

ABSTRACT

ZHANG, KE. Statistical Analysis of Compounds Using OBSTree and Compound Mixtures Using Nonlinear Models. (Under the direction of Dr. Jacqueline M. Hughes-Oliver.)

A novel tree-structured data-mining tool is proposed to automatically search for and find high performance classification and important quantitative structure-activity relationships (QSARs) hidden in large data sets. The presence or absence of multiple chemical features is implemented to identify more informative splitting rules. A stochastic optimization scheme combined with a new splitting criterion and a post-trimming procedure is developed to find global optimum splitting variables. The algorithm is also ready to serve as a powerful predictive tool for estimating unknown biological activities according to the chemical structures.

We also investigate several statistical issues in chemical mixture studies. With a thorough review of different concepts of additivity the criteria for evaluating a concept of additivity are discussed and a particular concept of additivity is generalized to some complicated studies. A nonlinear dose-response model is initially developed for binary mixtures. The model can be easily generalized to a mixture of M chemicals. Different types of test statistics under multiplicity adjustments are proposed to test the interactions.

**Statistical Analysis of Compounds Using OBSTree and Compound
Mixtures Using Nonlinear Models**

by

Ke Zhang

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

STATISTICS

Raleigh

2006

APPROVED BY:

Dr. Jacqueline M. Hughes-Oliver
Chair of Advisory Committee

Dr. Sidney Stanley Young

Dr. Cavell Brownie

Dr. David A. Dickey

To

Yinglin

Meixian

and

Ning

Biography

Ke Zhang was born in Wuhan, China on Aug. 31, 1979. He received his undergraduate degree in biotechnology from Wuhan University and his master's degree in biomathematics from North Carolina State University. He was selected as a Merck & Co., Inc. BARDS (Biostatistics and Research Decision Sciences) Graduate Fellow during the 2003-2004 academic year and was selected as a Gertrude M. Cox Academic Achievement Award Fellow, Outstanding Ph.D. Candidate for the 2004-2005 academic year. He is a member of Phi Kappa Phi Honor Society and Mu Sigma Rho, National Statistical Honor Society.

Acknowledgements

I would never have been able to finish my dissertation without the guidance of my committee members, help from friends, and support from my family.

There is one person above all others who deserves my deepest gratitude and respect for her continued support during the writing of this dissertation: my advisor and academic parent, Dr. Jacqueline M. Hughes-Oliver. I want to thank her for her excellent expert guidance, inspiration, patience, and especially her confidence in my ability.

My great thanks also go to Dr. Sidney Stanley Young who led many thought-provoking conversations with me and taught me lots of practical issues beyond the textbooks. I would like to thank Dr. Cavell Brownie, Dr. David A. Dickey and Dr. William H. Swallow for serving on my committee. Their insights and suggestions have contributed a great deal to my research.

I am also grateful to my research group members, Bingming Yi from Glaxo-SmithKline, Katja Remlinger from GlaxoSmtihKline, Atina Brooks, Jiajun Liu and Haojun Ouyang from North Carolina State University. They gave me invaluable assistance in bi-weekly group meetings.

In addition, I would like to thank Dr. Sastry G. Pantula, Dr. William H. Swallow, and Dr. Leonard A. Stefanski for their constant help in providing my financial support.

Finally, and always, my personal special thanks go to my family. My parents, Yinglin Zhang and Meixian Xu, always support me and encourage me with their best wishes. I am so proud of being their son. I thank my wife, Ning Ding, for her always

being there cheering me up and standing by me.

Contents

List of Tables.....	ix
List of Figures.....	xi
1 Introduction.....	1
2 Summary of Research and Results.....	7
2.1 Summary of OBSTree	7
2.1.1 Motivation and Main Features	7
2.1.2 Application Results	9
2.2 Summary of Concepts of Additivity in Chemical Mixture Studies . . .	11
2.3 Summary of Statistical Nonlinear Models and Hypothesis Tests in Chem- ical Mixture Studies	14
2.3.1 Motivations and New Nonlinear Models	14
2.3.2 Hypothesis Tests	16
2.3.3 Application Results	21
I Statistical Analysis of Compounds Using OBSTree ...	24
3 Recursive Partitioning and Its Extensions.....	25
3.1 Introduction	25
3.2 SCAM	26
3.3 Random Forest	28
3.4 BFIRM and RP/SA	28
3.5 Motivations of OBSTree Development	31
4 Analysis of High-Dimensional Structure-Activity Screening Datasets Using the Optimal Bit String Tree.....	34
4.1 Introduction	36
4.2 Recursive Partitioning	39
4.3 OBSTree	42

4.3.1	Splitting Criterion	46
4.3.2	Initial Conditions	47
4.3.3	Optimal Chromosome Selection	48
4.3.4	Simulated Annealing Portion	49
4.3.5	Trimming Procedure	50
4.3.6	Stopping Criteria	50
4.4	Simulation Studies	52
4.4.1	Study 1: Simulated Descriptors and Simulated Responses . .	52
4.4.2	Study 2: McMaster and Xue Compounds with Simulated Re- sponses	57
4.5	MAO Data Set Application	61
4.5.1	Predictive Tree Models and Cross Validation	61
4.5.2	Chemistry Interpretation	63
4.5.3	Stability of OBSTree	66
4.6	Discussion AND Summary	68
4.7	ACKNOWLEDGEMENTS	70

II Statistical Analysis of Compound Mixtures Using Non-linear Models 76

5	Concepts of Additivity in Chemical Mixture Studies	77
5.1	Introduction	77
5.2	Classifications of Mixture Action Mechanism	78
5.3	Definitions of Additivity	83
5.3.1	Loewe Additivity and Bliss Independence	83
5.3.2	Combination Index	86
5.3.3	Tallarida's Additivity	88
5.3.4	Interaction Index	92
5.4	How to Develop a Good Concept of Additivity?	94
5.5	Summary	101

6 Statistical Nonlinear Models and Hypothesis Tests in Chemical Mixture Studies 102

6.1	Introduction	102
6.2	Review of Nonlinear Models for Mixtures	104
6.2.1	Early Works on Modeling Interactions within Mixtures	104
6.2.2	Recent Models	107
6.3	A New Nonlinear Model for Binary Mixtures	112
6.3.1	The Model	112
6.3.2	Testing for Interaction	118
6.3.3	How to Classify Individual Combinations?	125

6.4	A Nonlinear Model for Mixtures of M Components and Hypotheses Tests	133
6.5	Real Data Sets with Quantal Response	136
6.5.1	Amethopterin and 6-mercaptopurine Data Set	136
6.5.2	Malathion and Parathion Data Set	141
6.5.3	Tramadol and Acetaminophen Data Set	144
6.5.4	Marathion, Parathion and Piperonyl Butoxide Data Set	155
6.6	Simulation Study	161
6.7	Design of Experiments	178
6.8	Discussion and Summary	182
Bibliography		184
Appendices		194
A Test Statistics in Studies with Quantal Responses		195
A.1	Likelihood Function	195
A.2	Hypothesis Test of $H_0 : s_1 = 0$ vs. $H_1 : s_1 > 0$	196
A.3	Hypothesis Tests of Individual Combinations	198
A.3.1	Likelihood function	198
A.3.2	Reparameterization	199
A.3.3	Wald Statistics with Observations on Sample Space Boundary	203
B SAS Programs		208
B.1	$H_0^i : s_i = 0$ vs. $H_1^i : s_i > 0$ in a Binary Study	208
B.1.1	Wald Test Statistic	208
B.1.2	Score Test Statistic	214
B.1.3	Likelihood Ratio Test	221
B.2	$H_{0,i} : \delta_i = 0$ vs. $H_{1,i} : \delta_i > 0$ in Tramadol and Acetaminophen Data Set	222
B.3	$H_0 : s_i = 0$ vs. $H_1 : s_i > 0$ in a Tertiary Mixture Study	236
B.3.1	Wald Test Statistic	236
B.3.2	Score Test Statistic	250
B.3.3	Likelihood Ratio Test Statistic	263
B.4	$H_{0,i} : \delta_i = 0$ vs. $H_{1,i} : \delta_m > 0$ in a Tertiary Mixture Study	266
B.4.1	Wald Test Statistic	266
B.4.2	Score Test Statistic	275
B.4.3	Likelihood Ratio Test	283
C Real Data Sets		289
C.1	Amethopterin and 6-mercaptopurine Data Set	289
C.2	Malathion and Parathion Data Set	291
C.3	Tramadol and Acetaminophen Data Set	292
C.4	Marathion, Parathion and Piperonyl Butoxide Data Set	294

List of Tables

2.1	Concepts of Additivity for A Binary Mixture	13
4.1	Four Biological Mechanisms Specified in Terms of the Descriptor Sets and the 1-0 Chromosomes.	52
4.2	Confusion Matrix Comparison for Study 1 (P=Prediction, T=True, Acc.=Accuracy, O.A.=Overall Accuracy).	58
4.3	Confusion Matrix Comparison for Study 2 (P=Prediction, T=True, Acc.=Accuracy, O.A.=Overall Accuracy)	60
4.4	Confusion Matrix Comparison for MAO Data Set (P=Prediction; T=True; Acc.=Accuracy; O.A.=Overall Accuracy.)	62
4.5	Stability of CV trees	67
5.1	Two-way Classification of Joint Action	79
6.1	Some (6.3) Binary Interaction Models	114
6.2	Different Conclusions of Interaction from (6.7)	116
6.3	Test results of amethopterin and 6-mercaptopurine data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.	139
6.4	Test results of marathion and parathion data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.	143
6.5	Test results of tramadol and acetaminophen data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.	147
6.6	Two-Sided test results of individual combinations of tramadol and acetaminophen. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is 72.	148
6.7	One-Sided test results of individual combinations of tramadol and acetaminophen. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is 144.	149

6.8	Test results of marathion, parathion and piperonyl butoxide Data. The total number of tests adjusted for multiplicity is 6.	158
6.9	Two-Sided test results of tertiary combinations on sample space boundary. The total number of tests adjusted for multiplicity is 24.	160
6.10	Estimates of Interactive Model for Tramadol and Acetaminophen Data Set	163
6.11	Estimated Type I Error with $N_{MC} = 1000$	166
6.12	Estimated MC power for one-sided tests of the most synergistic/antagnistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.123$ and $s_2 = -0.53$. The total number of tests adjusted for multiplicity is 30.	174
6.13	Estimated MC power for one-sided tests of the most synergistic/antagnistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.2$ and $s_2 = -1.4$. The total number of tests adjusted for multiplicity is 30.	175
6.14	Estimated MC power for one-sided tests of the most synergistic/antagnistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.2$ and $s_2 = -2$. The total number of tests adjusted for multiplicity is 30.	176

List of Figures

4.1	Flowchart for OBSTree	45
4.2	(a) The standard RP tree of Study 1 with minimum node size 5. Values shown in the terminal nodes are response values for the majority class of the node. Active terminal nodes are indicated with thick borders. The distributions of compounds for the four active terminal nodes are 0/20/0/0, 5/0/0/5, 0/0/0/7 and 0/0/20/0, respectively, from left to the right, where the numbers of compounds of four different potency levels (i.e. 0, 1, 2 and 3) are separated by “/”. (b) An OBSTree tree of Study 2; $T_0 = 12$, $T_{min} = 0.001$, $\alpha = 0.99$, $K = 10$, $t_p = 1$, $L_{max} = 3$ and minimum node size is 5. Splitting variables are given for each split with the set of descriptors and the corresponding chromosome separated by “/”. Distributions of compounds for the four active terminal nodes are 0/0/20/0, 0/20/0/0, 0/0/0/20 and 0/0/0/20, respectively, from the highest depth to the lowest depth.	55
4.3	Tree model comparison for MAO. Tree from OBSTree with penalty entropy splitting criterion, tree from RP/SA with absolute difference splitting criterion and the standard RP tree with deviance splitting criterion. Active terminal nodes are indicated with thick borders. Splitting variables are given for all splits. The distributions of compounds are shown in the terminal nodes, where the numbers of compounds of four different potency levels (i.e. 0, 1, 2 and 3) are separated by “/”.	63
4.4	Sample compounds in MAO data set. Compound 1, which is called Pargyline, is well known as an MAO inhibitor with potency three. Compound 2 has potency value zero; Descriptor 183, identified by OBSTree as needing to be absent for activity, is represented by the dashed arrows. Compounds 3 and 4 are both junk compounds, even though they appear in the first active terminal node of the standard RP tree. Compound 4 also appears in the second active terminal node of the RP/SA tree.	64

5.1	Isobologram for a combination of Drug 1 and Drug 2 with the dose-additive isobole corresponding to response Y	84
6.1	Dose-response curves of drug 1 with dose of drug 2=0, 2, 6, from two different models.	118
6.2	The scatter plot of the dose of 6-mercaptopurine versus the dose of Amethopterin.	137
6.3	Isobologram of amethopterin and 6-mercaptopurine.	140
6.4	The scatter plot of the dose of malathion versus the dose of parathion.	141
6.5	Isobologram of marathion and parathion.	145
6.6	The interaction status of each of 72 combinations in TRAM and APAP data set. “Test=No Test” refers to the method of using sign of $\mu_i - \mu_i^A$. “Test=Wald” refers to the Wald one-sided tests under B-H procedure. “Test=Score” refers to the score one-sided tests under B-H procedure. “Test=LRT” refers to the likelihood ratio tests under B-H procedure. The doses are plotted in a log scale.	151
6.7	Isobologram of Tramadol and Acetaminophen.	153
6.8	The Q-Q plot of the ordered Pearson residuals vs. the standard normal quantiles	154
6.9	The 3-D plot of dose point included in matathion, parathion and piperonyl butoxide data set.	156
6.10	The interaction status of each of 24 combinations in marathion, parathion and piperonyl butoxide data set. “Test=No Test” refers to the method of using sign of $\mu_i - \mu_i^A$. “Test=Wald” refers to the Wald one-sided tests under B-H procedure. “Test=Score” refers to the score one-sided tests under B-H procedure. “Test=LRT” refers to the likelihood ratio tests under B-H procedure. Dose1 refers to the dose of marathion; dose2 refers to the dose of parathion; dose3 refers to the dose of piperonyl butoxide. The unit of doses is uM . The doses are plotted in a log scale.	162
6.11	The Q-Q plot of the ordered Pearson residuals vs. the standard normal quantiles	163
6.12	The scatter plot of the dose of APAP versus the dose of TRAM.	164
6.13	The plot of MC estimated power versus s_2 with $N_{MC} = 1000$, $s_1 = 0.1$ and $w_k = 60$	167
6.14	The plot of MC estimated power versus s_1 with $N_{MC} = 1000$, $s_2 = -0.3$ and $w_k = 60$	168
6.15	The plot of MC estimated power versus Δ with $N_{MC} = 1000$, and $w_k = 60$	169
6.16	The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = 0.1$ and $s_2 = 0.8$	171

6.17	The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = -0.06$ and $s_2 = -0.3$	172
6.18	The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = 0.135$ and $s_2 = -0.65$	173

Chapter 1

Introduction

With the development of high throughput screening (HTS) techniques, large structural/biological activity data sets are now the norm rather than the exception (Sitampalam, Kahl, and Janzen 1997; Silverman, Campbell, and Broach 1998). Consequently, effective methods capable of managing large amounts of data and converting it into utilizable information are in great demand. It is an accepted tenet of medicinal chemistry that molecular structure is highly related to biological activity (McFarland and Gans 1986); two compounds with similar chemical structure are very likely to have similar biological potency with respect to one or more assays. Therefore, finding quantitative structure-activity relationships (QSARs) plays a critical role in analyzing chemical data sets.

The challenges existing in QSARs modeling and compound classification include ability to deal with multiple active mechanisms, i.e., molecular diversity, ability to handle a huge number of chemical descriptors with a large proportion of irrelevant descriptors, identification of high order nonlinear interactions among chemical de-

scriptors, model interpretations, computational efficiency and prediction accuracy.

Tree-based data-mining classification methods (Hawkins and Kass 1982, pp. 269-302; Breiman, Friedman, Olshen, and Stone 1984) are broadly utilized in QSARs exploration and potency prediction studies. The fundamental idea is to divide a diverse, heterogeneous group of objects into several homogeneous sub-classes according to specific structural descriptor rules, and each terminal class consists of compounds sharing some common chemical structures. Recursive Partitioning (RP) is a particular class of tree-based methods where partitioning is performed recursively to grow a tree. This greedy iterative procedure is simple, yet powerful, and is able to identify complex nonlinear relationships and strong interactions existing in large SAR data sets (Young and Hawkins 1995; Chen, Rusinko, Tropsha, and Young 1999; Rusinko, Farmen, Lambert, Brown, and Young 1999). In Chapter 2 we give a brief summary of the remaining chapters. Chapter 3 reviews the one version of standard RP and a series of its extensions. It also describes the motivation of developing our novel algorithm, OBSTree. Chapter 4 considers the details of OBSTree algorithm. The simulation and real application results are presented to illustrate the main features of the new algorithm.

One of the pioneering works that implement standard RP algorithm in drug discovery is called Statistical Classification of Activities of Molecules (SCAM), which was developed by Rusinko et al (1999). Chapter 3 starts with the introduction of SCAM. SCAM uses the binary descriptors to grow a two-way RP tree. In other words, one chemical descriptor is chosen to form the splitting variable at each split such that the original group of compounds can be easily divided into two subclasses

by binary property of the descriptor. The main advantage of the SCAM lies in that it can identify multiple active mechanisms to some extent and active mechanisms can be easily interpreted as a combination of several splitting descriptors that connect the terminal node and the root node in a RP tree.

Several extensions aiming to overcome the limitations of SCAM are also discussed in Chapter 3. Random Forest, proposed by Brien (2001), generated a large number of RP trees based on the bootstrap samples and randomly selected subsets of descriptors. The prediction was made by aggregating all the bootstrap RP trees. This ensemble method improves the prediction accuracy since it reduces the variance of the single tree. Binary Formal Inference-Based Recursive Modeling (BFIRM), proposed by Cho, Shen and Hermsmeier (1999), employed multiple-descriptor strategy to perform splitting at each node and showed advantages in analyzing QSAR over standard RP since multiple descriptors, as a group, are more biologically meaningful and more effective in defining a QSAR rule. The multiple descriptors are selected with a procedure ranking the individual descriptor first in BFIRM, whereas another algorithm, Recursive Partitioning and Simulated Annealing (RP/SA) (Blower, Fligner, Verducci, and Bjoraker 2002), selects optimal combination of descriptors without considering the individual importance of each descriptor. The main argument for developing RP/SA is that the optimal descriptor set does not necessarily consist of individual important descriptors. Some descriptors may be important only if they are considered in a combination. Both BFIRM and RP/SA can be viewed as an effort to improve the bias of the single RP tree. They perform binary splits with the positive daughter nodes containing the compounds in which all splitting multiple descriptors are simul-

taneously present. SA is implemented by RP/SA as a stochastic optimization tool to find global optimum descriptor set.

The motivation for proposing Optimal Bit String Tree (OBSTree) is to consider a situation where the absence of some blocking chemical descriptors is as critical as the presence of some active descriptors to explain potent compounds. Consequently, a more biologically meaningful splitting variable should be a combination of multiple descriptors plus the presence or absence of each descriptor. In OBSTree, the presence or absence of splitting descriptors is coded as a “chromosome”, a string of 1-0 bits. The splitting variable selection includes the optimization of the descriptor set and its chromosome.

Chapter 4 introduces the new optimization scheme of OBSTree with the assumption that potency is an ordinal variable. A new splitting criterion, penalty entropy, extends the general entropy criterion that groups categorical responses by grouping the active compounds only. The criterion penalizes a pure node with a high proportion of inactive compounds by treating each inactive compound as a different category. With penalty entropy the optimal chromosome for a fixed descriptor set is found by a thorough search. The whole splitting variable is optimized with the SA incorporated with a weighted sampling scheme that searches the descriptors from “singly-important” group and “general descriptor” group with different weights. This speeds up the searching and substantially improves the optimality of the final result when the number of descriptors is large. A post-trimming procedure is performed after the optimal splitting variable is identified. The main purpose is to remove some irrelevant descriptors from the splitting variable without decreasing the optimality.

The procedure allows those users without much prior knowledge of active mechanisms in the data set to initiate the algorithm easily. The application results of OBSTree show that this novel algorithm successfully identifies the presence or absence of key descriptors and yields good interpretation from the model.

