03	-81.166	< 2e-16	***
x	-2.106e-01	6.781e-03	-31.058	< 2e-16	***
w	4.599e-02	4.477e-03	10.271	< 2e-16	***
I(A^2)	-5.040e-04	4.885e-05	-10.317	< 2e-16	***
I(B^2)	4.183e-03	5.177e-05	80.806	< 2e-16	***
I(a^2)	-7.755e-03	6.116e-04	-12.679	< 2e-16	***
I(b^2)	-8.438e-04	5.724e-04	-1.474	0.14057	
I(x^2)	4.175e-02	1.006e-03	41.515	< 2e-16	***
A:B	-4.021e-03	5.381e-05	-74.717	< 2e-16	***
A:a	2.334e-02	2.107e-04	110.774	< 2e-16	***
A:z	1.073e-03	3.357e-04	3.196	0.00141	**
A:x	-1.894e-02	2.305e-04	-82.162	< 2e-16	***
A:w	-7.347e-04	2.235e-04	-3.287	0.00103	**
B:a	-2.300e-02	2.108e-04	-109.119	< 2e-16	***
B:b	-2.439e-04	1.727e-04	-1.412	0.15795	
B:y	-5.911e-04	2.923e-04	-2.023	0.04322	*
B:z	-1.996e-03	3.309e-04	-6.031	1.85e-09	***
B:x	1.687e-02	2.346e-04	71.906	< 2e-16	***
B:w	-6.959e-04	2.221e-04	-3.133	0.00175	**
a:b	-1.255e-03	5.517e-04	-2.274	0.02303	*
a:z	8.621e-03	1.233e-03	6.992	3.38e-12	***
a:x	-1.073e-01	8.141e-04	-131.744	< 2e-16	***
a:w	-1.588e-03	7.750e-04	-2.049	0.04055	*
b:y	-1.598e-03	1.077e-03	-1.483	0.13809	
b:x	-1.446e-03	7.862e-04	-1.840	0.06589	.
b:w	-2.304e-03	7.379e-04	-3.122	0.00182	**
y:z	-5.751e-03	1.780e-03	-3.231	0.00125	**
y:x	-3.661e-03	1.385e-03	-2.644	0.00824	**
z:x	1.057e-01	1.430e-03	73.913	< 2e-16	***
z:w	-2.816e-03	1.372e-03	-2.053	0.04017	*
x:w	-4.054e-03	9.538e-04	-4.250	2.21e-05	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.01667 on 2756 degrees of freedom
Multiple R-squared: 0.999, Adjusted R-squared: 0.9989
F-statistic: 7.346e+04 on 36 and 2756 DF, p-value: < 2.2e-16

Metamodel of *FER* Based on Intermediate Values – (Experiment of Deterministic Salvo Model)

Call:
glm(formula = FER ~ A + B + a + b + y + z + w + x + I(A^2) +
I(B^2) + I(a^2) + I(b^2) + I(y^2) + I(z^2) + I(w^2) + I(x^2) +
A:B + A:a + A:b + A:y + A:z + A:w + A:x + B:a + B:b + B:y +
B:z + B:w + B:x + a:b + a:y + a:z + a:x + b:y + b:z + b:w +
b:x + y:z + y:w + y:x + z:w + z:x - 1, family = gaussian(link =
"log"),
data = data[which(data\$FER <= 10 & data\$FER > (1/10)),])

Deviance Residuals:

Min	1Q	Median	3Q	Max
-2.2771	-0.1093	0.0678	0.1671	3.9568

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	0.5985661	0.0164853	36.309	< 2e-16	***
B	-0.6235580	0.0175300	-35.571	< 2e-16	***
a	1.3184311	0.0453854	29.050	< 2e-16	***
b	-2.3711369	0.0689508	-34.389	< 2e-16	***
y	2.9630241	0.1075552	27.549	< 2e-16	***
z	-1.2266388	0.0766962	-15.993	< 2e-16	***
w	0.6770460	0.0575480	11.765	< 2e-16	***
x	-0.3471367	0.0592375	-5.860	5.09e-09	***
I(A^2)	-0.0141660	0.0005522	-25.652	< 2e-16	***
I(B^2)	0.0100001	0.0005577	17.931	< 2e-16	***
I(a^2)	-0.2927608	0.0053986	-54.229	< 2e-16	***
I(b^2)	0.4585547	0.0076492	59.948	< 2e-16	***
I(y^2)	0.4996161	0.0221221	22.585	< 2e-16	***
I(z^2)	-0.1522390	0.0162081	-9.393	< 2e-16	***
I(w^2)	-0.0639663	0.0089379	-7.157	1.02e-12	***
I(x^2)	-0.0184237	0.0084954	-2.169	0.030180	*
A:B	0.0056235	0.0007253	7.754	1.18e-14	***
A:a	-0.0770922	0.0022379	-34.448	< 2e-16	***
A:b	-0.0510446	0.0026484	-19.274	< 2e-16	***
A:y	0.1659883	0.0045169	36.748	< 2e-16	***
A:z	0.0836234	0.0035621	23.476	< 2e-16	***
A:w	-0.0061927	0.0023457	-2.640	0.008331	**
A:x	0.0360270	0.0024605	14.642	< 2e-16	***
B:a	0.0848740	0.0023255	36.497	< 2e-16	***
B:b	0.0492590	0.0026891	18.318	< 2e-16	***
B:y	-0.1762930	0.0044377	-39.727	< 2e-16	***
B:z	-0.0898242	0.0035968	-24.973	< 2e-16	***
B:w	0.0091257	0.0023760	3.841	0.000125	***
B:x	-0.0429113	0.0024406	-17.582	< 2e-16	***
a:b	0.0984635	0.0088399	11.138	< 2e-16	***
a:y	-0.1471904	0.0142676	-10.316	< 2e-16	***
a:z	0.4333551	0.0104661	41.406	< 2e-16	***

```

a:x      0.1164555  0.0070859  16.435 < 2e-16 ***
b:y     -1.0138683  0.0172177 -58.885 < 2e-16 ***
b:z     -0.0698626  0.0138033  -5.061 4.39e-07 ***
b:w      0.0408654  0.0097619   4.186 2.91e-05 ***
b:x     -0.0507963  0.0095371  -5.326 1.07e-07 ***
y:z      0.1062503  0.0217522   4.885 1.09e-06 ***
y:w     -0.0962869  0.0160043  -6.016 1.98e-09 ***
y:x      0.0923888  0.0148874   6.206 6.12e-10 ***
z:w      0.0192316  0.0109755   1.752 0.079828 .
z:x     -0.0830650  0.0114549  -7.251 5.12e-13 ***
---

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.1365019)

```

Null deviance: 13628.76 on 3307 degrees of freedom
Residual deviance: 445.68 on 3265 degrees of freedom
AIC: 2843

```

Number of Fisher Scoring iterations: 6

Metamodel of AB – (Experiment of Stochastic Salvo Model)

Call:

```

lm(formula = B_Loss ~ A + B + na + nb + pa + pb + ny + nz + py +
    pz + u + v + sdu + sdv + I(A^2) + I(B^2) + I(na^2) + I(v^2) +
    I(pb^2) + I(pz^2) + A:B + A:na + A:pa + A:ny + A:pz + A:v +
    A:sdu + B:na + B:pa + B:nz + B:py + B:u + B:sdv + na:pa +
    na:nz + na:pz + na:v + nb:ny + nb:pz + pa:pb + pa:pz + pa:v +
    pb:ny + ny:u + nz:pz + nz:v + nz:sdu + nz:sdv + py:pz + py:u +
    py:sdu + pz:v + pz:sdv + sdu:sdv - 1, data = sdata)

```

Residuals:

	Min	1Q	Median	3Q	Max
	-6.1890	-0.5765	-0.0592	0.6311	3.7313

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
A	0.090760	0.263460	0.344	0.730635
B	-0.319003	0.256712	-1.243	0.214632
na	-0.539413	0.925517	-0.583	0.560299
nb	0.063383	0.368920	0.172	0.863665
pa	-11.442139	3.749599	-3.052	0.002409 **
pb	1.595556	4.138692	0.386	0.700030
ny	-0.333685	0.901909	-0.370	0.711571
nz	5.387828	1.128334	4.775	2.42e-06 ***
py	-6.438813	3.290273	-1.957	0.050964 .
pz	3.379206	5.221924	0.647	0.517879
u	-6.396545	4.932497	-1.297	0.195347
v	13.741816	9.704773	1.416	0.157460
sdu	-40.319371	14.760939	-2.731	0.006548 **
sdv	20.757853	14.840362	1.399	0.162567
I(A^2)	-0.023050	0.006748	-3.416	0.000692 ***
I(B^2)	-0.010610	0.006805	-1.559	0.119637

I (na^2)	-0.312370	0.103064	-3.031	0.002577	**
I (v^2)	-26.969585	10.243602	-2.633	0.008754	**
I (pb^2)	-4.284320	2.566364	-1.669	0.095719	.
I (pz^2)	6.954996	2.570171	2.706	0.007063	**
A:B	0.014833	0.005590	2.654	0.008242	**
A:na	0.099498	0.020763	4.792	2.23e-06	***
A:pa	0.557011	0.115148	4.837	1.80e-06	***
A:ny	0.063276	0.033542	1.886	0.059860	.
A:pz	-0.252724	0.115358	-2.191	0.028971	*
A:v	0.503590	0.228557	2.203	0.028068	*
A:sdu	0.868237	0.565806	1.535	0.125594	.
B:na	0.097482	0.021121	4.615	5.10e-06	***
B:pa	0.264542	0.120853	2.189	0.029104	*
B:nz	-0.228102	0.034065	-6.696	6.29e-11	***
B:py	0.180903	0.110259	1.641	0.101543	.
B:u	0.396021	0.246786	1.605	0.109247	.
B:sdv	-1.362276	0.576350	-2.364	0.018514	*
na:pa	2.737459	0.426689	6.416	3.50e-10	***
na:nz	-0.175755	0.121124	-1.451	0.147454	.
na:pz	-0.822087	0.423302	-1.942	0.052740	.
na:v	3.823960	0.867137	4.410	1.29e-05	***
nb:ny	-0.274180	0.119946	-2.286	0.022719	*
nb:pz	0.594925	0.420490	1.415	0.157796	.
pa:pb	3.319997	2.346801	1.415	0.157839	.
pa:pz	-5.498914	2.347887	-2.342	0.019605	*
pa:v	13.915998	4.679184	2.974	0.003094	**
pb:ny	1.631409	0.689392	2.366	0.018375	*
ny:u	-2.473579	1.337708	-1.849	0.065085	.
nz:pz	-1.858825	0.680449	-2.732	0.006543	**
nz:v	-3.680238	1.361372	-2.703	0.007120	**
nz:sdu	-5.225304	3.374184	-1.549	0.122166	.
nz:sdv	-7.103907	3.340808	-2.126	0.034004	*
py:pz	-3.636981	2.360003	-1.541	0.123985	.
py:u	7.529285	4.751195	1.585	0.113722	.
py:sdu	24.868804	11.471741	2.168	0.030686	*
pz:v	-7.744945	4.742154	-1.633	0.103112	.
pz:sdv	-16.054195	11.452169	-1.402	0.161638	.
sdu:sdv	118.698699	58.209499	2.039	0.042007	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.065 on 458 degrees of freedom
Multiple R-squared: 0.978, Adjusted R-squared: 0.9754
F-statistic: 376.3 on 54 and 458 DF, p-value: < 2.2e-16

Metamodel of ΔB Based on Intermediate Values – (Experiment of Stochastic Salvo Model)

Call:

```
lm(formula = B_Loss ~ A + B + na + nb + pa + pb + ny + nz + py +
    pz + u + v + sdu + sdv + I(B^2) + I(ny^2) + I(sdu^2) + A:na +
    A:nb + A:pa + A:pb + A:nz + A:py + A:v + A:sdv + B:nb + B:nz +
    B:pz + B:v + B:sdv + na:pa + na:nz + na:v + nb:pa + nb:py +
    nb:u + pa:nz + pa:pz + pa:v + pb:nz + pb:py + pb:u + pb:sdv +
    ny:nz + ny:pz + ny:u + nz:pz + nz:v + pz:v + u:sdv + v:sdv +
    sdu:sdv + na:pb - 1, data = sdata[which(sdata$B_Loss/sdata$B <=
    0.95 & sdata$B_Loss/sdata$B >= 0.02), ])
```

Residuals:

	Min	1Q	Median	3Q	Max
	-0.69629	-0.10864	-0.00138	0.10829	0.62807

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	-1.296457	0.047707	-27.175	< 2e-16	***
B	0.818343	0.055851	14.652	< 2e-16	***
na	-6.468792	0.152603	-42.390	< 2e-16	***
nb	0.423107	0.146931	2.880	0.004232	**
pa	-23.793129	0.832786	-28.571	< 2e-16	***
pb	0.240106	0.751081	0.320	0.749406	
ny	13.092872	1.969602	6.647	1.18e-10	***
nz	7.909226	0.286423	27.614	< 2e-16	***
py	0.258312	0.548100	0.471	0.637737	
pz	14.690217	0.727092	20.204	< 2e-16	***
u	3.494403	1.231422	2.838	0.004815	**
v	-22.486306	1.757537	-12.794	< 2e-16	***
sdu	-14.355836	4.436087	-3.236	0.001330	**
sdv	6.124624	4.375487	1.400	0.162492	
I(B^2)	-0.003571	0.001509	-2.366	0.018531	*
I(ny^2)	-4.222842	0.664665	-6.353	6.72e-10	***
I(sdu^2)	25.022201	13.284508	1.884	0.060472	.
A:na	0.263861	0.004911	53.730	< 2e-16	***
A:nb	-0.009631	0.004465	-2.157	0.031714	*
A:pa	1.083751	0.026710	40.575	< 2e-16	***
A:pb	-0.068501	0.025195	-2.719	0.006886	**
A:nz	-0.037485	0.007666	-4.890	1.56e-06	***
A:py	-0.055554	0.025890	-2.146	0.032597	*
A:v	1.873204	0.053287	35.153	< 2e-16	***
A:sdv	-0.239905	0.126763	-1.893	0.059263	.
B:nb	-0.006966	0.004788	-1.455	0.146590	
B:nz	-0.246032	0.007984	-30.816	< 2e-16	***
B:pz	-0.519527	0.026334	-19.728	< 2e-16	***
B:v	-0.837440	0.052148	-16.059	< 2e-16	***
B:sdv	-0.185192	0.125499	-1.476	0.140958	
na:pa	4.888733	0.103012	47.458	< 2e-16	***
na:nz	-0.136365	0.029259	-4.661	4.52e-06	***
na:v	8.524161	0.199300	42.771	< 2e-16	***
nb:pa	-0.274791	0.096207	-2.856	0.004549	**
nb:py	-0.171869	0.092152	-1.865	0.063029	.
nb:u	0.392172	0.173574	2.259	0.024488	*

```

pa:nz      -1.055150    0.160822   -6.561 1.98e-10 ***
pa:pz      -1.794693    0.534116   -3.360 0.000867 ***
pa:v       33.448590    1.096519   30.504 < 2e-16 ***
pb:nz       0.358764    0.141830    2.530 0.011870 *
pb:py       1.093682    0.502490    2.177 0.030200 *
pb:u       -1.886291    0.924668   -2.040 0.042122 *
pb:sdv      3.730445    2.416915    1.543 0.123641
ny:nz       0.064492    0.041557    1.552 0.121611
ny:pz      -0.388218    0.147069   -2.640 0.008678 **
ny:u       -0.658619    0.287006   -2.295 0.022350 *
nz:pz      -4.561445    0.153040  -29.805 < 2e-16 ***
nz:v       -9.294009    0.296356  -31.361 < 2e-16 ***
pz:v      -16.039193    1.064656  -15.065 < 2e-16 ***
u:sdv     -13.231011    4.944754   -2.676 0.007815 **
v:sdv     -13.627428    5.101720   -2.671 0.007921 **
sdu:sdv    46.589156   12.776013    3.647 0.000307 ***
na:pb      -0.168787    0.091922   -1.836 0.067196 .