We also investigate the interactions of chemical mixtures, which is of great interest in practice. For example, the synergism in efficacy of a drug combination therapy and the antagonism in toxicity of the combination therapy may have potential therapeutic advantages in clinical trials since it implies that a better therapeutic effect with a reduced adverse effect can be achieved with the same or even smaller amounts of the drugs.

The work includes two related statistical issues. First is about developing a concept of additivity to determine and characterize interactions of the chemical mixtures. It is well known that any conclusion of the interaction depends on the concept of additivity. There, however, exist different additivity definitions leading to conflicting results. As far as we know, a thorough review of different concepts of additivity is currently unavailable and is of great interest. Chapter 5 reviews several widely-used concepts of additivity. Our work aims to investigate the necessary conditions or assumptions for different concepts of additivity and to clarify the mathematical derivations of those concepts. Some criteria for evaluating additivity definitions are discussed and a particular concept of additivity, interaction index, is generalized to handle some complicated studies including “site-site” interaction studies (Raffa et. al. 2000; Tallarida 2000, 2001).

The second issue is about using statistical models to describe the dose-response relationships of the chemical mixtures and developing statistical hypothesis tests to detect the interactions. In Chapter 6 we propose a nonlinear model for a binary mixture and generalize it to a mixtures of M components. The model can naturally accommodate additivity, synergism, antagonism and dose-dependent interaction without requiring any prior knowledge of biological activity mechanisms. Three different types of test statistics combined with different multiple testing adjustment methods are proposed to test two parameters in the model such that a conclusion of the overall interaction can be drawn. In addition, we construct three types of two-sided test statistics and three types of one-sided test statistics to test interaction status of individual combinations. These tests can be especially useful to determine which combinations are interactive, or more specifically, are synergistic or antagonistic. The model and the testing procedures were applied to four real data sets with quantal responses. SAS programs were developed to calculate different test statistics. The testing results based on our model are highly consistent with the experimental results. Simulation studies were conducted to examine the power of different testing procedures.

Chapter 2

Summary of Research and Results

2.1 Summary of OBSTree

2.1.1 Motivation and Main Features

Tree-based data-mining classification methods are broadly utilized in QSARs exploration and potency prediction studies. The fundamental idea is to divide a diverse, heterogeneous group of objects into several homogeneous sub-classes according to specific structural descriptor rules, and each terminal class consists of compounds sharing some common chemical structures.

Recursive Partitioning (RP) is a particular class of tree-based methods where partitioning is performed recursively to grow a tree. It has been shown (Cho, Shen, and Hermsmeier 2000; Blower et al. 2002) that using multiple descriptors to define splitting variable in RP trees is advantageous over using a single descriptor. Current multiple-descriptor RP tree methods, however, have the following problems. First and

foremost, the importance of the absence of descriptors is totally ignored in forming the splitting variable. There are situations, however, where absence of some descriptors also plays an important role in explaining activity mechanisms. Such cases include blocking sub-structures of a compound that can interact with the active sub-structure of the compound to inhibit its potency. Second, the appropriate number of descriptors involved in explaining active mechanisms is always highly questionable since this is either arbitrarily pre-specified by researchers or calculated by an incorrect inclusion sequence of descriptors (see Sections 3.5 and 4.3 for further details). Third, it is challenging to win both on the optimality of the splitting variable and computational time. A more efficient searching scheme is of great demand if the searching space of splitting variable is formed by considering multiple descriptors and the presence or absence of each descriptor.

OBSTree is developed to try to solve the three problems. One main feature of this method is introduction of the new concept of chromosome for identifying key splitting chemical sub-structures. Each chromosome is a binary string of 1-0 bits with one indicating the presence of a molecular sub-structure as represented by a descriptor and zero indicating the absence of that molecular sub-structure. A set of K descriptors combined with a specific chromosome forms the splitting variable of OBSTree, which results in a two-way split. Only those compounds that contain the exact splitting rule where all K selected descriptors take values exactly according to the selected chromosome are included in the positive daughter node. The remaining compounds go to the negative daughter node. A specially designed splitting criterion was developed to account for the fact that we are not necessarily attempting pure separation

for both nodes, only for the positive daughter node. A post-trimming procedure is incorporated in the algorithm such that researchers can be allowed to initially set a comparably large number of descriptors for splitting variable optimization. The procedure will remove the noisy information in a data-driven way. A weighted sampling scheme is combined with SA to serve as a new searching algorithm. The application results show that this searching algorithm can substantially improve the optimality of the results within a practical computing time. We omit a detailed description of OBSTree and refer to Section 4.3 for further details. The flowchart of OBSTree algorithm is provided in Figure 4.1 (p. 45).

2.1.2 Application Results

The performance of OBSTree was examined in two simulation studies and one real data set.

Simulation Study 1

The first simulated data set contains 1000 compounds, 500 binary descriptors, and an activity rate of 8% corresponding to a four-level categorical response. The 80 active compounds were created to arise from four different “activity mechanisms”. Each of “activity mechanism” consists of a combination of five descriptors and the corresponding 1-0 chromosome.

OBSTree is able to identify a complete class of active compounds on every split because it uses both a descriptor set and chromosome to capture an exact biological

activity mechanism. Moreover, it is clear that the trimming procedure is effective. In all cases it was able to trim the excess noise variable to reduce K from 10 (the default starting value) to the required five for all mechanisms.

Another multiple-descriptor tree, called RP/SA (Blower et al. 2002), fails to identify any active class since it does not consider any absent descriptor in splitting variables. The standard RP tree finds only seven of the 20 compounds in Mechanism I, five of the 20 compounds in Mechanism II. It also misclassifies five junk compounds as falling in Mechanism II. The main reason for the worse result of the standard RP tree is that using a single descriptor to make splits for identifying one class of active compounds can inadvertently divide other active classes.

The prediction performance of OBSTree, standard RP, and RP/SA was compared using 5-fold cross-validation (CV). OBSTree gives the best overall accuracy percent.

Simulation Study 2

The second data set is a mixed one consisting of 8076 real compounds with only the response values being simulated. Four known classes of compounds were set as our active compounds and there were 76 active compounds in total. 1260 atom pair descriptors were calculated using PowerMV (Liu, Feng, and Young 2004).

5-fold CV analysis again was conducted to compare three different methods: OBSTree, RP/SA, and standard RP. OBSTree gives the best overall performance with the highest overall accuracy percent.

MAO Data Set Application

A set of 1646 monoamine oxidase (MAO) inhibitors was provided by Abbott Laboratories (Brown and Martin 1996). 1380 atom pair descriptors were used to generate tree models.

5-fold CV analysis comparison among three methods (i.e., OBSTree, RP/SA, and standard RP) shows similar overall accuracy percent. For this data set, the biggest benefit offered by OBSTree is interpretation of the molecular sub-structures identified as being important. All three methods tried to identify a group of pargyline-like compounds with a well-known biological activity mechanism. OBSTree captured all critical active sub-structures and identified an absent sub-structure that can potentially block the activities of compounds through a neighborhood effect. RP, however, only captured part of the required chemical sub-structures. Also, the neighborhood effect was not detected by the standard RP tree. RP/SA identified some irrelevant sub-structures due to an inappropriate setting of the number of descriptors in the splitting set (K).

2.2 Summary of Concepts of Additivity in Chemical Mixture Studies

Determining synergism and antagonism is of great interest in chemical mixture studies. The definition of synergism and antagonism depends on the concept of additivity. There are a number of ways to define additivity, resulting in different concepts

of additivity. Unfortunately, there is no consensus answer to the question: “ Which additivity concept is the best?” As far as we know, a thorough review of different concepts of additivity is unavailable and of great demand.

We review several widely-used concepts of additivity in Section 5.3, which include Loewe additivity (dose/concentration addition), Bliss independence (response addition), combination index, Tallarida’s additivity and interaction index. The first four concepts are related to the joint action mechanisms of the mixtures, whereas interaction index is developed free of any joint action mechanisms. The mathematical forms of these additivity concepts are summarized in Table 2.1 using the notation described in Chapter 5. Detailed information can be found in Section 5.3.

Table 2.1: Concepts of Additivity for A Binary Mixture

Concepts of Additivity	Joint Action Mechanism	Mathematical Expression
Loewe additivity (Dose/Concentration addition)	Similar action	$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1$
Bliss independence (Response addition)	Dissimilar action	$Y = 1 - (1 - f_1(x_1))(1 - f_2(x_2))$
Combination index	Mutually exclusive ligands obeying mass-action law	$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1$
	Mutually non-exclusive ligands obeying mass-action law	$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} + \frac{x_1x_2}{ED_1(Y)ED_2(Y)} = 1$
Tallarida's additivity	Independent action	$x_1 + \beta x_2 = ED_1(Y)$ ${}^a \beta = \frac{ED_1(Y)}{ED_2(Y)}$
		$x_1 + \beta x_2 = ED_1(Y)$ ${}^c \beta = \frac{ED_1(Y)}{ED_2(Y)}$ or $\beta = \frac{ED_1(f_2(x_2))}{x_2}$
Interaction index	Independent on mechanisms	$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1$

^a β =Relative potency of Component 1 to Component 2

^b β is a constant over different response levels

^c β is a variable over different response levels. See Section 5.3.3 for further details.

The criteria for evaluating a concept of additivity are discussed in Section 5.4. We support the ideas of Berenbaum (1989) on the interaction index that a good concept of additivity should be defined to be generally applicable for most cases and free of joint action mechanisms. We also recognize that it is critical to use a chemically additive system (i.e., sham mixture in interaction index) to define the quantitative properties in a concept of additivity. Using an example of “site-site interaction” study, we illustrated how the chemically additive system of interaction index can be generalized to a more complicated study. Given a mixture of M chemicals with k categorical explanatory variables and dose considered in modeling the response, a general form of the concept of additivity is

$$\sum_{i=1}^M \left(\sum_{z_k=1}^{Z_k} \cdots \sum_{z_1=1}^{Z_1} \frac{x_{iz_1 \cdots z_k}}{ED_i(Y, z_1 = z_1, \cdots, z_k = z_k)} \right) = 1,$$

where z_1, \dots, z_k are k categorical variables, $x_{iz_1 \cdots z_k}$ is the dose of chemical i in the mixture system with a fixed set of $z_1 = z_1, \dots, z_k = z_k$, and $ED_i(Y, z_1 = z_1, \dots, z_k = z_k)$ is the dose of chemical i alone with a fixed set of $z_1 = z_1, \dots, z_k = z_k$ that can yield response Y .

2.3 Summary of Statistical Nonlinear Models and Hypothesis Tests in Chemical Mixture Studies

2.3.1 Motivations and New Nonlinear Models

A thorough review of different dose-response models for chemical mixtures is given in Sections 6.2.1 and 6.2.2. Please refer to Table 6.1 for the forms of several binary

interaction models.

There exist two popular types of modeling strategies, each of which has its own limitations. First, a simple statistical model can be used to describe interaction without requiring prior knowledge of activity mechanisms or details of how interaction occurs. An example is the generalized-linear-model-type model (Brunden et al. 1983; Meadows, Gennings, Carter, and Bae 2002). Its simplicity is also its downfall, however, in that the model is not able to capture complicated dependencies between dose level and type or strength of interaction. In other words, the model cannot accommodate the situation where interactions may occur only at a subset of combination points or be restricted to a specific dose combination area (Tallarida, Porreca, and Cowan 1989; Dawson, Carter, and Gennings 1995; Hamm, Carter, and Gennings 2005). This situation is subsequently referred to as dose-dependent interaction. Different from the first modeling strategy, some complicated models were proposed to flexibly accommodate complex interaction situations including dose-dependent interaction, and to interpret data in a more chemically meaningful way. The Rider and LeBlanc (2005) interaction model and the Hamm et al. (2005) threshold interaction model are two examples reviewed in Section 6.2.2. These models, however, assume prior knowledge that is typically not available. For instance, the Rider and LeBlanc interaction model needs to classify chemicals in the mixture based on their biological activity mechanisms and the Hamm et al. threshold interaction model requires pre-specification of interaction boundary shape. In addition, the models are mathematically complicated, thus, require large data collection for estimation. This also makes it impractical to extend the models for a mixture of more than three components.

We propose a new nonlinear model that combines the advantages of two modeling strategies. The new model for a binary mixture is

$$E[Y] = \mu = q(\beta_1 + \beta_2 x_1 (1 + x_2)^{s_1} + \beta_3 x_2 (1 + x_1)^{s_2}),$$

where Y is the response variable from an exponential family with a density function $f(y; \mu)$, $q^{-1}(\cdot)$ is a link function, x_1 and x_2 are doses of two chemicals, and s_1 and s_2 are two parameters to be estimated. Under the assumption that $q(\cdot)$ is a monotonically increasing function and every chemical has a monotonically increasing dose-response relationship, additivity is implied by $s_1 = s_2 = 0$, overall antagonism by $s_1 < 0$ and $s_2 < 0$, overall synergism by $s_1 > 0$ and $s_2 > 0$, and dose-dependent interaction by $s_1 s_2 < 0$. The model can be easily extended to describe a mixture of M components. A general form of our nonlinear model for a mixture of M components is given as

$$E[y] = \mu = q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_m h_m\right),$$

where $h_m = (1 + \sum_{j=1, j \neq m}^M x_j)^{s_j}$, $m = 1, \dots, M$. Again, this model can accommodate different types of interaction as well as additivity. Neither prior knowledge of biological activity mechanisms nor that of shape of interaction boundaries is required to develop the model.

2.3.2 Hypothesis Tests

Testing for Interaction

We only summarize the testing procedures for a binary mixture in this section. The methodology principles can be easily extended to a mixture of M components.

Please refer to Section 6.4 for more details about the testing procedures for a mixture of M components. In order to determine the interaction situation over the entire dose range for a binary mixture, we wish to conduct the following four one-sided tests of

$$(1) \quad H_0^1 : s_1 = 0 \text{ vs. } H_1^1 : s_1 > 0$$

$$(2) \quad H_0^2 : s_1 = 0 \text{ vs. } H_1^2 : s_1 < 0$$

$$(3) \quad H_0^3 : s_2 = 0 \text{ vs. } H_1^3 : s_2 > 0$$

$$(4) \quad H_0^4 : s_2 = 0 \text{ vs. } H_1^4 : s_2 < 0.$$

Multiple testing adjustment methods can be applied to control the family-wise error rate in a strong sense. We implemented a simple Bonferroni adjustment (Miller 1981, pp. 67-70) and a modified Bonferroni test procedure proposed by Hochberg (1988) separately in our real applications. For each test, three types of one-sided test statistics were considered: Wald, score and likelihood ratio.

We illustrate all steps using H_0^1 versus H_1^1 , where $\boldsymbol{\theta} = (s_1, s_2, \beta_1, \beta_2, \beta_3)^T$ is partitioned as $\boldsymbol{\theta}^T = (\theta_1 = s_1, \boldsymbol{\theta}_2^T = (s_2, \beta_1, \beta_2, \beta_3)^T)$.

Suppose data is available for K mixture dose points as responses y_1, \dots, y_K and dose combinations (x_{1k}, x_{2k}) , $k = 1, \dots, K$. Assuming the K points are independently distributed, then the likelihood is

$$L(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_K) = \prod_{k=1}^K f(y_k; \mu_k),$$

where

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}).$$

The Wald test statistic is

$$A_1^W = (\widehat{s}_1 - 0) \left\{ K^{-1} [I_K(\widehat{\boldsymbol{\theta}})^{-1}]_{11} \right\}^{-1} (\widehat{s}_1 - 0),$$

where \widehat{s}_1 and $\widehat{\boldsymbol{\theta}}$ are maximum likelihood estimators under $H_0^1 \cup H_1^1$, and $I_K(\widehat{\boldsymbol{\theta}})$ is the average expected Fisher information matrix evaluated at $\widehat{\boldsymbol{\theta}}$ and $[I_K(\widehat{\boldsymbol{\theta}})^{-1}]_{11}$ is the upper (1,1) element of the inverse of $I_K(\widehat{\boldsymbol{\theta}})$.

The score test statistic is

$$A_1^S = S_{s_1}(\tilde{\boldsymbol{\theta}}) \left\{ K(\tilde{I}_{11} - \tilde{I}_{12} \tilde{I}_{22}^{-1} \tilde{I}_{21}) \right\}^{-1} S_{s_1}(\tilde{\boldsymbol{\theta}}) - \inf \left\{ (S_{s_1}(\tilde{\boldsymbol{\theta}}) - b) \left\{ K(\tilde{I}_{11} - \tilde{I}_{12} \tilde{I}_{22}^{-1} \tilde{I}_{21}) \right\}^{-1} (S_{s_1}(\tilde{\boldsymbol{\theta}}) - b) : b \in H_0^1 \cup H_1^1, \text{i.e., } b \geq 0 \right\},$$

where $\tilde{\boldsymbol{\theta}}$ is the maximum likelihood estimator under the null hypothesis, $S_{s_1}(\tilde{\boldsymbol{\theta}})$ is the score function of s_1 evaluated at $\tilde{\boldsymbol{\theta}}$, and \tilde{I}_{11} , \tilde{I}_{12} , \tilde{I}_{22} , and \tilde{I}_{21} are partitioned components of the information matrix evaluated at $\tilde{\boldsymbol{\theta}}$.

The likelihood ratio test can be constructed as

$$A_1^L = -2 \log \left\{ \frac{\sup_{\boldsymbol{\theta} \in H_0^1} L(\boldsymbol{\theta}; y_1, \dots, y_K)}{\sup_{\boldsymbol{\theta} \in (H_0^1 \cup H_1^1)} L(\boldsymbol{\theta}; y_1, \dots, y_K)} \right\} = -2 \log L(\tilde{\boldsymbol{\theta}}) + 2 \log L(\widehat{\boldsymbol{\theta}}).$$

All three test statistics, A_1^W , A_1^S , A_1^L have the same asymptotic distribution under the null hypothesis, namely a chi-bar-squared distribution. A chi-bar-squared distribution is a mixture of chi-squared distributions. Please refer to Section 6.3.2 for further details.

Testing Individual Combinations

Testing procedures are proposed to individually identify all potentially interactive dose combinations, or, even better, to tell if the interaction is synergistic or

antagonistic.

Suppose K mixture dose points $(x_{11}, x_{21}), \dots, (x_{1K}, x_{2K})$ with corresponding responses y_1, \dots, y_K are observed, and we are interested in identifying interaction status for $i = 1, \dots, N (N \leq K)$ of these dose combinations. The basic question is whether an additive model adequately predicts the expected response of combination i , i.e., is $\mu_i = E(y_i)$ well represented by μ_i^A , where $\mu_i^A = q(\beta_1 + \beta_2 x_{1i} + \beta_3 x_{2i})$ is obtained from an additive model. Additionally, we may want to detect synergism as $\mu_i > \mu_i^A$ or antagonism as $\mu_i < \mu_i^A$. To avoid overfitting when estimating μ_i^A , our parametric interaction model is implemented to describe all the data except combination i . Moreover, rather than treating μ_i as known without error after substituting observed value y_i , we recognize the variability of y_i and assume that y_i has a density function $f(y_i; \mu_i)$ with $\mu_i = E[y_i]$. Please note that we do not assume our interaction model on μ_i . Please refer to Section 6.3.3 for further details.

The issue of multiple testing is again encountered because a series of tests will be pursued for each of N combination dose points. Bonferroni and Hochberg adjustment procedures that both strongly control family-wise error rate could be used if N is small. For large N , Benjamini and Hochberg (1995) proposed another Bonferroni-type procedure (subsequently referred to as the B-H procedure) that aimed to control the false discovery rate rather than family-wise error rate and consequently resulted in increased power. We implemented the Hochberg adjustment procedure and the B-H procedure in our real applications.

Suppose we want to test

$$H_{0,i,two} : \mu_i - \mu_i^A = 0 \quad \text{vs.} \quad H_{1,i,two} : \mu_i - \mu_i^A \neq 0,$$

for $i = 1, \dots, N$. Let $\boldsymbol{\lambda} = (\mu_i, \beta_1, \beta_2, \beta_3, s_1, s_2)^T$. Under the assumption that all dose combinations are independent the likelihood function for $\boldsymbol{\lambda}$ is

$$L_i(\boldsymbol{\lambda}; y_1, \dots, y_K) = f(y_i; \mu_i) \prod_{k \neq i, k \in K} f(y_k; \mu_k),$$

where

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}), k \neq i.$$

Three traditional test statistics: Wald, score and likelihood ratio can be easily calculated for this two-sided test. Please refer to Section 6.3.3 for further details.

One-sided Wald, score and likelihood ratio test statistics can be developed for testing synergism and antagonism, i.e., testing

$$H_{0,i} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i} : \mu_i > \mu_i^A$$

and

$$H_{0,i'} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i'} : \mu_i < \mu_i^A.$$

The model needs to be reparameterized before we implement the one-sided score statistic. Define $\mu_i = q(\beta_1 + \beta_2 x_{1k} + \beta_3 x_{2k} + \delta_i)$, where $-\infty < \delta_i < \infty$. Consequently, testing $H_{0,i}$ vs. $H_{1,i}$ and $H_{0,i'}$ vs. $H_{1,i'}$ becomes testing

$$H_{0,i} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i} : \delta_i > 0$$

and

$$H_{0,i'} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i'} : \delta_i < 0 .$$

The likelihood function under the reparameterization can be written as

$$L_i(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_K) = f(y_i; \mu_i) \prod_{k \neq i, k \in K} f(y_k; \mu_k),$$

where

$$\mu_i = q(\beta_1 + \beta_2 x_{1k} + \beta_3 x_{2k} + \delta_i)$$

and

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2})k \neq i.$$

Based on the reparameterized likelihood function three one-sided test statistics can be developed in a similar way as described before. Please refer to Section 6.3.3 for further details.

2.3.3 Application Results

Real Data Sets

We applied our interaction model and testing procedures on four real data sets with quantal response. SAS programs (see Appendix B) were developed to compute different test statistics. After testing signs of s_i , $i = 1, \dots, M$, with three test statistics, we conclude that one of the data set has overall synergism, one overall antagonism, and two dose-dependent interaction. The results are consistent with the experimental findings and the analysis results from other models. For the two data sets where dose-dependent interaction was identified we conducted testing procedures for individual combinations. Different types of interactions could be determined for different combinations. Two goodness-of-fit statistics, the deviance (Agresti 2002, pp. 141-142) and the Pearson statistic (Agresti 2002, p. 220), were calculated to check

the model fit for each data set. It should be noted that the model provided a poor fit for two data sets. Therefore, the test results, especially those for individual dose combinations, need to be interpreted with great caution. Please refer to Section 6.5 for all real application results.

Simulation

In order to check the power of our proposed testing procedure, we conducted a simulation study for a binary mixture with quantal responses.

Using a fixed set of β_1 , β_2 , and β_3 as well as different combinations of (s_1, s_2) as the truth, we simulated cases with either an overall synergistic effect, an overall antagonistic effect or a dose-dependent interaction. A Monte Carlo (MC) method was implemented to estimate the power of each test statistic. As expected, the power increases when the magnitude of interaction increases. The power of three types of statistics are quite close to each other for the cases with overall antagonism or dose-dependent interaction in our simulation. The likelihood ratio test statistic has larger power than the score test statistic and smaller power than the Wald statistic for the overall synergism case. The type I error of each of the three statistics is generally less than 0.05 with the score statistic slightly more conservative than the other two.

In addition, the relationship between sample size and power was examined with three different fixed sets of s_1 and s_2 , corresponding to overall synergism, overall antagonism, and dose-dependent interaction. As expected, the power decreases as sample size decreases. For the overall simulated synergism, the Wald statistic is more robust to sample size than the likelihood ratio and score test statistics in our

simulations. For the overall antagonism and dose-dependent interaction, three types of statistics yielded quite similar power with the same sample size.

The power of testing individual combinations was also examined in simulation studies. In general, the power of all methods will decrease dramatically, when the number of hypotheses is relatively large. This is the cost of multiplicity control. The B-H procedure can substantially increase the power compared to the Hochberg adjustment since it controls the false discovery error rate. The power of the score test statistic in our example is slightly less than the other two statistics. This could be a general situation since score test statistic is generally smaller and more conservative in controlling type I error than the other two. Detailed results can be found in Section 6.6.

Part I

Statistical Analysis of Compounds Using OBSTree

Chapter 3

Recursive Partitioning and Its Extensions

3.1 Introduction

With the development of high throughput screening (HTS) techniques, large structural/biological activity data sets are now the norm rather than the exception (Sitampalam, Kahl, and Janzen 1997; Silverman, Campbell, and Broach 1998). How to effectively manage large amounts of data and accurately convert it into utilizable information is of great interest. It is an accepted tenet of medicinal chemistry that molecular structure is highly related to biological activity (McFarland and Gans 1986); two compounds with similar chemical structure are very likely to have similar biological potency with respect to one or more assays. Therefore, finding quantitative structure-activity relationships (QSARs) plays a critical role in analyzing chemical

data sets.

The task becomes more difficult as the size of the data set increases. The complexity of large data sets includes the fact that there are multiple active mechanisms and more than one active class of compounds and high order nonlinear interaction exists among descriptors and biological activity. Consequently, model interpretations, computational efficiency and prediction accuracy become challenging issues for developing a desired QSAR model.

Tree-based data-mining classification methods (Hawkins and Kass 1982, pp. 269-302; Breiman, Friedman, Olshen, and Stone 1984) are broadly utilized in QSARs exploration and potency prediction studies. The fundamental idea is to divide a diverse, heterogeneous group of objects into several homogeneous sub-classes according to specific structural descriptor rules, and each terminal class consists of compounds sharing some common chemical structures. Recursive Partitioning (RP) is a particular class of tree-based methods where partitioning is performed recursively to grow a tree. This greedy iterative procedure is simple, yet powerful, and is able to identify complex nonlinear relationships and strong interactions existing in large SAR data sets (Young and Hawkins 1995; Chen, Rusinko, Tropsha, and Young 1999; Rusinko, Farmen, Lambert, Brown, and Young 1999).

3.2 SCAM

SCAM, which was developed by Rusinko et al. (1999), uses binary descriptors to grow RP trees for the purpose of uncovering QSARs. They, along with Young

and Hawkins (1995), were among the first to use RP for QSAR identification. The full chemical structure of every compound in SCAM is described by a bit-string indicating the presence or absence of different molecular sub-structures. For each split, one descriptor is chosen as the splitting variable such that compounds with a 1-value for this descriptor form the positive daughter node and compounds with a 0-value for this descriptor form the negative daughter node. A splitting criterion is needed to evaluate the splitting performance of each descriptor candidate. SCAM uses a t-test for comparing means of the two daughter nodes as the splitting criterion and a multiple-testing adjusted p-value to determine terminal nodes.

A number of other splitting criteria have been developed and implemented. Some, such as deviance, entropy and the Gini Index, have been particularly useful applied to categorical response variables (see, e.g., Breiman et al. 1984; Hastie, Tibshirani, and Friedman 2001). The descriptor that optimizes the splitting criterion, i.e., the one that yields the most different daughter nodes, is selected to be the final splitting variable. The resulting daughter nodes then become parent nodes for other splits. This recursive partitioning continues until some stopping criterion, such as a pre-specified threshold of p-value in SCAM, is met. Each terminal node of the SCAM tree is regarded as a class of compounds that share similar chemical structures and biological activity. It is assumed that critical molecular sub-structures responsible for biological activity are identified as the combination of all the splitting descriptors that connect the terminal node and the root node.

3.3 Random Forest

Random Forest, as an ensemble method, was proposed by Breiman (2001). It was applied to drug discovery with main efforts to improve the prediction accuracy (Svetnik et al. 2003). The training procedure of Random Forest is fairly simple. It starts with drawing a bootstrap sample of compounds with replacement. A standard RP tree grows on the sample with a randomly selected subset of descriptors. The bootstrap sampling and tree development procedure will be repeated for a large number of times such that multiple trees (forest) are formed. Notice each sample tree is created using different compounds and descriptors. All trees are aggregated to generate final prediction result based on majority vote rule. The insights of the algorithm (Svetnik et al. 2003) tells us that Random Forest is primarily a variance reduction technique with little effect on the bias. This can be empirically explained as the basic component of the forest is still the standard RP tree and the algorithm uses ensembles of trees to decrease the variability. In other words, the algorithm may fail when error rate due to the bias is intrinsically large in a single tree model and dominates the whole error rate. In addition, the interpretation of Random Forest is generally hard.

3.4 BFIRM and RP/SA

One disadvantage of SCAM is using a single descriptor as the splitting variable since a single descriptor may not be able to adequately identify a class of compounds or fully explain a biological activity mechanism. In other words, an active class is

generally identified by a pathway containing more than one split in a standard RP tree. Those single splitting descriptors that appear in the early stage of a standard RP tree may be collectively important for one activity class, but these splits can result in adverse separation of other activity classes. Using a combination of descriptors to define each split can efficiently overcome this drawback, albeit at the cost of greater computational complexity. In addition, multiple descriptors, as a group, are more biologically meaningful and more effective in defining a QSAR rule; this is fully discussed in Chapter 4.

Binary Formal Inference-Based Recursive Modeling (BFIRM), which was proposed by Cho et al. (1999), extends RP in that a multiple-descriptor strategy is employed to perform splitting at each node. Selection of a set of binary descriptors in BFIRM starts with a ranking procedure that sorts the descriptors from the best to the worst based on an F-statistic, which is the ratio of the variances between the two daughter nodes and within the two daughter nodes with higher value being more optimal. The set of multiple descriptors is formed by greedily incorporating descriptors into the splitting set based on the ranked sequence. No additional descriptors will be included when the p-value of the F-statistic resulted from the current descriptor combination exceeds a pre-specified F-value threshold. The purpose of the procedure is to identify as many related descriptors as possible under the constraint that the final splitting set cannot yield a insignificant split. Cho et al. (2002) show that better QSAR analysis results are achieved with multiple descriptors due to the more informative splitting rule identified by a set of descriptors instead of a single descriptor.

Blower et al. (2002), however, argue that the best single descriptors found by

BFIRM do not necessarily guarantee that the combination is also the best in optimizing the splitting criterion. It is likely that some descriptors can only become important when they are considered in a combination. An individually unimportant descriptor, however, has a low probability of being selected in BFIRM. Thus, the search space should consist of all possible combinations of K descriptors, where K is a pre-specified number. The RP/SA technique of Blower et al. (2002) is one such multiple-descriptors search strategy. Blower et al. (2002) argue that their optimal splitting set may include several descriptors that are not individually important. A two-way split is formed with positive daughter node consisting of compounds simultaneously having all splitting descriptors and negative daughter node consisting of the remaining compounds. The optimal set of descriptors is identified using simulated annealing (SA). SA is a stochastic optimization method that moves between different local optima. It accepts all uphill steps and some downhill steps such that it is able to avoid becoming trapped at a local optimum and find a globally optimal result.

Both of BFIRM and RP/SA can be viewed as an effort to reduce the bias of the standard RP model. In addition, both splitting procedures of BFIRM and RP/SA imply that the positive daughter node is more homogeneous than the negative daughter node since the combination of descriptors are simultaneously present in all compounds in the positive node. Consequently, an ideal situation for BFIRM and RP/SA is that the positive node at each split creates a terminal node that represents a unique class of active compounds. Blower et al. (2002) show that a left-leaning RP/SA tree is always grown whenever RP/SA can effectively find the different classes of active compounds.

3.5 Motivations of OBSTree Development

Three issues motivate us to develop the novel algorithm OBSTree of which explicit details are described in Chapter 4.

First and foremost, the importance of the absence of descriptors cannot be ignored. Both RP/SA and BFIRM automatically assume that only the presence of multiple descriptors is critical in deriving a significant split. This assumption works well when the different biological mechanisms can be fully explained by the presence of different sets of descriptors. There are situations, however, where absence of some descriptors also plays an important role in explaining activity mechanisms. Such cases include blocking sub-structures of a compound that can interact with the active sub-structure of the compound to inhibit its potency. Both RP/SA and BFIRM have difficulty in identifying QSAR rules that include the absence of descriptors. It is essential for us to consider both the set of descriptors and the presence/absence of each descriptor within the set when forming the splitting variable.

Second, the appropriateness of number of descriptors involved in explaining active mechanisms is always highly questionable. RP/SA requires the number of descriptors for splitting to be pre-specified and to be fixed during the tree growth process. It is usually hard to define a sensible number with little knowledge on the QSARs information of the data set. This parameter, however, can significantly affect the results of RP/SA. No complete active mechanisms can be found if the pre-specified number is less than the real cases. An even worse situation is that no mechanisms can be found and computational time goes to infinity when the pre-specified number is much

larger than the true mechanisms and the requirement of compounds simultaneously possessing all splitting descriptors is difficult to satisfy. BFIRM has a data-driven process to define a variable number of descriptors. It, however, starts with including the individually best descriptor. Any following step in the procedure is conditioning on the previously selected descriptors. The first chosen descriptor, however, is questionable by itself since no fact shows that the optimal combination of descriptors should contain the most individually important descriptor. Therefore, the total number of descriptors may be incorrect due to the first wrongly selected descriptor. A data-driven approach to simultaneously defining the appropriate descriptor set and the number of descriptors is desired.

Third, it is challenging to win both on the optimality of the splitting variable and computational time. SA approach implemented by RP/SA, in theory, will converge to the global optimum, but in practice there is no such guarantee due to the limited computational time. The greedy method of BFIRM is also not an efficient tool to optimizing descriptor set. A more efficient searching scheme is of great demand if the searching space of splitting variable is formed by considering multiple descriptors and the presence or absence of each descriptor.

OBSTree is developed to try to solve the three problems. A concept of "chromosome" is proposed to quantitatively describe the presence or absence of multiple descriptors. The descriptor set plus the chromosome forms the splitting variable of OBSTree. A post-trimming procedure is incorporated in the algorithm such that the researchers can be allowed to firstly set a comparably large number of descriptors for splitting variable optimization. The procedure will remove the noisy information in

a data-driven way. A weighted sampling scheme is combined with SA to serve as a new searching algorithm. The application results show that this searching algorithm can substantially improve the optimality of the results within a practical computing time. The details of OBSTree are shown in Chapter 4.

Chapter 4

Analysis of High-Dimensional Structure-Activity Screening Datasets Using the Optimal Bit String Tree

This chapter has been submitted to *Technometrics* and has been tentatively accepted
by *Technometrics*

Analysis of High-Dimensional Structure-Activity Screening Datasets Using the Optimal Bit String Tree

Ke Zhang¹, Jacqueline M. Hughes-Oliver¹,
S. Stanley Young²

¹Department of Statistics, North Carolina State University, Raleigh, NC 27695-8203

²National Institute of Statistical Sciences, Research Triangle Park, NC 27709-4006

Abstract

A new classification method called the Optimal Bit String Tree is proposed to identify quantitative structure-activity relationships (QSARs). The method introduces the concept of a *chromosome* to describe the presence/absence context of a combination of descriptors. A descriptor set and its optimal chromosome form the splitting variable. A new stochastic searching scheme that contains a weighted sampling scheme, simulated annealing, and a trimming procedure optimizes splitting variable. Simulation studies and application to screening monoamine oxidase (MAO) inhibitors show that OBSTree is advantageous in accurately and effectively identifying QSAR rules and finding different classes of active compounds.

KEY WORDS: Classification; Drug discovery; High throughput screening; Prediction; QSAR; Simulated annealing.

4.1 Introduction

With the development of high throughput screening (HTS) techniques, large structural-biological activity data sets are now the norm rather than the exception (Sittampalam, Kahl, and Janzen 1997; Silverman, Campbell, and Broach 1998). Consequently, effective methods capable of managing large amounts of HTS data and converting it into utilizable information are in great demand. It is an accepted tenet of medicinal chemistry that molecular structure is highly related to biological activity as measured in an *in vitro* assay (McFarland and Gans 1986); two compounds with similar chemical structure are very likely to have similar biological activity with respect to one or more assays. Therefore, finding quantitative structure-activity relationships (QSARs) plays a critical role in analyzing chemical data sets.

A QSAR problem can be described as a classification or regression problem in statistics; we focus on the classification version throughout the paper. The biological activity of a compound serves as the response variable, while the predictor variables, which are also referred to descriptors, represent quantitative topological descriptions of the compound's structure. The task of classifying compounds is complicated by at least three features of HTS data. First, datasets can be quite large (with respect to number of compounds, number of predictors, or both). Second, biologically active compounds tend to be rare, so that screening data is relatively uninformative about active compounds. And third, even though actives are rare, there may be several different kinds of actives, each with its own QSAR that needs to be identified. Traditional statistical methods may not be suitable for determining all such QSARs.

Tree-based classification methods (Hawkins and Kass 1982, pp. 269-302; Breiman, Friedman, Olshen, and Stone 1984) divide a diverse, heterogeneous group of objects into several homogeneous sub-classes according to specific descriptor rules, and each terminal class consists of compounds sharing some common chemical structures. Recursive Partitioning (RP) is a particular class of tree-based methods where partitioning is performed recursively to grow a tree. This greedy iterative procedure is simple, yet powerful, and is able to identify complex nonlinear relationships and strong interactions existing in large structure-activity datasets (Young and Hawkins 1995; Chen, Rusinko, Tropsha, and Young 1999; Rusinko, Farmen, Lambert, Brown, and Young 1999).

We propose an RP-based method called Optimal Bit String Tree (subsequently referred to as OBSTree) that incorporates a weighted sampling scheme (for efficiency) and a trimming procedure (for robustness to user input). The method assumes binary structural descriptors are available for all compounds. One main feature of this method is introduction of the new concept of chromosome for identifying key splitting chemical sub-structures. Each chromosome is a binary string of 1-0 bits with one indicating the presence of a molecular sub-structure as represented by a descriptor and zero indicating the absence of that molecular sub-structure. A set of K descriptors combined with a specific chromosome forms the splitting variable of OBSTree, which results in a two-way split. Only those compounds that contain the exact splitting rule where all K selected descriptors take values exactly according to the selected chromosome are included in the positive daughter node. The remaining compounds go to the negative daughter node. A specially designed splitting criterion is needed

to account for the fact that we are not necessarily attempting pure separation for both nodes, only for the positive daughter node. Indeed, because we anticipate the existence of multiple mechanisms, the negative node is expected to contain active compounds, and thus be impure, during the initial splitting process.

The optimal splitting variable of OBSTree is determined by a two-stage procedure. In the first stage, simulated annealing (SA) is combined with a weighted sampling scheme in order to search for an optimal splitting variable consisting of K descriptors and a K -bit chromosome, where K is a pre-specified number larger than the number of descriptors required to define an implicit biological mechanism. The second stage includes a trimming procedure where the number of descriptors and length of the chromosome identified in the first-stage is reduced from K to an optimal K^* .

The remainder of this paper is organized as follows. In Section 4.2, three related models are reviewed and the main advantages of OBSTree are outlined. These related models are: Statistical Classification of Activities of Molecules (SCAM) (Rusinko et al. 1999), Binary Formal Inference-Based Recursive Modeling (BFIRM) (Cho, Shen, and Hermsmeier 2000), and Recursive Partitioning/Simulated Annealing (RP/SA) (Blower, Fligner, Verducci, and Bjoraker 2002). Section 4.3 contains a detailed description of OBSTree. Section 4.4 demonstrates performance on two at least partially simulated multiple activity mechanism data sets, and Section 4.5 considers application to the well-known dataset of screening for monoamine oxidase (MAO) inhibitors from Abbott Laboratories (Brown and Martin 1996). Concluding remarks are given in Section 4.6.

4.2 Recursive Partitioning

SCAM (Rusinko et al. 1999) uses binary descriptors to grow RP trees for the purpose of uncovering QSARs. Rusinko et al. (1999), along with Young and Hawkins (1995), were among the first to use RP for QSAR identification. Full chemical structure of every compound in SCAM is described by a bit-string indicating the presence or absence of different molecular sub-structures. For each split, one descriptor is chosen as the splitting variable such that compounds with a 1-value for this descriptor form the positive daughter node and compounds with a 0-value for this descriptor form the negative daughter node. A splitting criterion is needed to evaluate the splitting performance of each descriptor candidate. SCAM uses a t-test for comparing means of the two daughter nodes as the splitting criterion and a multiple-testing adjusted p-value to determine terminal nodes.