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1944 on 342 degrees of freedom
Multiple R-squared: 0.9992, Adjusted R-squared: 0.9991
F-statistic: 7883 on 53 and 342 DF, p-value: < 2.2e-16

Metamodel of *FER*– (Experiment of Stochastic Salvo Model)

```

glm(formula = FER ~ A + B + na + nb + pa + pb + ny + nz + py +
    pz + u + v + sdu + sdv + I(A^2) + I(B^2) + I(na^2) + I(nb^2) +
    I(u^2) + I(v^2) + I(pa^2) + I(pb^2) + I(sdu^2) + I(sdv^2) +
    A:B + A:na + A:nb + A:pa + A:pb + A:ny + A:nz + A:py + A:pz +
    A:v + A:sdu + A:sdv + B:na + B:nb + B:pa + B:pb + B:ny +
    B:nz + B:py + B:pz + B:u + B:v + B:sdu + B:sdv + na:nb +
    na:pa + na:pb + na:nz + na:pz + na:u + na:v + na:sdv + nb:pb +
    nb:ny + nb:nz + nb:py + nb:pz + nb:u + nb:v + nb:sdu + pa:ny +
    pa:nz + pa:pz + pa:u + pa:v + pa:sdv + pb:ny + pb:nz + pb:py +
    pb:u + pb:v + ny:py + ny:u + ny:sdu + nz:pz + nz:v + pz:u +
    pz:v + pz:sdu + pz:sdv + u:sdu + u:sdv + v:sdu + v:sdv +
    sdu:sdv - 1, family = gaussian(link = "log"), data =
sdata[which(sdata$FER <
    100 & sdata$FER > 0), ])

```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.129136	-0.024533	-0.002639	0.016845	0.140494

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
A	0.598713	0.060927	9.827	< 2e-16 ***
B	-0.708379	0.064586	-10.968	< 2e-16 ***
na	2.038120	0.179408	11.360	< 2e-16 ***
nb	-1.556555	0.221268	-7.035	7.78e-11 ***
pa	8.689229	1.004411	8.651	9.93e-15 ***
pb	-7.860494	1.369882	-5.738	5.56e-08 ***
ny	1.301504	0.386983	3.363	0.000990 ***

nz	-2.477461	0.262330	-9.444	< 2e-16	***
py	4.833005	0.611512	7.903	6.83e-13	***
pz	-6.355411	0.804931	-7.896	7.14e-13	***
u	-13.195575	1.757296	-7.509	6.03e-12	***
v	14.056943	2.217312	6.340	2.87e-09	***
sdu	15.148124	4.657135	3.253	0.001429	**
sdv	-1.015091	4.095370	-0.248	0.804598	
I(A^2)	-0.010661	0.001463	-7.289	1.99e-11	***
I(B^2)	0.005998	0.001309	4.584	9.91e-06	***
I(na^2)	-0.221033	0.014378	-15.372	< 2e-16	***
I(nb^2)	0.151333	0.021606	7.004	9.16e-11	***
I(u^2)	8.787389	1.426923	6.158	7.14e-09	***
I(v^2)	-8.708848	1.391033	-6.261	4.28e-09	***
I(pa^2)	-3.004415	0.404920	-7.420	9.81e-12	***
I(pb^2)	2.334480	0.654980	3.564	0.000498	***
I(sdu^2)	-18.867707	9.184972	-2.054	0.041791	*
I(sdv^2)	22.140635	9.266925	2.389	0.018198	*
A:B	0.004311	0.001550	2.781	0.006148	**
A:na	-0.054277	0.003882	-13.981	< 2e-16	***
A:nb	-0.036497	0.006015	-6.067	1.12e-08	***
A:pa	-0.222519	0.024885	-8.942	1.86e-15	***
A:pb	-0.049011	0.034464	-1.422	0.157197	
A:ny	0.064154	0.010243	6.263	4.22e-09	***
A:nz	0.087207	0.007085	12.309	< 2e-16	***
A:py	0.166332	0.020176	8.244	1.01e-13	***
A:pz	0.176540	0.021622	8.165	1.58e-13	***
A:v	-0.162269	0.047512	-3.415	0.000831	***
A:sdu	-0.188411	0.096850	-1.945	0.053703	.
A:sdv	0.157100	0.105974	1.482	0.140440	
B:na	0.057758	0.004387	13.166	< 2e-16	***
B:nb	0.046822	0.005728	8.174	1.50e-13	***
B:pa	0.175593	0.022873	7.677	2.40e-12	***
B:pb	0.144623	0.032780	4.412	2.01e-05	***
B:ny	-0.070744	0.010187	-6.945	1.25e-10	***
B:nz	-0.087939	0.006811	-12.911	< 2e-16	***
B:py	-0.187701	0.022360	-8.394	4.30e-14	***
B:pz	-0.186326	0.023661	-7.875	8.02e-13	***
B:u	0.135539	0.040736	3.327	0.001117	**
B:v	0.223544	0.041663	5.366	3.20e-07	***
B:sdu	0.122605	0.092945	1.319	0.189256	
B:sdv	-0.162823	0.102769	-1.584	0.115339	
na:nb	0.036353	0.014501	2.507	0.013307	*
na:pa	-1.135588	0.077934	-14.571	< 2e-16	***
na:pb	0.249937	0.092981	2.688	0.008046	**
na:nz	0.351071	0.020518	17.110	< 2e-16	***
na:pz	0.703321	0.080307	8.758	5.37e-15	***
na:u	-0.535489	0.125004	-4.284	3.37e-05	***
na:v	-1.024657	0.140346	-7.301	1.87e-11	***
na:sdv	0.537304	0.366507	1.466	0.144856	
nb:pb	0.814568	0.164773	4.944	2.13e-06	***
nb:ny	-0.288159	0.047567	-6.058	1.17e-08	***
nb:nz	-0.033728	0.021596	-1.562	0.120575	
nb:py	-0.669987	0.082211	-8.150	1.72e-13	***
nb:pz	-0.183921	0.065167	-2.822	0.005452	**
nb:u	0.413189	0.148954	2.774	0.006283	**