A number of other splitting criteria have been developed and implemented. Some, such as deviance, entropy, and the Gini Index, have been particularly useful when applied to categorical response variables (see, e.g., Breiman et al. 1984; Hastie, Tibshirani, and Friedman 2001). The descriptor that optimizes the splitting criterion, i.e., the one that yields the most different daughter nodes, is selected to be the final splitting variable. The resulting daughter nodes then become parent nodes for other splits. This recursive partitioning continues until some stopping criterion, such as a pre-specified threshold of p-value in SCAM, is met. Each terminal node of the SCAM tree is regarded as a class of compounds that share similar chemical structures and biological activity. It is assumed that critical molecular sub-structures responsible

for biological activity are identified as the combination of all the splitting descriptors that connect the terminal node and the root node.

The main disadvantage of using a single descriptor as the splitting variable is that a single descriptor may not be able to adequately identify a class of compounds or fully explain a biological activity mechanism. In other words, an active class is generally identified by a pathway containing more than one split in a standard RP tree. Those single splitting descriptors that appear in the early stage of a standard RP tree may be collectively important for one activity class, but these splits can result in adverse separation of other activity classes. Using a combination of descriptors to define each split can efficiently overcome this drawback, albeit at the cost of greater computational complexity.

BFIRM was proposed by Cho et al. (2000). It extends SCAM in that a multiple-descriptor strategy was employed to perform splitting at each node. Selection of the set of binary descriptors for a split starts with a ranking procedure that sorts all descriptors from best to worst based on an F-statistic, with higher values being more optimal. The set of multiple descriptors for a split is formed by greedily incorporating descriptors if their F-statistic exceeds a pre-specified threshold. Cho et al. (2000) show that better QSAR results are achieved with multiple descriptors due to the more informative splitting rule identified by a set of descriptors instead of a single descriptor.

Blower et al. (2002), however, argue that the best single descriptors found by BFIRM do not necessarily guarantee that the combination is also the best in op-

timizing the splitting criterion. It is likely that some descriptors can only become important when they are considered in a combination. Thus, the search space should consist of all possible combinations of K descriptors, where K is a pre-specified number. The RP/SA technique of Blower et al. (2002) is one such multiple-descriptor search strategy. Blower et al. (2002) argue that their optimal splitting set may include several descriptors that are not individually important. A two-way split is formed with positive daughter node consisting of compounds simultaneously having all splitting descriptors and negative daughter node consisting of the remaining compounds. The optimal set of descriptors is identified using SA, which is a stochastic optimization tool.

The splitting procedure of RP/SA implies that the positive daughter node is more homogeneous than the negative daughter node since the combination of descriptors are simultaneously present in all compounds in the positive node. Consequently, an ideal situation for RP/SA is that the positive node at each split creates a terminal node that represents a unique class of active compounds. Blower et al. (2002) show that a left-leaning RP/SA tree is always grown whenever RP/SA can effectively find the different classes of active compounds.

Both RP/SA and BFIRM automatically assume that only the presence of multiple descriptors is critical in deriving a significant split. This assumption works well when the different biological mechanisms can be fully explained by the presence of different sets of descriptors. There are situations, however, where absence of some descriptors also plays an important role in explaining activity mechanisms. Such cases include antagonistic sub-structures of a compound that can interact with the active sub-

structure of the compound to inhibit its potency. Both RP/SA and BFIRM have difficulty in identifying QSAR rules that include the absence of descriptors. It is essential for us to consider both the set of descriptors and the presence/absence of each descriptor within the set when forming the splitting variable. We also need a data-driven approach for deciding on an appropriate descriptor set size K .

4.3 OBSTree

We propose a new method, OBSTree, that allows the splitting variable to include absent descriptors. OBSTree merges the new concept of a chromosome with the old concept of a descriptor set to form the splitting variable. A chromosome is a string of 1-0 bits where each bit is the value of one of the descriptors of the descriptor set. Any set of K descriptors can have at most 2^K distinct chromosomes. It is obvious that the splitting variable of RP/SA is a special case of our OBSTree splitting variable because the splitting variables of RP/SA always uses the splitting chromosomes coded as a string of 1's.

Even for a modest size K , selection of the best K descriptors and corresponding chromosome is computationally quite challenging. If there are P descriptors available, then the search space for a single split is of size $\binom{P}{K}(2^K)$, and this can get very large even for small values of P . Like Blower et al. (2002), we employ SA for a stochastic search to identify the set of K descriptors. Unlike Blower et al. (2002), we use a weighted sampling scheme in SA, with a modified SA cooling procedure. Moreover, we perform complete enumeration over all chromosomes to investigate the effect of

both absent and present descriptors.

SA (Aarts and Korst 1989; Chib and Greenberg 1995) is an iterated hill climbing technique that pursues global optimization by moving between different local optima. These local optima may represent randomness in the response surface or different activity mechanisms. In theory, SA will converge to the global optimum, but in practice there is no such guarantee. SA is implemented in our algorithm to determine the optimal splitting variable, namely an optimal descriptor set and its corresponding optimal chromosome. Specific parameters for our implementation are given in Sections 3.2 and 3.4. For additional details on SA itself, we refer readers to the references listed above. Other search techniques, e.g. genetic algorithms or the exchange algorithm, can also be used in place of SA.

The motivation for developing a weighted sampling scheme is to combine, and take advantage of, both multiple-descriptors selection procedures found in RP/SA and BFIRM. Our argument is that although the individually best descriptors cannot guarantee the best combination they definitely have high probability of being included in forming the best combination. In other words, information gained from ranking descriptors individually should be considered in finding the optimal combination of the absent/present descriptors. The weighted sampling scheme requests that a group of individually important descriptors be pre-identified by running one or more standard RP trees. All descriptors that create a split in the standard RP tree are collectively referred to as “singly-important” and the remaining descriptors are the “general descriptor” group. These two groups together form a complete candidate descriptor set in OBSTree, but we sample from them at different rates. Hence, de-

scriptors within the same group are treated equally but the two groups have different sampling weights in the weighted sampling scheme. Additional details are given in Section 3.4.

Another advantage of OBSTree is that the number of splitting descriptors and the length of the corresponding splitting chromosome within OBSTree is selected by a data-driven trimming procedure. While RP/SA requires a pre-specified and fixed size K across different splits, we contend that it is difficult to select a sensible number without prior knowledge of the QSAR information contained in the data set and that this number may change from split to split. In contrast, OBSTree specifies a large size ($K = 10$ is the current setting) for the splitting sets in order to be sure to capture the key splitting features during the SA step. After the optimal splitting variable is selected by SA, the trimming procedure is used to delete noisy information from the splitting variable. This is done mainly by reducing the number of descriptors within the splitting set and consequently shrinking the length of the chromosome. The result is that K can vary over splits, thus allowing the data to identify the different amounts of information for each different activity mechanism. Details on chromosome selection and the trimming procedure are given in Sections 4.3 and 4.5.

Under the assumption of a categorical response, a new splitting criterion called penalty entropy is proposed to classify only the active compounds; details are given in Section 4.1. A flow chart of the entire OBSTree approach is shown in Figure 4.1.

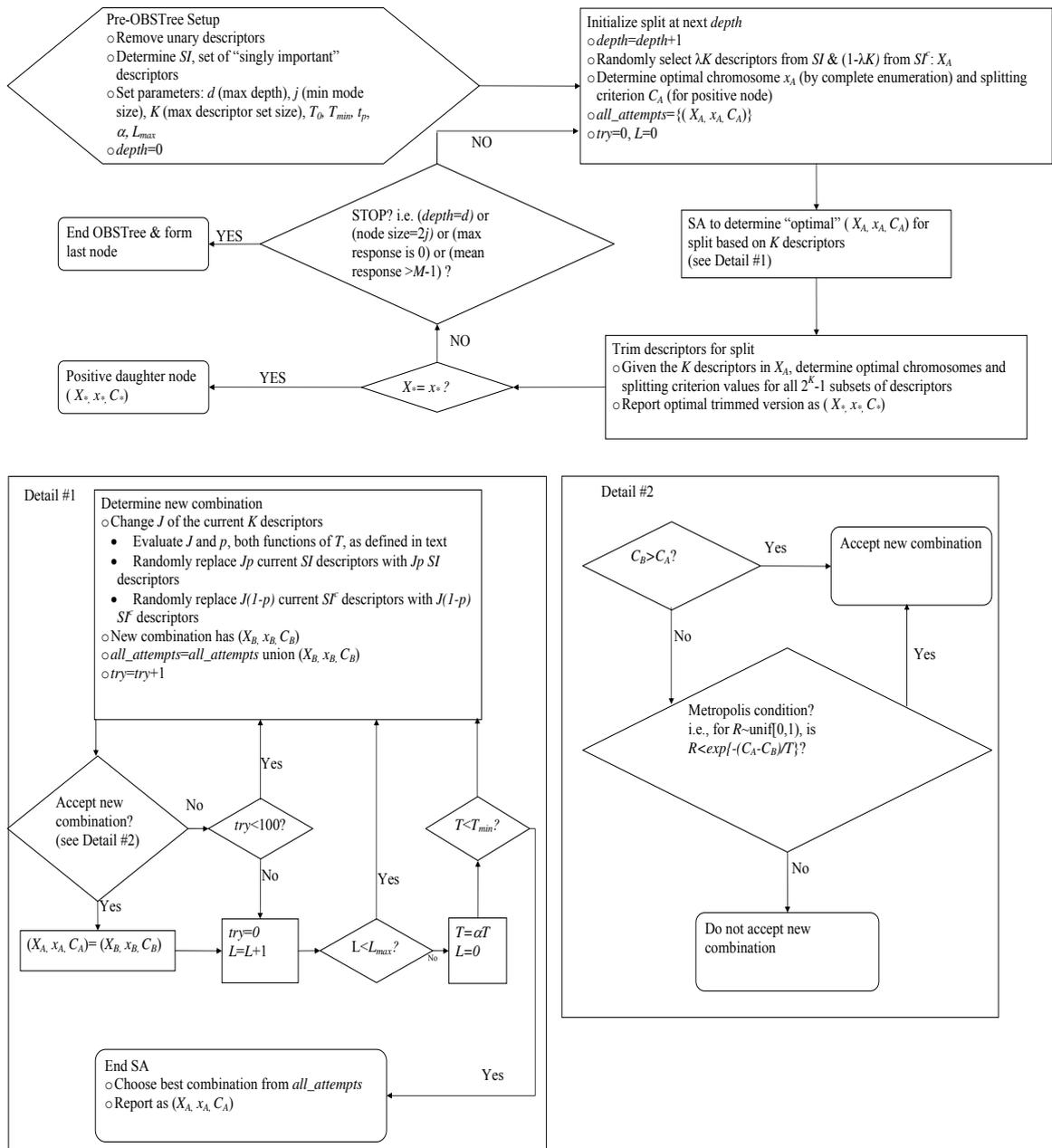


Figure 4.1: Flowchart for OBSTree

4.3.1 Splitting Criterion

Consider a categorical response with $M + 1$ outcomes $0, 1, \dots, M$. We assume compounds with zero response are inactive. Entropy is a measure of purity of nodes for categorical data. Suppose the $M + 1$ classes of outcomes have probabilities of $p_0, p_1, p_2, \dots, p_M$ in the node. Entropy (actually, the negative of entropy) for one daughter node is given as $\sum_{i=0}^M p_i \log p_i$ and takes value zero for a pure node. Conventionally, a weighted sum of entropies from two daughter nodes, with weights being the marginal probabilities of the two daughter nodes given the parent node, forms the splitting criterion.

Buja and Lee (2001) considered one-sided purity for two-class classification trees in order to find only one node with clear majority label. Similarly, our goal is to find a pure positive node on every split. In addition, we want to identify non-zero (active) compounds in the positive nodes. Therefore, both the purity of each positive node as well as the proportion of zeroes in the node should be taken into account. We propose a new splitting criterion called penalty entropy to evaluate splitting variables in OBSTree. The penalty entropy regards each zero within the positive node as a new pseudo category and calculates the general entropy of the positive node under the pseudo-category framework. Consequently, the penalty entropy of a node is defined as

$$p_0 \log \frac{1}{N} + \sum_{i=1}^M p_i \log p_i,$$

where N is the total number of compounds in the node.

The best splitting variable should maximize the penalty entropy and the optimal

value that a split can yield is zero. The advantage of this criterion is the ability to penalize a pure node of zeroes. To illustrate this point, suppose there are two possible splits. Split one yields a positive node with 10 compounds with response three and split two yields a positive node with 10 compounds with response zero. They are judged equally good by entropy, even though split two gives a class of junk compounds and is of no interest in our studies. Penalty entropy treats every zero within the positive node as a new pseudo class. Therefore, split two yields a positive node containing 10 pseudo classes and the penalty entropy value is $\log(\frac{1}{10})$. The penalty entropy value of the first split is still zero. Thus, the first case is more likely to be selected as a better split with the penalty entropy method. Notice that we only focus our calculation on the penalty entropy of the positive daughter node instead of measuring entropy difference between the parent node and the daughter nodes induced by the split. It is reasonable to only consider the positive daughter node since the most interest lies in identification of a homogeneous class of active compounds in a positive node.

4.3.2 Initial Conditions

OBSTree requires several inputs and initial conditions. First, the descriptor pool is obtained by removing all unary (takes only a single value) descriptors from the data set. Second, the descriptor pool must be subdivided into “singly-important” and “general descriptor” groups, as described above. Third, a depth d must be given as the maximum tolerable depth for growing the tree. This results in at most $d+1$ terminal nodes. Fourth, a minimum node size j is needed to prevent meaningless splits

from nodes that are too small. Fifth, a maximum descriptor set size, K , is needed. Sixth, the relative sampling proportions λ and $1 - \lambda$ of the two groups of descriptors, i.e., different weights in weighted sampling, should be set. And seventh, initial conditions are needed for the SA, including the initial temperature T_0 , the minimum temperature cutoff T_{min} , a critical temperature t_p on which the sampling rates for the weighted sampling scheme depends, the maximum number of transitions L_{max} at a given temperature, the maximum number of failures, *try*, for attempting to replace current splitting variable that is allowed before the transition counter increases, and the temperature reduction rate α . Some guidance as to intelligent specification of those parameters is given in Section 7.

4.3.3 Optimal Chromosome Selection

Given a set of K descriptors, the optimal chromosome is identified by calculating the splitting criterion for values of all 2^K possible chromosomes of the descriptor set. Generally, there are multiple chromosomes that give the best splitting value using the penalty entropy criterion. Two conditions are checked sequentially to break ties. First, the tentative positive node size is considered. Those chromosomes with the largest positive node size are selected. A second tie-breaker is to choose the chromosome yielding the fewest inactive compounds in the positive node.

4.3.4 Simulated Annealing Portion

Temperature, which decreases at rate α , determines the next SA step from the current stage using several functional dependencies. The total number of descriptors to be replaced in the current splitting variable, J , is obtained as $J = K - 1$ if $T > 2$, $J = \lfloor K/2 \rfloor$ if $10^{-2} \leq T \leq 2$, and $J = \lfloor K/4 \rfloor$ if $T < 10^{-2}$. Clearly, J decreases as T decreases so that the combination of K descriptors becomes more stable with low temperature.

For controlling the weighted sampling procedure, define p ($0 < p < 1$) such that Jp descriptors within the “singly-important” group and $J(1 - p)$ descriptors within the “general descriptor” group are randomly selected and dropped from the current combination of K descriptors. Jp and $J(1 - p)$ descriptors are then randomly sampled from the group of “singly important” descriptors and the group of “general descriptors,” respectively, to replace the J dropped descriptors. The p is also a function of T given by $p = 0.6$ if $T > t_p$ and is sampled from Uniform(0,1) if $T \leq t_p$, where t_p is pre-specified based on T_0 and T_{min} . At least one descriptor from each group has to be changed at every search. When T is large, SA is very unstable so the guided search that results from setting $p > 0.5$ allows SA to quickly find some local optima. For small T , less guidance is needed so p is unconstrained. SA terminates when the current temperature is less than the pre-specified minimum temperature T_{min} , and the best splitting variable seen thus far is selected.

4.3.5 Trimming Procedure

When an optimal splitting variable containing an optimal splitting set of K descriptors and the corresponding optimal chromosome of length K is identified, it is reasonable to assume the existence of some noise within the splitting variable since a generally large K value is initialized to over-capture the active mechanism information. Deleting redundant or unrelated descriptors from the splitting variable makes the interpretation more sensible. Statistically, reducing the number of descriptors and shrinking the length of the corresponding chromosome may even improve the split. Thus, a trimming procedure is developed to select a subset of the optimal splitting variable identified by SA. The penalty entropy values of all $2^K - 1$ subsets of the splitting variable (except the empty set) are calculated. Subsets with the maximum splitting values are then selected, with ties broken as described earlier. If there are still multiple subsets left, the subset containing the fewest descriptors and the shortest length of chromosome is the final splitting variable. Therefore, this trimming procedure yields a variable size of the splitting sets at each partitioning, which is believed to be a more reasonable way to define active mechanisms than the splitting sets with fixed size.

4.3.6 Stopping Criteria

OBSTree recursively uses the SA algorithm to split nodes until the stopping criteria described below are met. One simple way is to specify a maximum tolerable depth to which the tree grows and to continue to this depth if splits are possible.

Alternatively, cross-validation (CV) can be used to choose the final depth within the maximum tolerable depth.

A stopping criterion is needed for CV. Given any consecutive depths, i and $i + 1$, and the number of active compounds that are inaccurately predicted at depth i ($n_{mis,i}$), a statistical test on π , the probability of correctly predicting an active that was previously misclassified, is conducted with $H_0^{i+1} : \pi \leq 0$ and $H_1^{i+1} : \pi > 0$. Assuming the $n_{mis,i}$ compounds represent independent and identically distributed Bernoulli trials, the maximum likelihood estimator of π is $\hat{\pi} = \frac{\Delta n}{n_{mis,i}}$, where Δn is the increment of actives correctly predicted from depth i to $i + 1$. A large-sample standard normal form of Wald-type statistic (Casella and Berger 2002, p. 485) rejects H_0^{i+1} if

$$Z = \frac{\hat{\pi} - 0}{\sqrt{\hat{\pi}(1 - \hat{\pi})/n_{mis,i}}}$$

is large. The final depth d is set to i when the null hypothesis H_0^{i+1} is first accepted at significance level α ($= 0.05$ by default) during the tree growing process. Note that the final depth d will be at most equal to the maximum tolerable depth. In other words, the tree will stop at the maximum tolerable depth even if this test is never accepted.

Three other default stopping criteria are: (a) a node will be a terminal node if the maximum potency of individual compounds in the node is zero, i.e., it contains only inactive compounds; (b) a node will be a terminal node if the node size is less than or equal to the $2j$, i.e., daughter nodes would violate the requirement for minimum node size j ; (c) a node will be a terminal node if the mean of the node is greater than $M - 1$, i.e., it contains mostly highly active compounds if the categorical response is

Table 4.1: Four Biological Mechanisms Specified in Terms of the Descriptor Sets and the 1-0 Chromosomes.

Mechanism Type	Potency	Active Mechanism						
I	3	1	2	3	4	5	←	descriptor set
		1	0	1	0	1	←	chromosome
II	3	5	6	7	8	9		
		0	1	1	1	1		
III	2	3	11	12	13	17		
		1	1	1	1	1		
IV	1	15	16	17	18	19		
		1	1	0	1	1		

actually ordinal.

4.4 Simulation Studies

4.4.1 Study 1: Simulated Descriptors and Simulated Responses

Data Set

A synthetic dataset was constructed to contain 1000 compounds, 500 binary descriptors, and an activity rate of 8% corresponding to a four-level categorical response. The 80 active compounds were created to arise from four different “activity mechanisms” whose descriptor set, corresponding chromosomes, and responses are given in Table 4.1.

This simulated dataset has several interesting and realistic features. First, only 17 of the 500 descriptors are relevant for describing QSARs, so feature selection will be critical. Second, Mechanisms I and II are partially confounded in that they both give

rise to a response value of three and they are both defined using Descriptor 5. Other mechanisms are also partially confounded, albeit to a lesser extent than Mechanisms I and II: Mechanisms I and III share Descriptor 3 while Mechanisms III and IV share Descriptor 17. Partial confounding of two mechanisms imply that RP where each partition is based on only one descriptor can adversely break apart one mechanism in search of the other mechanism; see Figure 4.2(a) and its discussion below.

Third, all descriptors are generated to be rare and thus representative of atom pair descriptors (Carhart, Smith, and Venkataraghavan 1985). In fact, all junk descriptors (those 483 not used to describe any mechanism) are generated as independent Bernoulli with success probability 0.01. Letting \mathbf{W} represent the vector of 17 important descriptors for a compound, we limit its possible values to those for which either exactly one mechanism is present or no mechanisms are present. For any particular mechanism, sampling is done without replacement from the set of possible realizations of \mathbf{W} where only that mechanism exists. For all junk compounds, sampling is done without replacement such that no mechanisms exist.

Predictive Tree Models

A standard RP tree was grown by the `tree` function of the `tree` package (Ripley 2005) in R (R Development Core Team 2004) using deviance as the splitting criterion. Several other RP techniques were investigated, including the `rpart` (Therneau and Atkinson 2005) and `randomForest` (Breiman and Cutler 2005) packages in R, Partitionator (Lambert 2004), and SAS/Enterprise Miner 4.1 (SAS Institute, Inc. 2000), but `tree` gave, by far, the best result. The result from `tree` is shown in Figure 4.2(a),

and the result from OBSTree is shown in Figure 4.2(b).

The CV approach of Section 3.6 terminated OBSTree at depth four, which exactly matches the number of mechanisms contained in the dataset. For identifying “singly important” descriptors, we combined information from three different implementations of standard RP: `tree` (Ripley 2005) from R, which uses the deviance criterion; Partitionator (Lambert 2004), which uses an F-test criterion; and SAS/Enterprise Miner 4.3 (SAS Institute, Inc. 2004) with entropy reduction criterion. Descriptors appearing in at least one of these three trees are included in the group of “singly important” descriptors; there are 18 such descriptors.

Although the standard RP tree finds compounds from all four active groups, it only finds seven of the 20 compounds in Mechanism I, five of the 20 compounds in Mechanism II, and it misclassifies five junk compounds as falling in Mechanism II. The main reason for the worst result of the standard RP tree is that using a single descriptor to make splits for identifying one class of active compounds can inadvertently divide other active classes. In our example, the first split using Descriptor 17 is good for Mechanisms III and IV but caused fractures in Mechanisms I and II that prevented the tree from completely finding these mechanisms.

OBSTree is able to identify a complete class of active compounds on every split because it uses both a descriptor set and chromosome to capture an exact biological activity mechanism. This approach reduces dependence between different splits of the tree so that the different active classes can be effectively found. Moreover, it is clear that the trimming procedure is effective. In all cases it was able to trim the

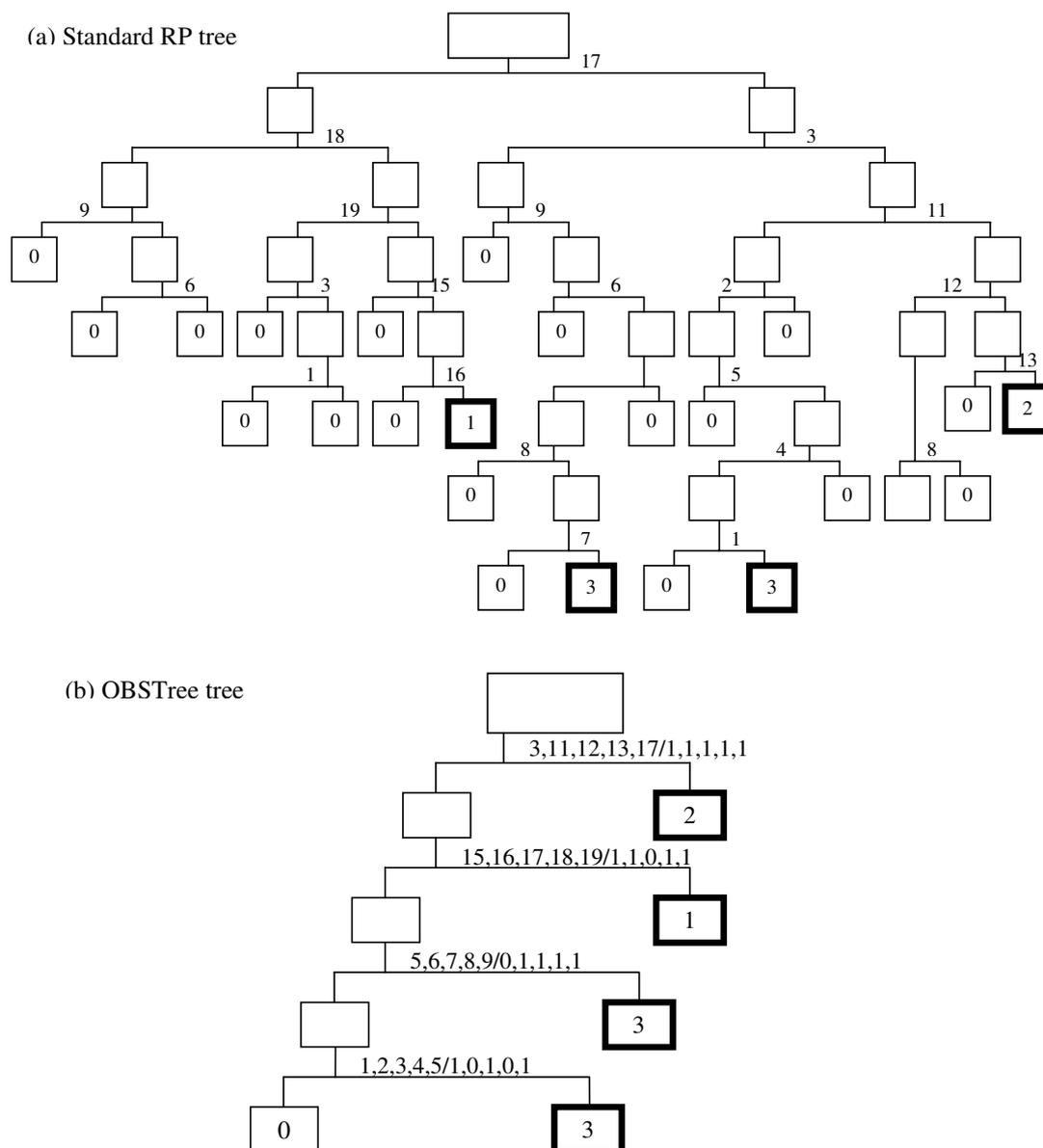


Figure 4.2: (a) The standard RP tree of Study 1 with minimum node size 5. Values shown in the terminal nodes are response values for the majority class of the node. Active terminal nodes are indicated with thick borders. The distributions of compounds for the four active terminal nodes are 0/20/0/0, 5/0/0/5, 0/0/0/7 and 0/0/20/0, respectively, from left to the right, where the numbers of compounds of four different potency levels (i.e. 0, 1, 2 and 3) are separated by “/”. (b) An OBSTree tree of Study 2; $T_0 = 12$, $T_{min} = 0.001$, $\alpha = 0.99$, $K = 10$, $t_p = 1$, $L_{max} = 3$ and minimum node size is 5. Splitting variables are given for each split with the set of descriptors and the corresponding chromosome separated by “/”. Distributions of compounds for the four active terminal nodes are 0/0/20/0, 0/20/0/0, 0/0/0/20 and 0/0/0/20, respectively, from the highest depth to the lowest depth.

excess noise variable to reduce K from 10 (the default starting value) to the required five for all mechanisms.

Sensitivity Analysis

The effect of parameter K was examined by initializing OBSTree with different K s. OBSTree models could precisely identify the four activity mechanisms and fully capture all compounds for each class when $K = 10$ and 12, whereas the models failed in identifying one mechanism when $K = 7$ and failed in identifying all mechanisms when $K = 4$. The results suggest that any comparably large K combined with trimming procedure will yield good models. It should be noted, however, that an extremely large K significantly increases the computational burden.

In order to evaluate the prediction performance of OBSTree and the possible existence of overfitting, we used 5-fold CV to check the prediction or extra-sample error (Hastie et al. 2001). Three methods were compared: OBSTree; standard RP implemented in `tree` (Ripley 2005) using deviance (this was the best of all standard RP implementations that we used); and RP/SA as defined by Blower et al. (2002) using $K = 3$. All trees were limited to a depth of $d = 4$ even though such a depth gives unfair advantage to `tree` since it allows up to 16 terminal nodes for `tree` but only five terminal nodes for left-leaning trees such as OBSTree and RP/SA.

The predicted value of every terminal node is determined by the response value of the majority class in the node. Confusion matrices are given in Table 4.2. OBSTree gives the best overall performance with overall accuracy percent being 98.8%. If we focus only on identification of active compounds, the 88.75% ($= (20 + 20 + 31)/80$)

overall active accuracy percent of OBSTree far exceeds the 0% for both standard RP and RP/SA. OBSTree correctly classifies 71 of the 80 active compounds with a reasonable 77.5% hit rate for potency value of three, which is the most difficult to predict. Standard RP cannot predict any active compounds by depth 4 due to its inefficiency in finding active nodes. It has to grow to depth 5 or deeper to identify any activity mechanism for this data set. This is illustrated by the “exhaustive” confusion matrix generated by growing standard RP trees on each fold until minimum node sizes are achieved (four of them grow to depth 7). It can be seen that exhaustive standard RP gives excellent performance for predicting potency values one and two, but it is very poor at predicting potency value three. Recall that standard RP suffers from inadvertent separation of compounds having potency value three; see Figure 4.2. RP/SA performs quite badly on this data set, predicting no compound as active. The failure of RP/SA is predictable since the splitting variable of RP/SA has difficulty in identifying absence of descriptors.

Additional results (not presented here) on a similarly constructed dataset with only 51 descriptors also demonstrated OBSTree’s ability to quickly find QSARs without employing the weighted sampling scheme, i.e., without pre-screening descriptors to identify a set of singly important descriptors.

4.4.2 Study 2: McMaster and Xue Compounds with Simulated Responses

The second dataset is a mixed one consisting of 8076 real compounds with only the response values being simulated. 8000 compounds were randomly selected from

Table 4.2: Confusion Matrix Comparison for Study 1 (P=Prediction, T=True, Acc.=Accuracy, O.A.=Overall Accuracy)

Standard RP		T				Acc. %
		0	1	2	3	
P	0	920	20	20	40	92
	1	0	0	0	0	N/A
	2	0	0	0	0	N/A
	3	0	0	0	0	N/A
Hit%		100	0	0	0	O.A.: 92.0

RP/SA		T				Acc. %
		0	1	2	3	
P	0	918	20	20	40	91.98
	1	2	0	0	0	0
	2	0	0	0	0	N/A
	3	0	0	0	0	N/A
Hit%		99.78	0	0	0	O.A.: 91.8

OBSTree		T				Acc.%
		0	1	2	3	
P	0	917	0	0	9	99
	1	0	20	0	1	100
	2	0	0	20	0	100
	3	3	0	0	31	91.2
Hit%		99.6	100	100	77.5	O.A.: 98.8

Exhaustive RP		T				Acc. %
		0	1	2	3	
P	0	910	1	0	34	96.3
	1	3	19	0	0	86.4
	2	0	0	20	0	100
	3	7	0	0	6	46.2
Hit%		98.9	95	100	15	O.A.: 95.5

the training set that was provided by HTS Data Mining and Docking Competition. The competition was sponsored by the McMaster University HTS Lab (please refer to website <http://hts.mcmaster.ca/HTSDataMiningCompetition.htm> for further details about the competition). We assigned all of these compounds to be inactive, i.e. to have response zero. For our active compounds, we used the Xue and Bajorath (2002) collection, consisting of a diverse data set of 21 different biological activity classes of compounds, from which we randomly selected four classes of compounds as our active compounds. The four groups include COX with 17 compounds, H3E with 21 compounds, HIV with 18 compounds and TKE with 20 compounds. The potencies of COX and H3E were set to one, and those of HIV and TKE were set to two. 1260 atom pair descriptors (Carhart et al. 1985; Rusinko et al. 1999) were calculated using PowerMV (Liu, Feng, and Young 2004).

The following variable selection process was applied to the dataset before running OBSTree. An ensemble of 500 trees were generated from ChemTree (Golden Helix Inc. 2000; Hawkins and Kass 1982) and 300 descriptors appearing at least once in 500 trees formed our descriptor pool. Output from `tree`, SAS/Enterprise Miner 4.1, and Partitionator were combined to identify a set of “singly important” descriptors. $K = 7$ and the same initial conditions as those set for the first simulated data were implemented to run OBSTree. The Wald test statistic of 5-fold CV identified the maximum depth of $d = 5$ in OBSTree. Three predictive models, OBSTree, Standard RP and RP/SA, were developed using the full data and compared up to depth 5. The confusion matrices of these models from 5-fold CV analysis are shown in Table 4.3.

OBSTree gives the best overall performance with highest overall accuracy of 64.5%

Table 4.3: Confusion Matrix Comparison for Study 2 (P=Prediction, T=True, Acc.=Accuracy, O.A.=Overall Accuracy)

Standard		T			Acc. %
RP		0	1	2	
P	0	7989	30	7	99.5
	1	7	8	0	53.3
	2	4	0	31	88.6
Hit%		99.9	21.1	81.6	O.A.: 99.4

RP/SA		T			Acc.%
		0	1	2	
P	0	7987	36	9	99.4
	1	0	2	0	100
	2	13	0	29	70.7
Hit%		99.8	5.3	76.3	O.A.: 99.3

OBSTree		T			Accuracy %
		0	1	2	
P	0	7992	24	3	99.7
	1	4	14	0	77.7
	2	4	0	35	89.7
Hit%		99.9	36.8	92.1	O.A.: 99.6

(= (14 + 35)/76) for identifying actives, compared to 51.3% for standard RP and 40.8% for RP/SA.

4.5 MAO Data Set Application

4.5.1 Predictive Tree Models and Cross Validation

A set of 1646 monoamine oxidase (MAO) inhibitors was provided by Abbott Laboratories (Brown and Martin 1996). There is some continued interest in MAO as a target for rational drug design in antidepressant therapy (Rusinko et al. 1999). The potencies are reported as an ordinal variable that can take the four values 0, 1, 2, and 3, where 0 indicates a junk compound. Potency values of 1, 2 and 3 are all regarded as active compounds, with three indicating the greatest level of activity. Again, AP descriptors were calculated by PowerMV. Of the 4662 atom pairs provided by PowerMV, only 1380 exhibited variability, so only these were retained in the data set.

We again combined output from `tree`, SAS/Enterprise Miner, and Partitionator to identify a set of “singly important” descriptors that was used as input for generating OBSTree tree. The Wald test of 5-fold CV suggested that a depth of two was sufficient. According to all the published literature for this MAO data set, two activity mechanisms can be clearly identified, both associated with potency value of three; see, for example, Rusinko et al.(1999). This literature also consistently demonstrates difficulty in identifying mechanisms for potency values one and two.

Table 4.4: Confusion Matrix Comparison for MAO Data Set (P=Prediction; T=True; Acc.=Accuracy; O.A.=Overall Accuracy.)

Standard RP		T				Acc. %
		0	1	2	3	
P	0	1339	112	78	37	85.5
	1	0	0	0	0	N/A
	2	0	0	0	0	N/A
	3	19	2	8	51	63.8
Hit%		98.6	0	0	58.0	O.A.: 84.4

RP/SA		T				Acc. %
		0	1	2	3	
P	0	1352	114	82	57	84.2
	1	0	0	0	0	N/A
	2	0	0	0	0	N/A
	3	6	0	4	31	75.6
Hit%		99.6	0	0	35.2	O.A.: 84.0

OBSTree		T				Acc. %
		0	1	2	3	
P	0	1351	113	82	43	85.0
	1	0	0	0	0	N/A
	2	0	0	0	0	N/A
	3	7	1	4	45	78.9
Hit%		99.5	0	0	51.1	O.A.: 84.8

Table 4.4 shows confusion matrices from 5-fold CV analysis of OBSTree, standard RP, and RP/SA, each using minimum node size of five.

Overall accuracy is around 84 percent for all three methods, but (matching reports from the literature) they all perform rather poorly for predicting potency values of one or two). For this data set, the biggest benefit offered by OBSTree is interpretation of the molecular sub-structures identified as being important.

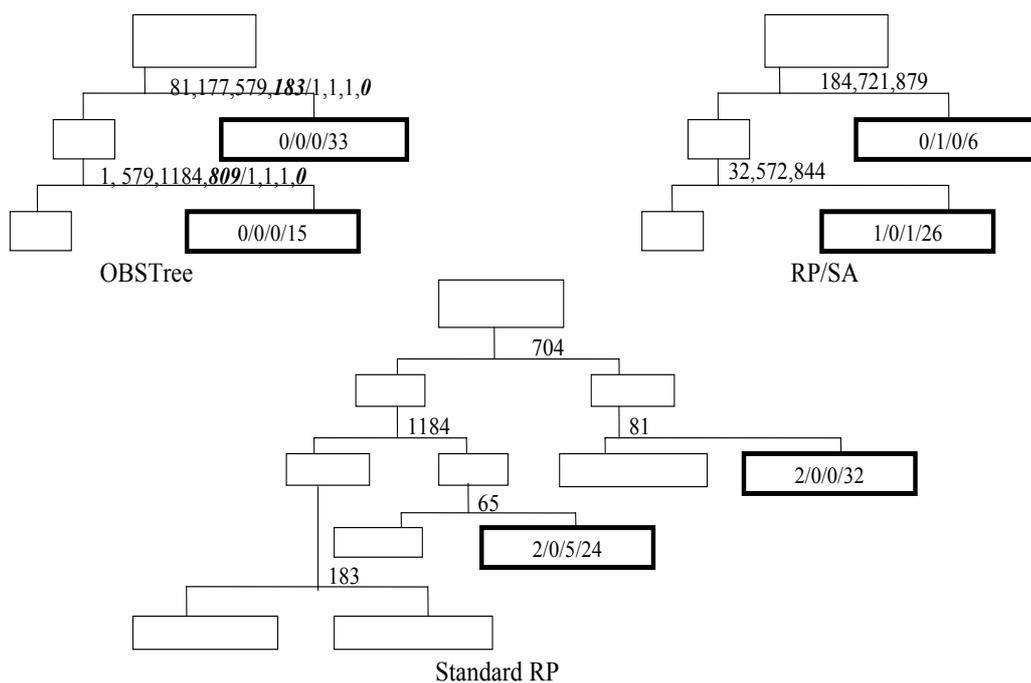


Figure 4.3: Tree model comparison for MAO. Tree from OBSTree with penalty entropy splitting criterion, tree from RP/SA with absolute difference splitting criterion and the standard RP tree with deviance splitting criterion. Active terminal nodes are indicated with thick borders. Splitting variables are given for all splits. The distributions of compounds are shown in the terminal nodes, where the numbers of compounds of four different potency levels (i.e. 0, 1, 2 and 3) are separated by “/”.

4.5.2 Chemistry Interpretation

Chemical interpretation is of great interest in QSAR studies. For illustration purposes, the training set of the 2^{nd} fold in our 5-fold CV study, which was generated by removing the 2^{nd} fold from the data set, is used to develop three different tree predictive models. Figure 4.3 shows portions of the trees where the first two active groups are identified.