nb:v	0.361255	0.138836	2.602	0.010250	*
nb:sdu	-0.728930	0.386872	-1.884	0.061587	.
pa:ny	0.158827	0.136432	1.164	0.246317	.
pa:nz	1.694248	0.127599	13.278	< 2e-16	***
pa:pz	3.287617	0.458199	7.175	3.68e-11	***
pa:u	-1.566261	0.838411	-1.868	0.063804	.
pa:v	-2.639175	0.875003	-3.016	0.003034	**
pa:sdv	-4.420245	2.266213	-1.950	0.053086	.
pb:ny	-0.886997	0.233568	-3.798	0.000216	***
pb:nz	0.229229	0.134473	1.705	0.090447	.
pb:py	-3.320924	0.587032	-5.657	8.19e-08	***
pb:u	2.994076	0.925402	3.235	0.001511	**
pb:v	-1.593803	0.872851	-1.826	0.069955	.
ny:py	1.015481	0.136298	7.450	8.30e-12	***
ny:u	0.587278	0.220574	2.662	0.008651	**
ny:sdu	-0.965424	0.590117	-1.636	0.104057	.
nz:pz	-1.103249	0.120046	-9.190	4.39e-16	***
nz:v	0.337200	0.251531	1.341	0.182194	.
pz:u	1.230042	0.622825	1.975	0.050214	.
pz:v	2.728135	0.696616	3.916	0.000139	***
pz:sdu	3.966799	1.755249	2.260	0.025345	*
pz:sdv	3.764299	1.999723	1.882	0.061826	.
u:sdu	-6.566471	2.884238	-2.277	0.024301	*
u:sdv	4.049675	3.282694	1.234	0.219374	.
v:sdu	-5.949936	3.765110	-1.580	0.116267	.
v:sdv	-14.118111	3.962793	-3.563	0.000500	***
sdu:sdv	-22.164039	8.900798	-2.490	0.013923	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.002507301)

Null deviance: 179.94000 on 231 degrees of freedom
Residual deviance: 0.35604 on 142 degrees of freedom
AIC: -660.21

Number of Fisher Scoring iterations: 7

Metamodel of *Winning Probability*– (Experiment of Stochastic Salvo Model)

Call:

```
lm(formula = Bwin ~ A + B + na + nb + pa + pb + ny + nz + py +
    pz + u + v + sdu + sdv + I(A^2) + I(nb^2) + I(ny^2) + I(pb^2) +
    I(py^2) + A:B + A:na + A:nb + A:pb + A:ny + A:nz + A:py +
    A:u + A:sdu + B:na + B:nb + B:pb + B:ny + B:nz + B:pz + B:u +
    B:v + na:nb + na:pb + na:nz + na:v + nb:pa + nb:pb + nb:ny +
    nb:nz + nb:py + nb:pz + nb:u + pa:pb + pa:nz + pa:pz + pb:ny +
    pb:nz + pb:py + pb:u + ny:nz + ny:pz + ny:u + nz:u + py:u +
    py:v + v:sdv - 1, data = sdata)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-0.55577	-0.07271	0.00352	0.06497	0.62170

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
A	0.0168619	0.0382393	0.441	0.659455	
B	-0.0770638	0.0290132	-2.656	0.008184	**
na	-0.0749519	0.0899175	-0.834	0.404969	
nb	-0.6608190	0.1451565	-4.552	6.83e-06	***
pa	-1.2698095	0.4105891	-3.093	0.002107	**
pb	-2.4284333	0.7580043	-3.204	0.001453	**
ny	4.0230887	1.3753397	2.925	0.003616	**
nz	0.2868545	0.1656431	1.732	0.084000	.
py	0.8310824	0.7573298	1.097	0.273059	
pz	0.4384174	0.3825465	1.146	0.252382	
u	-2.9268869	1.0069058	-2.907	0.003831	**
v	0.3924499	0.8514827	0.461	0.645091	
sdu	-2.9294586	1.0596097	-2.765	0.005932	**
sdv	-2.0169754	1.1789967	-1.711	0.087814	.
I (A^2)	0.0019594	0.0008994	2.178	0.029892	*
I (nb^2)	0.0609851	0.0139440	4.374	1.52e-05	***
I (ny^2)	-1.2099686	0.4734005	-2.556	0.010918	*
I (pb^2)	0.6879160	0.3535982	1.945	0.052339	.
I (py^2)	0.5714434	0.3605485	1.585	0.113684	
A:B	-0.0014267	0.0007515	-1.899	0.058263	.
A:na	-0.0062982	0.0027791	-2.266	0.023910	*
A:nb	-0.0174339	0.0027798	-6.272	8.37e-10	***
A:pb	-0.0686397	0.0152886	-4.490	9.07e-06	***
A:ny	0.0084704	0.0045091	1.879	0.060955	.
A:nz	0.0146424	0.0045355	3.228	0.001336	**
A:py	0.0245928	0.0157620	1.560	0.119401	
A:u	-0.0756102	0.0317234	-2.383	0.017566	*
A:sdu	0.2035379	0.0770506	2.642	0.008538	**
B:na	0.0054790	0.0028412	1.928	0.054426	.
B:nb	0.0204424	0.0028940	7.064	6.18e-12	***
B:pb	0.0825039	0.0159559	5.171	3.51e-07	***
B:ny	-0.0070456	0.0045790	-1.539	0.124580	
B:nz	-0.0157110	0.0046253	-3.397	0.000742	***
B:pz	-0.0568697	0.0156814	-3.627	0.000320	***
B:u	0.1042219	0.0331147	3.147	0.001757	**
B:v	0.0417636	0.0304089	1.373	0.170310	
na:nb	0.0160704	0.0099791	1.610	0.108011	
na:pb	0.0937407	0.0556056	1.686	0.092523	.
na:nz	0.0437815	0.0163292	2.681	0.007605	**
na:v	-0.1965963	0.1148950	-1.711	0.087751	.
nb:pa	0.1180311	0.0569138	2.074	0.038659	*
nb:pb	0.4009861	0.0559785	7.163	3.23e-12	***
nb:ny	-0.0577707	0.0159682	-3.618	0.000331	***
nb:nz	-0.0587969	0.0161587	-3.639	0.000306	***
nb:py	-0.1904252	0.0572145	-3.328	0.000945	***
nb:pz	-0.1228877	0.0573910	-2.141	0.032790	*
nb:u	0.8668149	0.1094927	7.917	1.91e-14	***
pa:pb	0.4637216	0.3138620	1.477	0.140248	
pa:nz	0.1938566	0.0891437	2.175	0.030175	*
pa:pz	0.4975095	0.3204068	1.553	0.121186	
pb:ny	-0.3660610	0.0932019	-3.928	9.92e-05	***
pb:nz	-0.4582127	0.0902553	-5.077	5.62e-07	***
pb:py	-0.7661659	0.3184530	-2.406	0.016534	*

```

pb:u      4.2658034  0.6203419   6.877 2.05e-11 ***
ny:nz     0.0527837  0.0258569   2.041 0.041795 *
ny:pz     0.1327022  0.0915368   1.450 0.147833
ny:u      -0.5646093  0.1816408  -3.108 0.002000 **
nz:u      -0.4437631  0.1858859  -2.387 0.017384 *
py:u      -1.3655867  0.6674598  -2.046 0.041341 *
py:v      -1.2685498  0.6169166  -2.056 0.040331 *
v:sdv     4.7382709  3.1150636   1.521 0.128940
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1414 on 451 degrees of freedom
Multiple R-squared:  0.7191, Adjusted R-squared:  0.6811
F-statistic: 18.93 on 61 and 451 DF,  p-value: < 2.2e-16

```

Metamodel of *Winning Probability* with Logistic Regression– (Experiment of Stochastic Salvo Model)

```

Call:
glm(formula = Bwin ~ (A + B + na + nb + pa + pb + ny + nz + py +
  pz + u + v + sdu + sdv)^2 + I(A^2) + I(B^2) + I(na^2) + I(nb^2) +
  I(ny^2) + I(nz^2) + I(u^2) + I(v^2) + I(pa^2) + I(pb^2) +
  I(py^2) + I(pz^2) + I(sdu^2) + I(sdv^2) - 1, family = binomial,
  data = sdata)

```