The chemical relevance of the first split obtained from OBSTree can be interpreted

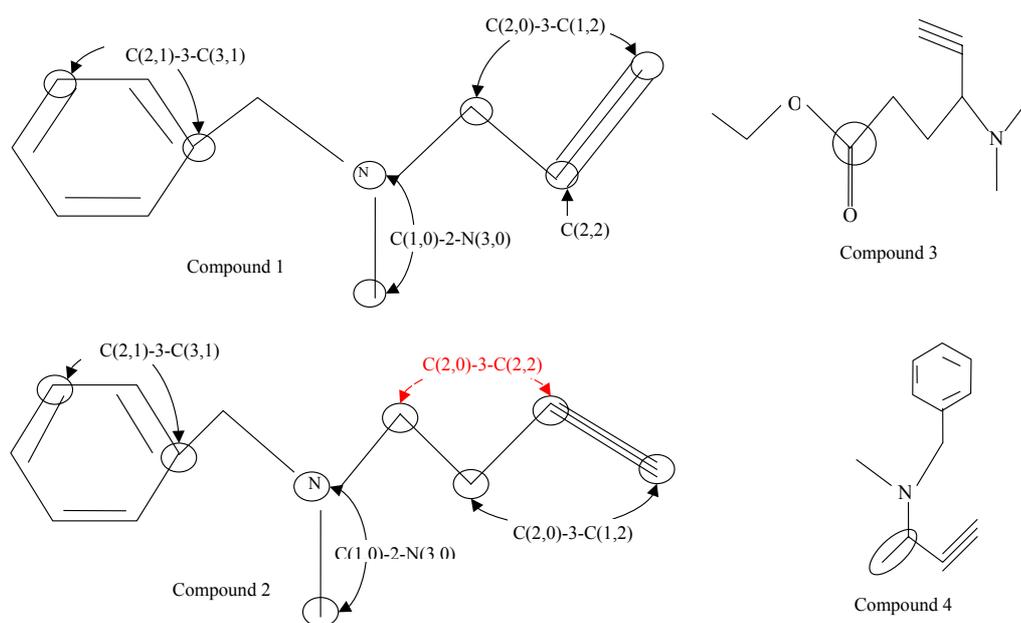


Figure 4.4: Sample compounds in MAO data set. Compound 1, which is called Pargyline, is well known as an MAO inhibitor with potency three. Compound 2 has potency value zero; Descriptor 183, identified by OBSTree as needing to be absent for activity, is represented by the dashed arrows. Compounds 3 and 4 are both junk compounds, even though they appear in the first active terminal node of the standard RP tree. Compound 4 also appears in the second active terminal node of the RP/SA tree.

as simultaneous presence of three AP descriptors (Descriptor 81:C(1,0)-2-N(3,0), Descriptor 177:C(2,0)-3-C(1,2) and Descriptor 579:C(2,1)-3-C(3,1)) and the absence of one AP descriptor (Descriptor 183:C(2,0)-3-C(2,2)). Maycock et al. (1976), Chen et al. (1999) and Rusinko et al. (1999) all report that the drug pargyline (Compound 1 in Figure 4.4) can irreversibly inhibit the neuronal monoamine oxidase by covalently attaching to the MAO flavin cofactor. Descriptors 81, 177, and 579 co-identify a tertiary nitrogen, a triple bond terminus, and an aromatic ring (Rusinko et al. 1999), respectively, which are all key features for pargyline-like inhibitors.

As indicative of the power of our method, OBSTree also identified an absent Descriptor 183:C(2,0)-3-C(2,2). This, combined with present descriptors, suggests that the distance between triple-bond terminus and another intermediate carbon or other focal active sub-structures such as tertiary nitrogen is critical. This is consistent with the statement by Chen et al. (1999) that the vicinity of the triple-bond terminus may significantly affect the activity of the inhibitors. Figure 4.4 gives an example compound (Compound 2) in the MAO data set that supports the finding by OBSTree and Chen et al. (1999) that Descriptor 183 should be absent from pargyline-like compounds. The experimental evidence shows that the potency of Compound 2 is zero. Compound 2 has all the active sub-structures identified in pargyline. The only difference between Compound 2 and pargyline is that it contains one more carbon in the chain $\text{-C-C-C}\equiv\text{C-}$, i.e., it possesses the Descriptor 183:C(2,0)-3-C(2,2) that OBSTree says should be absent in order for the compound to be an MAO inhibitor.

The first active terminal node of the standard RP tree identified the same class of compounds. It, however, only captured part of the required chemical sub-structures. Also, the neighborhood effect on the triple-bond terminus was not detected by the standard RP tree. Therefore, two junk compounds (Compounds 3 and 4 in Figure 4.4) and one fewer active compound were included in the node.

The RP/SA tree only identified a small part of the compounds contained in the first node of OBSTree. Descriptor 721:C(3,1)-5-O(2,0) of RP/SA tree, pointing to a nonterminal oxygen atom bonded to the phenyl, is not generally a key feature of the mechanism. It excludes a number of similar, active compounds from the first OBSTree node. The second terminal node of RP/SA contains the remaining compounds of this

class. It is very likely that an inappropriate setting of the number of descriptors in the splitting set (K) caused this problem. In contrast, the trimming procedure of OBSTree allows us to specify less conservative conditions on this parameter initially because we eventually drop the noninformative portion of the rules from the final splitting variable.

Similar comments and comparison hold for the other identified MAO mechanism. OBSTree tends to identify larger homogenous active groups than RP and RP/SA while capturing more relevant information on QSARs.

4.5.3 Stability of OBSTree

Stability of OBSTree was investigated using the Q measure proposed by Blower et al. (2002) for quantifying the effectiveness of a tree in identifying active groups of compounds and used by them to evaluate the stability of RP/SA. Large Q is optimal, where $Q(n) = (c/n) \cdot a_1 + (1 - c/n) \cdot a_2$, a_1 is the average activity of all positive terminal nodes occurring above the depth d_Q at which the n^{th} positive terminal node occurs, a_2 is the average activity of all positive terminal nodes occurring at the depth d_Q , and c is the number of positive terminal nodes occurring above (but not at) depth d_Q . Ten trees, each differing only by starting seeds used for the stochastic component of the algorithm, were grown using OBSTree for this MAO dataset and their Q values with $n = 2$ (because trees were grown only to depth $d = 2$) were compared. All ten trees resulted in pure positive terminal nodes (see, e.g., Figure 4.3), so Q was exactly three (its highest possible value) for all trees, thus indicating stability of the algorithm.

Table 4.5: Stability of CV trees

Fold	Activity Mechanism I		Activity Mechanism II	
	Number of total compounds	Percent % (Compounds in common)	Number of total compounds	Percent % (Compounds in common)
1	33	54.6	16	68.8
2	36	86.1	17	64.7
3	32	84.4	12	91.7
4	35	88.6	27	37.0
5	34	76.5	17	64.7

In addition, stability of OBSTree to perturbations of the training data was also studied since small perturbations of training set can result in dramatically different trees (Breiman 1998; Svetnik et al. 2003). We compared the complete tree grown from the full dataset to each CV tree, namely each of the five trees grown from the 5-fold CV exercise. OBSTree is considered stable to data perturbations if the nodes generated from each CV tree match the corresponding node from the complete tree. To assess the degree of matching, we recorded the percent of compounds common to both nodes. These percents are necessarily bounded away from 100% because each CV tree excludes 20% of the compounds.

As seen in Table 4.5, every CV tree captures more than 50% common compounds in activity mechanism I. Actually, four of them are more than 75%. Mechanism II is more sensitive to data perturbation, but four of the five folds capture more than 60% common compounds. The results show that OBSTree is fairly stable even though the perturbation rate of the training set is relatively large (20% hold-out data).

4.6 Discussion AND Summary

OBSTree is an effective algorithm for exploring QSAR rules of high-dimensional structure-activity data sets; our applications considered up to 1380 descriptors and 8076 compounds. OBSTree differs from standard RP and RP/SA in that both the presence and absence of descriptors are simultaneously taken into account to form the splitting variable. According to the results from the real data MAO application, present descriptors generally identify the active functional features of the compounds while absent descriptors generally point to the features that could decrease the activity of those functional features. In contrast, both standard RP and RP/SA tend to only identify the functional features of the compounds and consequently yield more heterogeneous terminal active classes. In addition, it is very likely that standard RP will successfully identify one active class but simultaneously divide members of other active classes. RP/SA may also inadvertently subdivide active classes due to inappropriate specification of the number of descriptors involved in a splitting set (K). We believe that OBSTree is able to overcome these drawbacks to yield a better predictive model.

OBSTree successively “peel off” an active node from the root node, resulting in a “left-leaning” tree rather than a “bushy” tree. PRIM algorithm (Friedman and Fisher 1999) produces a model of the same functional form, although it is motivated in terms of cutting out regions of the predictor space. PRIM considers the mean activity of the remaining box with each peeling-off procedure, which is similar to penalty entropy focusing on purity of positive daughter node. Furthermore, the bottom-

up pasting procedure in PRIM serves as the trimming procedure of OBSTree when dealing with binary predictors. The main difference between OBSTree and PRIM is that PRIM implements a sequential searching method, i.e., a sequence of peeling-off procedures, to identify the combination of descriptors for an active node rather than simultaneously considering multiple descriptors.

Although the advantages of OBSTree are obvious, some limitations should be noted. First, OBSTree is more computationally intensive than both standard RP and RP/SA. We coded all three algorithms in SAS/IML (SAS Institute, Inc. 1999) and found that the computational time of OBSTree is about five times longer than RP/SA and 10 times longer than standard RP when all tree models grow to the same depth. We are currently working to decrease the computational time. Second, a drawback that is shared by all statistical models for QSAR identification, not just OBSTree, is that care must be taken when interpreting the results. Because descriptors can be highly correlated, the same mechanism may be identified with different sets of descriptors and chromosomes. It is important to keep this in mind when comparing terminal nodes. Third, OBSTree requires specification of initial conditions, although this is not as onerous as it might first seem.

Any standard RP technique may be used to identify a list of “singly-important” descriptors. While this list does significantly increase computational efficiency when very large numbers of descriptors are available, we have found no need for its careful determination and OBSTree can still be effective even without it. The use of 5-fold CV to determine the final depth, d , can effectively control the over-fitting and complexity of the model instead relying on a totally user-specified parameter. Pre-specification of

K is not an issue as long as it is “reasonably” large because our trimming procedure can detect when a smaller K is needed. A very large K is not recommended because it will tremendously increase computational time. When it comes to tree-based models, minimum node size is a standard initial condition. Finally, initial conditions for the simulated annealing portion can follow standard recommendations; a reasonable approach is to start with the values we report in the paper.

Both standard RP and RP/SA can be regarded as special cases of the OBSTree algorithm. The concept of chromosome enables us to consider more meaningful splitting variables. A weighted sampling scheme combined with a modified simulated annealing procedure makes it possible to search for a global optimal splitting variable in a broadened searching space. A trimming procedure tends to make each split of the tree more interpretable. The penalty entropy criterion is particularly useful to classify compounds with ordinal response. Simulation and real data application results illustrate that OBSTree is a promising tool for identifying homogeneous activity classes with meaningful chemical and biological properties.

4.7 ACKNOWLEDGEMENTS

The authors are most grateful to: Jack Liu and Jun Feng for use of their software PowerMV for calculating atom pair descriptors; Golden Helix, Inc. for use of their software Partitionator for calculating recursive partitioning trees; Abbott Laboratories for use of their MAO data set; and to Atina Brooks, Jiajun Liu, and Haojun Ouyang for many lively discussions on these results.

REFERENCES

- Aarts, E., and Korst, J. (1989), *Simulated Annealing and Boltzmann Machine*, Chichester: Wiley.
- Blower, P., Fligner, M., Verducci J., and Bjoraker, J. (2002), "On Combining Recursive Partitioning and Simulated Annealing to Detect Groups of Biologically Active Compounds," *Journal of Chemical Information and Computer Sciences*, 42, 393-404.
- Brieman, L. (1998), "Arcing Classifiers (with discussion and a rejoinder by the author)," *Annals of Statistics*, 26, 801-849.
- Brieman, L. (2001), "Random Forest," *Machine Learning*, 45, 5-32.
- Breiman, L., and Cutler, A., R port by Andy Liaw and Matthew Wiener. (2005). randomForest: Breiman and Cutler's random forests for classification and regression. R package version 4.5-4.
URL <http://stat-www.berkeley.edu/users/breiman/RandomForests> (last accessed January 3, 2006)
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and Regression Trees*, New York: Chapman and Hall.
- Brown, R. D. and Martin, Y. C. (1996), "Use of Structure-Activity Data to Compare Structure-Based Clustering Methods and Descriptors For Use in Compound Selection," *Journal of Chemical Information and Computer Sciences*, 36, 572-584.

- Buja, A. and Lee, Y. (2001), "Data mining criteria for tree-based regression and classification," *Proceedings of KDD-2001*, 27-36.
- Carhart, R. E., Smith, D. H., and Venkataraghavan, R. (1985), "Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications," *Journal of Chemical Information and Computer Sciences*, 25, 64-73.
- Chen, X., Rusinko, A., Tropsha, A., and Young, S. S. (1999), "Automated Pharmacophore Identification for Large Chemical Data Sets," *Journal of Chemical Information and Computer Sciences*, 38, 1054-1062.
- Chib, S., and Greenberg, E. (1995), "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, 49, 327-335.
- Cho, S. J., Shen, C. F., and Hermsmeier, M. A. (2000), "Binary Formal Inference-Based Recursive Modeling Using Multiple Atom and Physicochemical Property Class Pair and Torsion Descriptors as Decision Criteria," *Journal of Chemical Information and Computer Sciences*, 40, 668-680.
- Golden Helix Inc. ChemTree Software. (2000), [http://www. goldenhelix.com/ ChemTreeManual/manual.htm](http://www.goldenhelix.com/ChemTreeManual/manual.htm) (last accessed January 3, 2006)
- Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning Data Mining, Inference, and Prediction*, New York: Springer.
- Hawkins, D. M. and Kass, G. V. (1982), "Automatic Interaction Detection," In *Topics in Applied Multivariate Analysis*, ed. D. M. Hawkins, Cambridge, U.K.: Cambridge University Press. pp. 269-302.

- Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983), "Optimization by Simulated Annealing," *Science*, 220, 67-78.
- Lambert, C. G. (2004), Partitionator Analysis Software, Golden Helix, Inc., <http://www.goldenhelix.com> (last accessed January 3, 2006).
- Liu, J., Feng, J., and Young, S. S. (2004), "PowerMV: A Software Environment for Statistical Analysis, Molecular Viewing, Descriptor Generation, and Similarity Search," <http://www.niss.org/PowerMV> (last accessed December 9, 2004).
- Maycock, A. L., Abeles, R. H., Salach, J. I., and Singer, T. P. (1976), "The Structure of the Covalent Adduct Formed by the Interaction of 3-(dimethylamino)-1-propyne And the Flavine of Mitochondrial Amine Oxidase," *Biochemistry*, 15, 114-125.
- McFarland, J. W. and Gans, D. J. (1986), "On the Significance of Clusters in the Graphical Display of Structure-Activity Data," *Journal of Medicinal Chemistry*, 29, 505-514.
- R Development Core Team. (2004), R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0,
URL <http://www.R-project.org>. (last accessed January 3, 2006)
- Ripley, B. (2005). tree: Classification and regression trees. R package version 1.0-19.
- Rusinko, A., Farmen, M. W., Lambert, C. G., Brown, P. L., and Young, S. S. (1999), "Analysis of a Large Structure/Biological Activity Data Set Using Recursive

Partitioning,” *Journal of Chemical Information and Computer Sciences*, 39, 1017-1026.

SAS Institute, Inc. (1999), *SAS/IML User’s Guide, Version 8*, Cary: SAS Institute, Inc.

— (2000), *Getting Started with Enterprise Miner Software, Release 4.1*, Cary: SAS Institute, Inc.

— (2004), *Getting Started With SAS Enterprise Miner 4.3*, Cary: SAS Institute, Inc.

Silverman, L., Campbell, R., and Broach, J. R. (1998), “New Assay Technologies for High Throughput Screening,” *Current Opinion in Chemical Biology*, 2, 397-403.

Sittampalam, G. S., Kahl, S. D. and Janzen, W. P. (1997), “High-throughput Screening: Advances in Assay Technologies,” *Current Opinion in Chemical Biology*, 1, 384-391.

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., and Feuston, B. P. (2003), “Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling,” *Journal of Chemical Information and Computer Sciences*, 43, 1947-1958 p.

Therneau, T. M., and Atkinson, B. R port by Brian Ripley <ripley@stats.ox.ac.uk>. (2005). rpart: Recursive Partitioning. R package version 3.1-22. S-PLUS 6.x original at <http://www.mayo.edu/hsr/Sfunc.html> (last accessed January 3, 2006)

Xue,L. and Bajorath, J. (2002), "Accurate partitioning of compounds belonging to diverse activity classes," *Journal of Chemical Information and Computer Sciences*, 42, 757-764.

Young, S. S., and Hawkins, D. M. (1995), "Analysis of a 2⁹ Full Factorial Chemical Library," *Journal of Medicinal Chemistry*, 38, 2784-2788.

Part II

Statistical Analysis of Compound Mixtures Using Nonlinear Models

Chapter 5

Concepts of Additivity in Chemical Mixture Studies

5.1 Introduction

The analysis of pharmacological and toxicological effects of chemical mixtures is of increasing interest. In clinical pharmacology, the exposure to a combination of drugs is involved in the therapies of diseases including AIDS and cancer. People are interested in assessment and characterization of potency due to using different drugs at different concentrations in combination in drug discovery and drug development. Any synergism in efficacy for a combination of drugs and antagonism in toxicity for the same drug combination may have potential therapeutic advantages since it implies that a better therapeutic effect with a reduced adverse effect can be achieved with the same or even smaller amounts of the drugs. An example illustrating this is given by Wolfe et al. (1978).

In general, synergism occurs when a combination of drugs produces a greater effect than the expected effects derived with additive (non-interactive) assumption. Alternatively, antagonism occurs when a mixture effect is lower than the level of response predicted under additivity. It is obvious that the definition of synergism and antagonism depends on the way in which the concept of additivity (no interaction) is defined. There is considerable literature about how to define additivity, resulting in different definitions of additivity. Different definitions of additivity may lead to the same mathematical formula from totally different perspectives. This causes considerable confusion and makes interpretation difficult. As far as we know, a thorough review of different concepts of additivity is currently unavailable and is of great demand. We review various types of mixture action mechanism classification in Section 5.2 since several popular concepts of additivity are highly dependent on the mixture action mechanisms. Section 5.3 clarifies the different natures of five widely-used concepts of additivity. The chemical conditions or assumptions to define concepts of additivity are considered and the mathematical derivations of different concepts are investigated. Section 5.4 discusses some criteria for evaluating a definition of additivity. We generalized a particular concept of additivity, the interaction index, to studies in which K categorical variables, as well as dose, are considered to model the response. Concluding remarks are given in Section 5.5.

5.2 Classifications of Mixture Action Mechanism

Although some additivity derivations take no account of mechanisms of mixture action (See Section 5.3), most concepts of additivity are defined with the classification

Table 5.1: Two-way Classification of Joint Action

	Similar	Dissimilar
Non-interactive	Simple similar	Independent
Interactive	Complex similar	Dependent

of mixture action mechanism in mind. It is worth considering these classifications together before we discuss additivity definitions.

1. Bliss (1939) is one of the forerunners to investigate the effect of a combination of two insecticides. He proposed three principle types of joint action: (a) similar joint action, where one component can be substituted at a constant proportion for the other since the components act at the same sites or the same system of receptors within the body, have similar modes of action and similar (parallel) dose-response curves, and behave like a dilution of each other; (b) independent joint action, where the components have different modes of action and act independently at different sites; (c) synergistic action, where the response of the mixture is greater than that predicted from isolated ingredients and antagonism is defined as the opposite of synergism. Substantial modifications were consequently made by numerous authors such as Finney (1942, 1952), Plackett and Hewlett (1952, 1967), and Hewlett and Plackett (1959). Most of these attempts were to develop more suitable mathematical models to analyze non-interactive (additive) and interactive mixture data based on Bliss's types of joint action. Hewlett and Plackett (1959) and Plackett and Hewlett (1961) used a two-way table (Table 5.1) to classify joint actions.

They defined similar or dissimilar action according to whether the sites of action of two drugs were the same or different, and interactive or non-interactive action according to one drug did or did not affect the action of the other. This two-way classification extends Bliss's method in dividing the last type of joint action, synergistic action, into two sub-categories based on "similarity". In a strict sense, the first two types of Bliss's joint action should both be regarded as additive (non-interactive) actions and the last type was the interaction case. We will suppose similar joint action to be synonymous with simple similar action and independent joint action to be synonymous with dissimilar independent action.

2. Loewe (1953) categorized mixtures into two groups by checking whether the components have "homodynamic" (i.e. similar) dose-response curves or "heterodynamic" (i.e. dissimilar) curves. This categorization is analogous to, though independent of, the above work of Bliss and Plackett and Hewlett. Notice that both synergistic/antagonistic (interactive) and additive (non-interactive) combinations can be included in either one of the two categories. Loewe (1953, 1957) developed an isobole method to define additivity (see Section 5.3.).
3. Chou and Talalay (1981,1984) focused on the ligands (molecules that bind to some receptors) obeying the law of mass action. One group is called mutually exclusive agents, where the same binding site is shared by the mixture agents, the occupation of one site by more than one agent simultaneously is not allowed, and dose-response curves are similar. The other one group is mutually non-exclusive agents, where the ligands bind at different sites and have no common

site of action and no similar dose-response curves. The first group fits well into the similar action type and the second group corresponds to the dissimilar action type. Ashford and Cobby (1979) presented a similar classification under the assumption that the action of a drug at any site results from the occupation of receptors and the occupation is governed by mass action law. Chow and Talalay (1984) developed two different forms of a “combination index” to define additivity for the two groups.(see Section 5.3.)

4. Finney (1952) found that the response of a mixture satisfying Bliss’s similar joint action could be calculated as the response of the weighted sum of the doses of each component. The response of a mixture satisfying independent joint action can be obtained by summing the responses of each component and subtracting the jointly occurring effects. Muska and Weber (1977) and Gessner (1988) accordingly termed two additive cases as dose addition (also termed concentration addition) and response addition (also termed effect addition or effect summation) with the purpose of avoiding the implications of Bliss’s classification that the modes of action are known to be similar or independent (Berenbaum 1989). THE USEPA (1999) cited the dose addition and the response addition as two concepts of no toxicologic interaction. Dose addition applies to the chemicals acting on similar biological systems and eliciting a common response, whereas response addition applies to the chemicals acting on different systems or produce independent effects. USEPA (1999) emphasized that the definitions (of no interaction) were operational and did not indicate specific toxicologic modes of action but general toxicologic modes of action.

5. Tallarida (2000, 2001) based his research on Bliss's classification. He, however, interpreted both of the first two types of Bliss's joint action as "independent action" (subsequently referred to as Tallarida's "independent action"). By this he meant that "each drug produces overtly similar effects (for example, each lowers blood pressure) such that all or part of one component may be substituted for the other in some proportion that is based on the dose-response relations of the two. (Tallarida 2000, pp.3)" He particularly excluded the case in which two overtly similar drugs act on the same receptor from his "independent action" class. It was argued that the effect of two drugs competing for a single common receptor depends on the bound concentration of the two. Consequently, the dose of one drug cannot be substituted for the other based solely on their individual dose-response curves in "dependent action". It seems, however, Tallarida's "independent action" also contains some simple similar cases where two drugs have similar curves and even act at the same biological system or site in the body. This classification is inconsistent with other classifications (e.g., Loewe 1953, Ashford and Cobby 1979; Chou and Talalay 1981,1984; Berenbaum 1989; USEPA 1999). Tallarida emphasized "independent action" because he derived his concept of additivity based on substitutable the property of the system (see Section 5.3).

5.3 Definitions of Additivity

5.3.1 Loewe Additivity and Bliss Independence

From Section 5.2, it can be seen that most mixture action mechanisms except Tallarida's independent action can be classified into two general types: similar and dissimilar action. Some general properties of the former are that the components act at the same site or similar biological system in the body, they have similar (parallel) dose-response curves and they may also have similar modes of action. In contrast, the properties of the dissimilar action mixtures include different action sites, different dose-response curves and different modes of action. Many authors (Bliss 1939; Finney 1971; Muska and Weber 1977; Drescher and Boedeker 1995; USEPA 1999) prefer using two different concepts of additivity for the two types of mixtures.

Loewe Additivity is designed for a mixture with similar action mechanism (Gessner 1995; USEPA 1999; Berenbaum 1989). The algebraic form is given as

$$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1,$$

where $ED_i(Y)$ is the dose of the i th agent alone that yields the response Y and x_i is the dose of the i th agent in the combination that yields the response Y , $i = 1, 2$. This type of additivity is called Loewe additivity (Greco et al. 1992 and Price et al. 2002), or dose/concentration addition (USEPA 1999; Drescher and Boedeker 1995). Synergism occurs, if the amounts of drugs generating the same fixed response are smaller than required under the additivity assumption. Consequently, the above

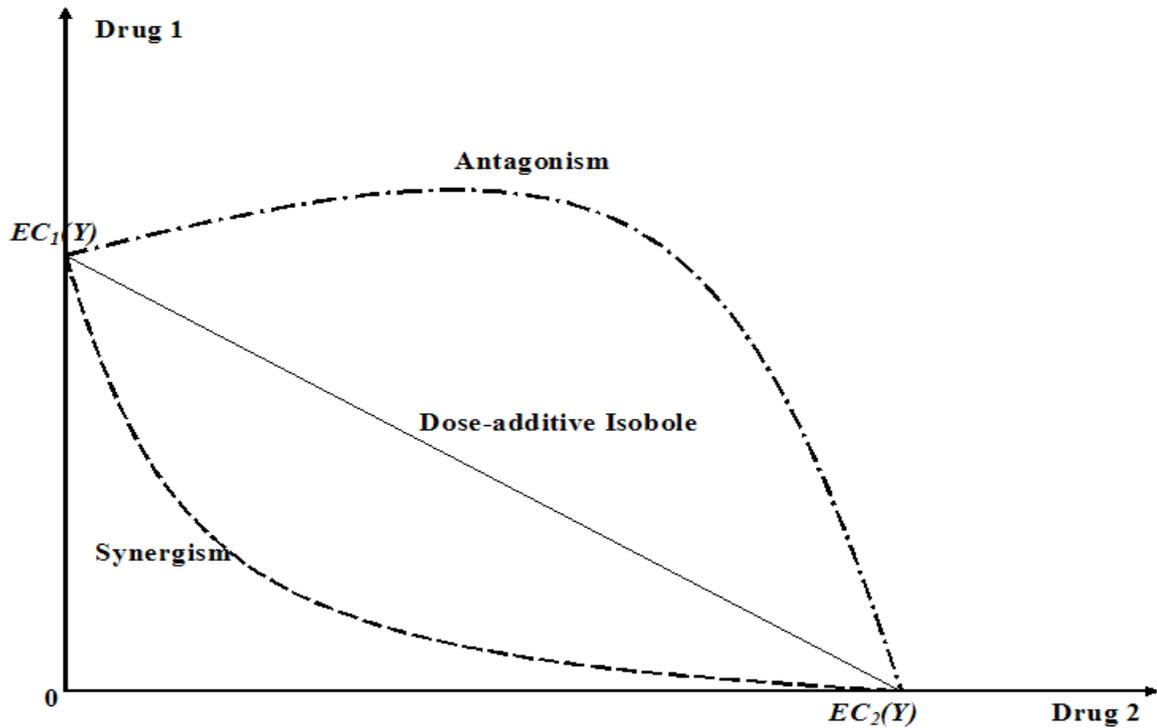


Figure 5.1: Isobologram for a combination of Drug 1 and Drug 2 with the dose-additive isobole corresponding to response Y

equality becomes an inequality as

$$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} < 1.$$

Similarly, a mixture yielding antagonism should satisfy the relationship

$$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} > 1.$$

Loewe (1953, 1957) introduced a method, which is now called the isobologram (or isobole) method, to present Loewe additivity (See Figure 5.1). The axes of isobologram coordinates represent doses of individual agents. For a binary mixture, all iso-effective curves or discrete dose pairs at which a fixed response is observed, are

termed isoboles. It is expected, if the additivity assumption is true, that the set of isoeffective dose pairs is a straight line with two intercepts defining the isoeffective single doses for two agents alone generating the fixed response. This line is called the zero-interaction isobole (Berenbaum 1989) or dose-additive isobole (USEPA 1999). In the isobologram graph, synergism or antagonism can be easily determined geometrically by identifying whether the dose pair point is below or above the no-interaction isobole line. Statistical methods can be used here to test if the point(s) really departs from the dose-additive isobole straight line.

A further simplification of Loewe additivity is possible when the relative potency that is estimated by the ratio of isoeffective doses of two agents is a constant over all effect levels. Let $f_1(x)$ and $f_2(x)$ be the dose-response functions for the two agents and ρ is the relative potency of the first agent to the second agent, we have

$$f_1(x) = f_2(\rho x)$$

for all x . Then a simplified form of Loewe additive response for dose pair (x_1, x_2) can be obtained as

$$Y = f(x_1, x_2) = f_1(x_1 + \rho x_2) = f_2(\rho^{-1}x_1 + x_2),$$

where $f(x, y)$ is the expected Loewe additive response of dose pair (x, y) . This simple model illustrates dose addition by converting one agent dose into an isoeffective dose of the other agent and using dose-response curve of a single agent to obtain response based on the assumption that the two dose-response curves are morphologically congruent (implied by a constant relative potency).

The dissimilar action mechanism can be regarded as two drugs behaving indepen-

dently of one another due to different modes of action or different action sites. The additivity for this type is described as a statistical independent event, with proportion of responders being drug effect. Therefore, the expected additive response of this mixture is given as,

$$Y = f_1(x_1) + f_2(x_2) - f_1(x_1)f_2(x_2) = 1 - (1 - f_1(x_1))(1 - f_2(x_2)).$$

This type of additivity is called Bliss independence (Greco et al., (1992) and Price et al. (2002)), or response addition (USEPA (1999)).

5.3.2 Combination Index

Chou and Talalay (1984) developed a method based on a combination index (CI) to determine additive/synergistic/antagonistic effects for mutually exclusive and mutually non-exclusive drugs. They assumed that the Hill equation can describe the dose-response curve for individual agents and that the action of agents obeys the law of mass action. A general form of the Hill equation is given as

$$f(x) = f_{min} + \frac{f_{max} - f_{min}}{1 + \left(\frac{EC(50\%f_{max})}{x}\right)^\rho},$$

where f_{min} is the minimum response, f_{max} is the maximum response, β is the Hill Slope parameter measuring the degree (order) of sigmoidicity of the curve, and sometimes indicating the minimum number of molecular binding sites for the agents. Considering a fractional effect with maximum effect 1 and minimum effect 0, the Hill equation can be written as

$$\frac{f(x)}{1 - f(x)} = \left(\frac{x}{ED(50\%)}\right)^\beta,$$

or as a common linear form,

$$\log \left(\frac{f(x)}{1 - f(x)} \right) = -\beta \log ED(50\%) + \beta \log(x).$$

The mass-action mechanism of mutually exclusive agents with the same order (the same β) dose-response curves yields an additive relation as,

$$\left\{ \frac{f(x_1, x_2)}{1 - f(x_1, x_2)} \right\}^{\frac{1}{\beta}} = \left\{ \frac{f_1(x_1)}{1 - f_1(x_1)} \right\}^{\frac{1}{\beta}} + \left\{ \frac{f_2(x_2)}{1 - f_2(x_2)} \right\}^{\frac{1}{\beta}}.$$

Suppose fractional response $f(x_1, x_2) = Y$ is produced at dose pair (x_1, x_2) . Therefore $f(x_1, x_2) = f_1(ED_1(Y)) = Y$ holds. Some simple algebraic manipulation of the additive relation with Hill equation gives

$$\left\{ \frac{f(x_1, x_2)}{1 - f(x_1, x_2)} \right\}^{\frac{1}{\beta}} = \frac{x_1}{ED_1(50\%)} + \frac{x_2}{ED_2(50\%)} = \left\{ \frac{f_1(ED_1(Y))}{1 - f_1(ED_1(Y))} \right\}^{\frac{1}{\beta}} = \frac{ED_1(Y)}{ED_1(50\%)}.$$

Similarly, we have

$$\frac{x_1}{ED_1(50\%)} + \frac{x_2}{ED_2(50\%)} = \frac{ED_2(Y)}{ED_2(50\%)}$$

since $f(x_1, x_2) = f_2(ED_2(Y)) = Y$. The above two equations implementing the concept of additivity for mutually exclusive case directly define a decision approach to determining whether the observed response Y is additive or not. The additive form of the decision rule is given as,

$$CI(\text{additivity}) = \frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1$$

Synergism occurs when $CI < 1$ and antagonism is implied when $CI > 1$. This type of additivity has the same formula as Loewe additivity except it follows from the mass-action law.

Chou and Talalay (1984) also argued that the above CI cannot be applied to mutually non-exclusive agents. They stated a different additive relationship for a mixture of mutually non-exclusive drugs obeying the same order (same β) conditions as,

$$\left\{ \frac{f(x_1, x_2)}{1 - f(x_1, x_2)} \right\}^{\frac{1}{\beta}} = \left\{ \frac{f_1(x_1)}{1 - f_1(x_1)} \right\}^{\frac{1}{\beta}} + \left\{ \frac{f_2(x_2)}{1 - f_2(x_2)} \right\}^{\frac{1}{\beta}} + \left\{ \frac{f_1(x_1)f_2(x_2)}{(1 - f_1(x_1))(1 - f_2(x_2))} \right\}^{\frac{1}{\beta}}.$$

It is obvious that when β is 1, i.e, in the first-order system, the above relation is mathematically equivalent to Bliss Independence. Without giving further mathematical or pharmacological explanations, the author extended this additive relation by analogy to define a new additive CI for mutually non-exclusive agents as follows,

$$\text{CI}(\text{additivity}) = \frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} + \frac{x_1x_2}{ED_1(Y)ED_2(Y)} = 1.$$

Again, $\text{CI} < 1$ implies synergism and $\text{CI} > 1$ implies antagonism.

5.3.3 Tallarida's Additivity

Tallarida (2000, 2001) focused his studies on mixtures obeying Tallarida's "independent action". His fundamental assumption is that all or part of one agent of the mixture can be substituted for the other in some proportion based solely on the individual dose-response relationships, i.e., the presence of one drug acts like the addition of a more concentrated or a diluted form of the other one. He argued that the substitution cannot be implemented if the agents of a mixture behave "dependently" (see Section 5.2).

With the substitutable property Tallarida's mathematical derivation of the expected additive response for a binary mixture of (x_1, x_2) depends on the relative

potency. If the relative potency of Drug 1 to Drug 2, β , is assumed to be the same over all response levels, the same formula as the simplified Loewe additivity (see Section 5.2.) can be derived, i.e., the expected additive effect of dose pair (x_1, x_2) , Y_{add} , is equal to the response generated by Drug 1 alone at dose $(x_1 + \beta x_2)$ or by Drug 2 alone at dose $(\beta^{-1}x_1 + x_2)$. In other words, the observed response of mixture, Y , is additive if

$$x_1 + \beta x_2 = ED_1(Y)$$

or

$$\beta^{-1}x_1 + x_2 = ED_2(Y).$$

Since β is a constant over all response levels, i.e., $\beta = ED_1(Y)/ED_2(Y)$, these two equations can be rearranged into a more familiar form

$$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1.$$

This should not be surprising because the simplified Loewe additivity is just a special case of Loewe additivity, and Tallarida's method has the same mathematical form as the simplified Loewe additivity.

When the relative potency varies with different response levels, the substitution amount of one drug for the other is not well defined in Tallarida's additivity. At least two possible choices are proposed in different papers, different results may be achieved with the same data. This is exemplified with an analysis of a mixture of a full and a partial agonist given by Grabovsky and Tallarida (2004).

Suppose Drug 1 and Drug 2 follow the Hill dose-response curves with the Hill

Slope parameter, ρ , being 1

$$f_1(x) = \frac{f_{1,max}}{1 + \left(\frac{ED_1(50\%f_{1,max})}{x}\right)} = \frac{x f_{1,max}}{x + ED_1(50\%f_{1,max})},$$

and

$$f_2(x) = \frac{f_{2,max}}{1 + \left(\frac{ED_2(50\%f_{2,max})}{x}\right)} = \frac{x f_{2,max}}{x + ED_2(50\%f_{2,max})},$$

respectively. When $f_{1,max} > f_{2,max}$, i.e., Drug 1 is a full agonist and Drug 2 is a partial agonist, the relative potency of Drug 1 to Drug 2, β , varies at different response levels. The substitution strategy is still implemented to define the concept of additivity for a dose pair (x_1, x_2) .

Grabovsky and Tallarida (2004) stated that Drug 2 at dose x_2 can be substituted with Drug 1 at dose βx_2 , where β , the relative potency of Drug 1 to Drug 2, satisfies $f_1(\beta x_2) = f_2(x_2)$. Please note that the relative potency of Drug 1 to Drug 2 is evaluated at response level of $f_2(x_2)$. The expected additive response Y_{add} of the dose pair (x_1, x_2) is given as

$$Y_{add} = f_1(x_1 + \beta x_2).$$

In other words, an observed response Y is additive if

$$x_1 + \beta x_2 = ED_1(Y).$$

Given the fact that β in substitution is evaluated at response level of $f_2(x_2)$, β can be calculated as

$$\beta = \frac{ED_1(50\%f_{1,max})}{x_2(f_{1,max} - f_{2,max}) + f_{1,max}ED_2(50\%f_{2,max})}$$

This β is not equal to the relative potency $\beta = ED_1(Y)/ED_2(Y)$ evaluated at the observed response level Y . Consequently, the additive relation $x_1 + \beta x_2 = ED_1(Y)$

derived above cannot be simplified to the form of Loewe additivity. Therefore, the additive isobole defined with this method is curved rather than a straight line and has a different mathematical form from Loewe additivity.

Tallarida (2000), however, implemented the Loewe additivity form to define the additive effect of the dose pair (x_1, x_2) even when relative potency, β , varies at different response levels. In other words, an observed response Y is additive if

$$\frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1.$$

It can be shown that this definition of additivity is mathematically equivalent to the statement that Y is additive if

$$x_1 + \beta x_2 = ED_1(Y),$$

where $\beta = ED_1(Y)/ED_2(Y)$. This mathematical expression also implies that the additive effect Y for a binary mixture of (x_1, x_2) can be derived by substituting Drug 2 at dose x_2 with Drug 1 at dose x_2 multiplied by β , the relative potency of Drug 1 to Drug 2 at response level of Y . Therefore, β in substitution is evaluated at the response level of Y instead of the response level of $f_2(x_2)$ as suggested in Grabovsky and Tallarida's paper (2004). Consequently, two conflicting concepts of additivity can be derived from a mixture study with variable relative potency under Tallarida's "independent interaction" assumption. No further clarification can be found to help people choose between the two different concepts; therefore, we do not recommend using Tallarida's additivity for mixture studies.

5.3.4 Interaction Index

The landmark work of Berenbaum (1989) presented a new concept of additivity called interaction index (II), which was derived from a totally different perspective while still possessing a traditional form.

Berenbaum argued that the concept of additivity should not depend on any knowledge of joint action mechanisms of the mixture. In other words, it is inappropriate to equate additivity or no interaction with the expected additive response under the assumption of a particular type of joint action mechanism. He stated that a fundamental difficulty of mechanistic methods was that the additivity and any consequent synergism or antagonism conclusion entirely depend on the current state of knowledge, which was highly accidental and temporal. Conflicting conclusions can be made on the same dose-response data set under different mechanism assumptions, which will cause trouble when the mixture action mechanism is ambiguous. In addition, the mechanistic additive model results in a situation where we can have more precise expectation on responses of a specific mixture as our understanding of mixture action mechanisms increases, and consequently there is less chance that we could declare any interaction if we keep progressively modifying our concepts of additivity with the improved knowledge of mechanisms.

Berenbaum suggested an empirical model, II, free of mechanism of mixture action for additivity definition. II was constructed solely on observed dose-response information of the agents and mixture, thus, was generally applicable for most cases. The

additive form of II for a mixture of n components is given as

$$\text{II}(\text{additivity}) = \sum_{i=1}^n \frac{x_i}{ED_i(Y)} = 1,$$

where x_i is the dose of the i th component in combination that can generate response Y and ED_i is defined as before. It coincidentally has the same form as isobole method of Loewe additivity for the binary case. Berenbaum proved the validity of the formula by constructing a sham combination.