```

Deviance Residuals:
      Min       1Q   Median       3Q      Max
-0.089823 -0.000105  0.000000  0.000000  0.116315

```

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	z value	Pr(> z)
A	-13.9797	24.1495	-0.579	0.563
B	7.8907	26.5971	0.297	0.767
na	-5.0600	77.0580	-0.066	0.948
nb	43.9513	106.9398	0.411	0.681
pa	10.9161	540.4758	0.020	0.984
pb	90.5189	463.2461	0.195	0.845
ny	-260.0779	1199.6797	-0.217	0.828
nz	-1.6760	99.1298	-0.017	0.987
py	-29.1744	415.4562	-0.070	0.944
pz	49.8723	455.5997	0.109	0.913
u	222.8053	861.6111	0.259	0.796
v	33.9849	604.4306	0.056	0.955
sdu	164.5948	1620.9920	0.102	0.919
sdv	30.2850	2363.1228	0.013	0.990
I (A^2)	-0.1635	0.6351	-0.257	0.797
I (B^2)	-0.1736	0.5282	-0.329	0.742
I (na^2)	0.3600	3.2520	0.111	0.912
I (nb^2)	-4.6077	11.2171	-0.411	0.681
I (ny^2)	77.3528	385.4642	0.201	0.841
I (nz^2)	NA	NA	NA	NA
I (u^2)	-130.7269	512.4751	-0.255	0.799
I (v^2)	-47.2222	422.2134	-0.112	0.911
I (pa^2)	-19.2887	101.5593	-0.190	0.849

I (pb^2)	-15.6007	123.5692	-0.126	0.900
I (py^2)	-16.6803	162.1549	-0.103	0.918
I (pz^2)	-24.4385	138.1007	-0.177	0.860
I (sdu^2)	-35.9448	2286.2779	-0.016	0.987
I (sdv^2)	-117.3433	2594.3629	-0.045	0.964
A:B	0.3648	0.7459	0.489	0.625
A:na	-0.7096	1.5873	-0.447	0.655
A:nb	1.5237	4.3926	0.347	0.729
A:pa	-0.9547	9.9601	-0.096	0.924
A:pb	3.6386	13.4463	0.271	0.787
A:ny	-0.5537	4.0161	-0.138	0.890
A:nz	0.6318	1.9857	0.318	0.750
A:py	-3.2234	9.6142	-0.335	0.737
A:pz	2.5842	9.5350	0.271	0.786
A:u	11.9862	21.8004	0.550	0.582
A:v	1.1290	14.5059	0.078	0.938
A:sdu	8.8436	36.2097	0.244	0.807
A:sdv	-6.3962	40.6061	-0.158	0.875
B:na	0.5184	1.1865	0.437	0.662
B:nb	-1.1844	2.4881	-0.476	0.634
B:pa	1.5560	9.6581	0.161	0.872
B:pb	-2.3915	9.1102	-0.263	0.793
B:ny	0.6224	2.7852	0.223	0.823
B:nz	-0.5317	1.6961	-0.314	0.754
B:py	2.8787	7.1116	0.405	0.686
B:pz	-2.3912	6.3110	-0.379	0.705
B:u	-6.9255	18.3670	-0.377	0.706
B:v	0.8727	12.3457	0.071	0.944
B:sdu	-5.5365	35.7073	-0.155	0.877
B:sdv	2.2712	37.6322	0.060	0.952
na:nb	1.2844	4.5810	0.280	0.779
na:pa	-6.2444	27.2309	-0.229	0.819
na:pb	10.0297	22.6889	0.442	0.658
na:ny	-2.3895	5.8581	-0.408	0.683
na:nz	2.1347	5.8792	0.363	0.717
na:py	3.9728	20.5351	0.193	0.847
na:pz	0.7187	17.4475	0.041	0.967
na:u	0.4364	55.2549	0.008	0.994
na:v	-3.5896	29.3001	-0.123	0.902
na:sdu	-17.8621	128.0810	-0.139	0.889
na:sdv	-12.6264	96.8459	-0.130	0.896
nb:pa	-3.6692	39.7808	-0.092	0.927
nb:pb	-8.4313	46.8887	-0.180	0.857
nb:ny	2.2259	11.3531	0.196	0.845
nb:nz	-0.4999	10.9415	-0.046	0.964
nb:py	7.9918	38.9691	0.205	0.838
nb:pz	-3.2706	44.9232	-0.073	0.942
nb:u	-36.4649	92.1577	-0.396	0.692
nb:v	7.8775	80.5020	0.098	0.922
nb:sdu	-21.6471	144.0024	-0.150	0.881
nb:sdv	57.5304	188.4603	0.305	0.760
pa:pb	15.4176	179.8417	0.086	0.932
pa:ny	3.5629	31.3454	0.114	0.910
pa:nz	10.4400	32.2772	0.323	0.746
pa:py	-29.5575	127.3527	-0.232	0.816

pa:pz	26.7543	169.3165	0.158	0.874
pa:u	31.5788	226.0910	0.140	0.889
pa:v	9.9801	276.7432	0.036	0.971
pa:sdu	-115.9085	572.1491	-0.203	0.839
pa:sdv	72.7353	587.2425	0.124	0.901
pb:ny	3.6490	45.3591	0.080	0.936
pb:nz	-10.3009	42.5642	-0.242	0.809
pb:py	25.7975	154.9897	0.166	0.868
pb:pz	-37.6190	166.6403	-0.226	0.821
pb:u	-60.3805	331.6039	-0.182	0.856
pb:v	-7.0274	244.7995	-0.029	0.977
pb:sdu	-41.3235	728.1384	-0.057	0.955
pb:sdv	-83.5601	920.4030	-0.091	0.928
ny:nz	2.8302	8.8158	0.321	0.748
ny:py	-9.8446	36.8526	-0.267	0.789
ny:pz	5.6205	39.6908	0.142	0.887
ny:u	10.4063	103.7099	0.100	0.920
ny:v	1.7901	62.7372	0.029	0.977
ny:sdu	59.8677	199.0103	0.301	0.764
ny:sdv	-2.0933	136.8025	-0.015	0.988
nz:py	-1.7037	36.1662	-0.047	0.962
nz:pz	-3.9658	30.6404	-0.129	0.897
nz:u	-11.6485	69.3543	-0.168	0.867
nz:v	-0.9790	62.9543	-0.016	0.988
nz:sdu	13.9403	182.8195	0.076	0.939
nz:sdv	-7.5343	156.3947	-0.048	0.962
py:pz	9.7304	144.7011	0.067	0.946
py:u	43.0941	288.7111	0.149	0.881
py:v	35.7187	251.1773	0.142	0.887
py:sdu	-120.6730	492.2065	-0.245	0.806
py:sdv	-29.7420	515.4537	-0.058	0.954
pz:u	2.3360	303.6398	0.008	0.994
pz:v	-8.5014	252.2257	-0.034	0.973
pz:sdu	41.5990	563.1294	0.074	0.941
pz:sdv	-66.6725	514.1481	-0.130	0.897
u:v	-32.6585	477.5968	-0.068	0.945
u:sdu	58.7415	1033.4505	0.057	0.955
u:sdv	31.2606	1335.3596	0.023	0.981
v:sdu	-124.4536	911.6912	-0.137	0.891
v:sdv	-212.4551	1251.5358	-0.170	0.865
sdu:sdv	302.0700	3160.9346	0.096	0.924

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 643.69977 on 512 degrees of freedom
Residual deviance: 0.13991 on 394 degrees of freedom
AIC: 271.07

Number of Fisher Scoring iterations: 18

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716–723.
- Armstrong, M. J. (2005). A stochastic salvo model for naval surface combat. *Operations Research*, 53(5), 830–841.
- Armstrong, M. J. (2011). A verification study of the stochastic salvo combat model. *Annals of Operations Research*, 186 (1), 23–38.
- Beall, T. R. (1990). *The development of a naval battle model and its validation using historical data* (Master's thesis). Retrieved from <http://hdl.handle.net/10945/34842>
- Cares, J. R. (1990). *The fundamentals of salvo warfare* (Master's thesis). Retrieved from <http://hdl.handle.net/10945/34844>
- Cioppa, T. M. & Lucas, T. W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics*, 49(1), 45–55.
- Chase, J.V. (1902). Sea fights: A mathematical investigation of the effect of superiority of force in combats upon the sea. Naval War College Archives, RG 8, Box 109, XTAV.
- Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society*, 20(2), 215–242.
- Faraway, J. J. (2015a). *Linear models with R second edition*. Boca Raton, FL: CRC Press.
- Faraway, J. J. (2015b). *Extending the regression models with R: Generalized linear, mixed effects and nonparametric regression models second edition*. Boca Raton, FL: CRC Press.
- Hughes, W. P. (1995). A salvo model of warships in missile combat used to evaluate their staying power. *Naval Research Logistics*, 42(2), 267–289.
- Hernandez, A. S., Lucas, T. W. & Carlyle, M. (2012). Constructing nearly orthogonal Latin hypercubes for any nonsaturated run-variable combination. *ACM Transactions on Modeling and Computer Simulation*, 22(4), 20:1-20:17.
- Keys, A., Fidanza, F., Karvonen, M. J., Kimura, N. & Taylor, H. L. (1972). Indices of relative weight and obesity. *Journal of Chronic Diseases*, 25, 329-343.
- Lanchester, F.W. (1914). Aircraft in warfare: The dawn of the fourth arm - No. V. *The Principle of Concentration. Engineering*, 98, 422–423.

- Li, C. H. (2018). Verification analysis of Armstrong's stochastic salvo equations using data farming (Master's thesis). Retrieved from <http://hdl.handle.net/10945/59709>.
- Lucas T. W. (2000). The stochastic versus deterministic argument for combat simulations: Tales of when the average won't do! *Military Operations Research*, 5(3), 9–28.
- Lucas, T. W. & McGunnigle, J. E. (2002). When is model complexity too much? Illustrating the benefits of simple models with Hughes' salvo equations. *Naval Research Logistics*, 50(3), 197–217.
- Lucas, T.W., Kelton, W.D. , Sanchez, S.M., Sanchez P.J. (2015) Changing the Paradigm: Simulation, Often the Method of First Resort,” *Naval Research Logistics*, 62, 2015, pp. 293-303
- Lucas, T. W., Kelton, W. D., Sanchez, S. M., Sanchez, P. J. & Anderson B.L. (2015). Changing the paradigm: Simulation, often the method of first resort. *Naval Research Logistics*, 62(4), 293-303.
- MacCalman, A. D., Vieira, H. & Lucas, T. W. (2017) Second order nearly orthogonal Latin hypercubes for exploring complex stochastic simulations. *Journal of Simulation*, 11(2), 137-150.
- Montgomery, D. C. (2005). *Design and Analysis of Experiments*, 6th edition. New York: Wiley.
- Osipov, M. (1915). The influence of the numerical strength of engaged forces on their casualties (Helmbold, R.L. & Rehm, A. S. Trans. Report Nos. CAA-RP-91-2). Retrieved from DTIC website: <http://www.dtic.mil/dtic/tr/fulltext/u2/a241534.pdf>
- Rstudio. (2018, May 15). Why RStudio? Retrieved from <https://www.rstudio.com/about/>
- Rokach, L., Maimon, O. (2008). *Data mining with decision trees: theory and applications*. Singapore: World Scientific Publishing.
- Sanchez, S. M., and Sanchez, P. J. (2005). Very large fractional factorials and central composite designs. *ACM Transactions on Modeling and Computer Simulation*, 15(4): 362–377.
- Sanchez, S. M., and Wan, H. (2012). Work smarter, not harder: A tutorial on designing and conducting simulation experiments. *Proceedings of the 2012 Winter Simulation Conference*, eds. C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, Piscataway, New Jersey: Institute of Electrical and Electronic Engineers, Inc., 1552–1563.
- Vieira, Jr., H. (2012). NOB_Mixed_512DP_v2.xlsx design spreadsheet [Computer software]. Retrieved from <https://my.nps.edu/web/seed/software-downloads>

- Vieira, Jr., H., Sanchez, S. M., Kienitz, K. H. & Belderrain, M. C. M. (2011a). Generating and improving orthogonal designs by using mixed integer programming. *European Journal of Operational Research* 215, 629–638.
- Vieira, Jr., H., Sanchez, S. M., Kienitz, K. H. & Belderrain, M. C. M. (2011b). Improved efficient, nearly orthogonal, nearly balanced mixed designs. *Proceedings of the 2011 Winter Simulation Conference*, 3605-3616.
- Vieira, Jr., H., Sanchez, S. M., Kienitz, K. H. & Belderrain, M. C. M. (2013). Efficient, nearly orthogonal-and-balanced, mixed designs: An effective way to conduct trade-off analyses via simulation. *Journal of Simulation*, 7, 264–275
- Wackerly, D., Mendenhall, W., & Scheaffer, R. (2007). *Mathematical statistics with applications*, 7th ed. Belmont, CA: Brooks/Cole.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California