The proof is based on an important argument that it is a self-evident and indisputable fact that the sham combination consisting of only one agent and its dilutions is additive. Suppose a binary mixture of (x_1, x_2) produces the corresponding response Y , the isoeffective doses of the individual agents, $ED_1(Y)$ and $ED_2(Y)$, can be obtained without knowing any mixture action mechanism. Now dilute the Drug 1 $(ED_2(Y)/ED_1(Y))$ -fold and call it Drug 3. It is obvious that the dose of Drug 3 alone generating response Y is $ED_2(Y)$, which is exactly equal to that of Drug 2. A sham combination containing x_1 of Drug 1 and x_2 of Drug 3 is constructed to share the same dose information as the real mixture. Y is an additive response if the response of the sham combination is equal to Y . This yields the following formula that can define the additivity,

$$Y = f(x_1, x_2) = f_1 \left(x_1 + x_2 \frac{ED_1(Y)}{ED_2(Y)} \right) = f_1(ED_1(Y)).$$

Under the assumption that the individual dose-response curves are monotonically increasing or decreasing the mathematical form of II can be derived as

$$\text{II}(\text{additivity}) = \frac{x_1}{ED_1(Y)} + \frac{x_2}{ED_2(Y)} = 1.$$

Under II the sham combination and the additive mixture share some important quantitative properties that include mixture dose, isoeffective single dose, and response. The proof can be readily generalized to a mixture of M components. A mixture shows synergism if $II < 1$ and shows antagonism if $II > 1$. Notice that the proof of II assumes no action mechanism of the mixture though it has the same mathematical form as Loewe additivity. The isobologram is also a good way to present II for a binary mixture.

5.4 How to Develop a Good Concept of Additivity?

It can be seen that there is no consensus answer to the question: “Which additivity concept is the best?”. Consequently, people cannot have a clear understanding of the terms such as “synergism” and “antagonism” during scientific communications. The worst part is that the regulatory agencies have to define their own additivity concepts for different cases to ensure that the assessors share the same meaning of these words when they review documents and make decisions. Some concepts of additivity, however, are hard to justify because of the lack of or the uncertainty in the desired information. For example, dose addition and response addition are two additivity concepts used by USEPA (1999) since “the underlying concepts are straightforward and in common use (USEPA 1999, pp.63).” In addition, USEPA cited that the primary criterion for selecting between two definitions of additivity is the similarity or independence among the chemicals in the mixture. It is, however, well known (Drescher

and Boedeker 1995) that such knowledge (similarity and independence) is rare or ambiguous for most chemicals in toxicology and ecotoxicology. It was also shown by a number of authors (e.g., Berenbaum 1989; Christensen and Chen 1985; Drescher and Boedeker 1995) that a synergistic effect assessed on the basis of dose addition, might at the same time be declared as an antagonistic effect with respect to response addition and vice versa. Therefore, even some popular definitions of additivity may ask for information that is difficult to collect or validate.

Before discussing how to develop a good concept of additivity, we should note that it would be appropriate to view the concept of additivity solely as a quantitative rule that can allow us to mathematically interpret results with some unequivocal terms such as “synergism” or “antagonism”. This concept should not contain many pharmacological, physiological or toxicologic explanations. In other words, any observed data of a mixture effect that is declared to be additive with a mathematical/statistical formulation of additivity only tells us that the mixture response is quantitatively being an additive effect. This cannot serve as a scientific proof that the agents of the mixture are chemically non-interactive. Any such conclusion should only be found from chemical and biological experiments with scientifically sound arguments.

We agree with Berenbaum (1989) that a good concept of additivity should be defined to be generally applicable for most cases, which implies that it cannot require thoroughly scientific understanding of the mixtures such as joint action mechanisms. We also agree that a concept of additivity should be based on quantitatively verifiable properties. For instance, II (Berenbaum 1989) was proposed as a generally applicable method that works by ignoring any mixture action mechanism information. The sham

combination is constructed with the same quantitative properties as the mixture of interest in terms of individual doses and isoeffective single doses. Therefore, the effect of the sham combination quantitatively defines an additive effect of the mixture.

In addition, the concept of additivity should be mathematically comparable to a chemically additive system (e.g. a sham combination in II) with respect to the quantitative properties incorporated in developing the concept. The selection of such a system, however, varies and highly depends on the objectives of the study. Berenbaum (1989) stated that the sham combination consisting of only one agent is indisputably chemically additive such that II developed on this assumption should be generally applicable. This statement is true for most dose-response studies and it should be recognized that incorporating a sham combination in the development of additivity is critical. However, some new studies focusing on so-called “self synergy” or “site-site interaction” (Raffa et al. 2000; Tallarida 2000, 2001) may challenge the correctness of the sham combination proposed in II. These studies explored the utilization of different sites in the administration of the same compound and found potential synergistic/antagonistic effect among different sites. In other words, one-drug system could be nonadditive. We use the following example to explain how the development of a chemically additive system should depend on the objectives of the experiments and how we could generalize II to a more complicated case.

Raffa et al. (1999, 2000) conducted several experiments to study the pain pathways affected by acetaminophen. The drug was administered in the brain (intracerebroventricular, or i.c.v.) and the spinal cord (intrathecal, or i.th.) separately first. It was found that i.c.v. injection on its own could not generate an observ-

able response, whereas i.th. administration alone produced dose-dependent response. Analysis (Tallarida 2000) of response data generated by simultaneous injection at both sites declared a significant synergism and led researchers to study if there is any release of second substance induced by acetaminophen in the brain interacting with the acetaminophen in the spinal cord. Tallarida used a probit model to describe the dose-response relationship. In addition, an additive response was intuitively set as the response generated by only injecting the drug of the same dose at the i.th. site alone. A general mathematical form for the additivity concept in such “site-site interaction” studies, however, was not given by Tallarida.

From a statistical point of view, the injection site is an important explanatory variable that can affect response in the studies of site-site interaction. For any single compound, the dose-response model should incorporate site as another covariate and the interaction between dose and injection site should be tested for significance. Therefore, a general probit dose-response model for such studies would be

$$\Phi^{-1}(Y) = \beta_0 + \beta_1 x + \beta_2 s + \beta_3 xs,$$

where s is a categorial variable for site, x is the dose, Y is the quantal response, and Φ^{-1} is the inverse CDF function of the standard Normal distribution function. Based on the model, we could have two different chemically additive systems in the “self-synergy” studies focusing on two sites.

Suppose a combination therapy containing Drug 1 at dose x_1 injected at site 1 and Drug 1 at dose x_2 injected at site 2 can generate a response of Y , Drug 1 of $ED(Y, s = 0)$ injected at site 1 alone generates Y , and Drug 1 of $ED(Y, s = 1)$

injected at site 2 individually generates Y , where $s = 0$ denotes site 1 and $s = 1$ denotes site 2. The general probit model above yields

$$ED(Y, s = 0) = \frac{\Phi^{-1}(Y) - \beta_0}{\beta_1}$$

and

$$ED(Y, s = 1) = \frac{\Phi^{-1}(Y) - \beta_0 - \beta_2}{\beta_1 + \beta_3}.$$

One additive system would be simultaneous injection of x_1 of Drug 1 at site 1 and x_2 of the same drug diluted $(ED(Y, s = 1)/ED(Y, s = 0))$ -fold at site 1. Notice that the isoeffective single dose of diluted Drug 1 injected at site 1 in this system, i.e., x_2 , is the same as dose of Drug 1 injected at site 2. Thus, this additive system is quantitatively equal to the real combination therapy with respect to agent doses and isoeffective single doses. Similarly, a different additive system can be constructed as a simultaneous injection of Drug 1 of x_2 at site 2 and Drug 1 diluted $(ED(Y, s = 0)/ED(Y, s = 1))$ -fold of x_1 at site 2. Therefore, there are two additive systems sharing the same quantitative properties of dose as the real combination. Consequently, Y is additive if

$$Y = \Phi\{\beta_0 + \beta_1 ED(Y, s = 0)\} = \Phi\left\{\beta_0 + \beta_1 \left(x_1 + x_2 \frac{ED(Y, s = 0)}{ED(Y, s = 1)}\right)\right\}$$

using the first additive system, or

$$Y = \Phi\{\beta_0 + \beta_2 + (\beta_1 + \beta_3) ED(Y, s = 1)\} = \Phi\left\{\beta_0 + \beta_2 + (\beta_1 + \beta_3) \left(x_2 + x_1 \frac{ED(Y, s = 1)}{ED(Y, s = 0)}\right)\right\}$$

using the second additive system. Since $\Phi(\cdot)$ is a monotone function, both conditions can be simplified as

$$\frac{x_1}{ED(Y, s = 0)} + \frac{x_2}{ED(Y, s = 1)} = 1.$$

This familiar form defines the concept of additivity in “self-synergy” studies, which can readily serve as a general rule free of joint action mechanisms. Notice that the concept is independent of the link function (i.e., probit function in our example) as long as the function is monotone in dose at a fixed injection site. Tallarida’s example is just a special case where no observed response can be produced at one injection site (i.c.v.) so that the isoeffective single dose of Drug 1 injected at this site is mathematical infinity for every response level. Plugging infinity in the concept above results in a reduced version with only one component in the formula. Consequently, the concept of additivity says that the response under two-site injection therapy is additive in this “site-site interaction” study if it equals the response generated by injecting the drug at i.th. site alone with the same dose. This definition is consistent with Tallarida’s intuitive selection.

It becomes complicated to define the concept of additivity when different drugs are administered at different sites. We consider a binary mixture injected at two different sites as an illustrating example. Suppose we have Drug 1 at dose x_{10} and Drug 2 at dose x_{20} injected at site 1, and Drug 1 at dose x_{11} and Drug 2 at dose x_{21} simultaneously injected at site 2. The general forms of individual dose-response functions are

$$Y = f_1(x, s),$$

and

$$Y = f_2(x, s).$$

If the observed response is Y , there are four isoeffective single doses in this study, $ED_1(Y, s = 0), ED_1(Y, s = 1), ED_2(Y, s = 0), ED_2(Y, s = 1)$, where $ED_1(Y, s = 0)$

denotes the dose of Drug 1 that, when administrated alone at site 1, produces the same effect Y and similar notation rules are applied to other terms. Similarly, we can construct four chemically additive systems as before. For example, one possible system contains Drug 1 of x_{10} , Drug 1 diluted $(ED_2(Y, s = 0)/ED_1(Y, s = 0))$ -fold of x_{20} , Drug 1 diluted $(ED_1(Y, s = 1)/ED_1(Y, s = 0))$ -fold of x_{11} and Drug 1 diluted $(ED_2(Y, s = 1)/ED_1(Y, s = 0))$ -fold of x_{21} simultaneously injected at site 1. With similar arguments, a concept of additivity is consequently given as

$$\frac{x_{10}}{ED_1(Y, s = 0)} + \frac{x_{20}}{ED_2(Y, s = 0)} + \frac{x_{11}}{ED_1(Y, s = 1)} + \frac{x_{21}}{ED_2(Y, s = 1)} = 1.$$

Please note that all four chemically additive systems will yield the same concept.

Theoretically, this approach to developing additivity concept can be generalized to a mixture of M chemicals with k categorical factors plus dose considered in modeling the response. A general form of the concept of additivity is

$$\sum_{i=1}^M \left(\sum_{z_k=1}^{Z_k} \cdots \sum_{z_1=1}^{Z_1} \frac{x_{iz_1 \cdots z_k}}{ED_i(Y, z_1 = z_1, \cdots, z_k = z_k)} \right) = 1,$$

where where z_1, \dots, z_k are k categorical variables, $x_{iz_1 \cdots z_k}$ is the dose of chemical i in the mixture system with a fixed set of $z_1 = z_1, \dots, z_k = z_k$, and $ED_i(Y, z_1 = z_1, \dots, z_k = z_k)$ is the dose of chemical i alone with a fixed set of $z_1 = z_1, \dots, z_k = z_k$ that can yield response Y . The monotonicity assumption of dose-response relation is necessary and it should not be a very restrictive constraint in practice, especially in drug development.

5.5 Summary

In summary, the development of a good definition of additivity depends on the objectives of studies. The factors that are examined in studies and have potentially significant effect on response should be quantitatively modeled and incorporated into additivity concept. We support Berenbaum on the ideas underlying II development; a good concept of additivity should be defined to be generally applicable for most cases and not depend on the joint action mechanisms. The chemically additive system originally proposed in II as a sham combination is generalized to accommodate more complex studies (e.g., “site-site interaction” studies) such that the concepts of additivity can be derived for such studies.

In addition, any additivity concept can solely serve as a quantitative comparison rule rather than a scientific explanation. Although the chemically additive systems are implemented in defining the concept of additivity, any additive or nonadditive conclusion drawn from our approach only states that the response of mixture is quantitatively equal or unequal to that of a chemically additive system. This quantitative comparison, however, can suggest and lead scientists to propose new hypotheses, design new experiments, and consequently find the new joint action mechanisms.

Chapter 6

Statistical Nonlinear Models and Hypothesis Tests in Chemical Mixture Studies

6.1 Introduction

Understanding dose-response relationships, especially finding synergism or antagonism, is of great practical importance in chemical mixture studies. In clinical pharmacology this helps us determine the optimum combination of doses for administration. In environmental risk assessment, understanding interactions among individual toxicants or environmental contaminants can facilitate establishing acceptable environmental exposure levels and consequently reducing human illness from exposures to multiple chemicals.

A nonlinear dose-response statistical model is proposed for binary mixture studies. The nonlinear form is developed using a well-known additivity model with doses

multiplied by a newly proposed factor under the assumption that each individual chemical has a monotonically increasing dose-response curve. The observations for model fitting should contain dose-response data of each individual component and of some combination points. The model is able to flexibly describe dose-dependent synergistic or antagonistic effects. Statistical test procedures are developed to test and identify interactions based on the model and the concept of additivity developed in Chapter 5. The model and test procedure can be readily generalized to study a mixture of M components.

The remainder of Chapter 6 is organized as follows. Section 6.2 reviews existing nonlinear models, while Sections 6.3 and 6.4 contain our proposals of nonlinear models for binary and M -component mixtures. Tests of interactive effects are also developed in Section 6.3 and 6.4. Section 6.5 applies the models to four real data sets. The hypothesis tests for examining interactions are conducted for each data set. Section 6.6 conducts a simulation study to check the relationship between test power and magnitude of interaction and the relationship between test power and sample size. Section 6.7 briefly reviews different experimental designs in compound mixture studies and shortly discusses the impact of study design on the analysis. Section 6.8 gives concluding remarks.

6.2 Review of Nonlinear Models for Mixtures

6.2.1 Early Works on Modeling Interactions within Mixtures

Early quantitative work in pharmacology focused on the law of mass action applied to two competing agents attempting to bind to the same receptor. This law of mass action states that “the velocity of a chemical reaction is proportional to the product of the concentration (or mass) of the reactants” (Motulsky and Christopoulos 2004, p. 187). To illustrate the basics of the law of mass action without competition, suppose a single agent A and the receptor R interact reversibly to form a complex AR , which can be described as a bimolecular reaction



According to the law of mass action, the rates of association and dissociation are given as

$$\text{Number of binding events per unit of time} = k_{on}[A][R],$$

and

$$\text{Number of dissociation events per unit of time} = k_{off}[AR],$$

where $[A]$, $[R]$, and $[AR]$ represent the concentrations of A , R , and AR , respectively, and k_{on} and k_{off} denote the association rate constant and the dissociation rate constant. At equilibrium, we have

$$k_{on}[A][R] = k_{off}[AR].$$

The equilibrium dissociation constant of A , C_{dA} , is consequently defined as

$$C_{dA} = \frac{k_{off}}{k_{on}} = \frac{[A][R]}{[AR]}$$

and Motulsky and Christopoulos (2004, p. 189) showed that the fractional occupancy by A , namely the fraction of all receptors that are bound to agent A , can be defined as

$$f_A = \frac{[A]}{[A] + C_{dA}}.$$

Gaddum (1957) considered a situation of two competing agents. The first agent A was an agonist that caused a response by binding to the receptor, and the other agent B was an antagonist that did not produce a response itself but competitively bound to the receptor. He derived the fractional occupancy by agonist A as

$$f_A = \frac{[A]}{[A] + C_{dA} \left(1 + \frac{[B]}{C_{dB}}\right)}, \quad (6.1)$$

where C_{dA} and C_{dB} are the equilibrium dissociation constant of A and that of B , respectively. The most useful point conveyed by this equation is that the presence of a competitor B multiplicatively increases the equilibrium dissociation constant of agent A by a factor equal to $1 + [B]/C_{dB}$. This in turn causes a decrease in the fractional occupancy of agonist A and hence reduces response levels caused by agent A .

Assuming the relationship between occupancy of A and final response does not change in presence of a competitor, fractional occupancy by A uniquely defines the response level caused by A so that dose-response relationships change as fractional occupancy changes (Motulsky and Christopoulos 2004, p.276). On the other hand, fractional occupancy by A may not be uniquely defined as a function of the concentration $[A]$. When competition is present according to (6.1), at least two different combinations (a, b) for concentrations of agents A and B result in equivalent fractional

occupancy by A :

$$\begin{aligned}
 \text{(i)} \quad & ([A], [0]) \Rightarrow f_A = \frac{[A]}{[A] + C_{dA}} & (6.2) \\
 \text{(ii)} \quad & \left([A] \left(1 + \frac{[B]}{C_{dB}} \right), [B] \right) \Rightarrow f_A = \frac{[A] \left(1 + \frac{[B]}{C_{dB}} \right)}{[A] \left(1 + \frac{[B]}{C_{dB}} \right) + C_{dA} \left(1 + \frac{[B]}{C_{dB}} \right)} = \frac{[A]}{[A] + C_{dA}}
 \end{aligned}$$

Therefore, the dose-response curve of A in the presence of antagonist B formulaically equals the original dose-response curve of A by itself with every concentration of A divided by $1 + [B]/C_{dB}$. If response is plotted as a function of log-dose of A , the effect of adding antagonist B is to slide the original log-dose-response curve of A by itself a distance of $\log(1 + [B]/C_{dB})$ units to the right.

Several modifications of the factor of $1 + [B]/C_{dB}$ have been proposed to consider more general cases where two agents are not competitive or the receptors with equal affinity to the agonist have different affinities to the antagonist. Schild (1957) proposed a factor of $1 + [B]^S/C_B$, where C_B is not the equilibrium dissociation constant of agent B if S is not equal to one. Lazareno and Birdsall (2003) developed a factor of $1 + ([B]/C_B)^S$, which reduces the correlation between S and C_B in Schild's factor. All three factors share two properties: 1) they all affect fractional occupancy by agonist A through a multiplier of C_{dA} , and 2) all factors become one if antagonist B is absent in the system. These properties imply that presence of antagonist only shifts the log-dose-response curve of agonist A to the right by the log of a particular factor without changing its shape (i.e., maximal response, minimal response, and slope). The extent of the shift varies in different cases depending on whether the antagonist is a competitive inhibitor or a noncompetitive inhibitor.

6.2.2 Recent Models

In order to describe the interaction between two active components, Brunden et al. (1983) and Meadows, Gennings, Carter, and Bae (2002) implemented a generalized linear model as

$$\mu = q(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2),$$

where x_1 and x_2 are the doses (or concentrations) of the two components, μ is the expected response and $q^{-1}(\cdot)$ is a link function in Generalized Linear Models (GLM's). The term $\beta_3 x_1 x_2$ captures the interaction effect. A hypothesis test of $\beta_3 = 0$ can be used to determine if there is a consistent pattern of interaction over the entire dose combination range. In other words, $\beta = 0$ implies no interaction, $\beta > 0$ implies synergism for all dose combinations if $q(\cdot)$ is an increasing function, and $\beta < 0$ implies antagonism for all dose combinations.

Rider and LeBlanc (2005) proposed different models based on activity mechanisms. If two chemicals are in the same “cassette,” i.e., acting in a similar way, concentration addition is assumed and a model is given as

$$\mu = \frac{1}{1 + \frac{1}{\left(\frac{x_1}{EC50_1} + \frac{x_2}{EC50_2}\right)^{\frac{\rho_1 + \rho_2}{2}}}},$$

where μ , x_1 , x_2 , $EC50_1$, and $EC50_2$ are defined similarly as before, while ρ_1 and ρ_2 are estimated slopes from fitting the Hill model to individual dose-response curves. When two chemicals are in different “cassettes,” i.e., acting in different ways, response

addition is assumed and a model is given as

$$\mu = 1 - \prod_{i=1}^2 \left(1 - \frac{1}{1 + \frac{1}{\left(\frac{x_i}{EC50_i}\right)^{\rho_i}}} \right).$$

If there is more than one chemical in each of N cassettes, an integrated additivity model is

$$\mu = 1 - \prod_{i=1}^N \left(1 - \frac{1}{1 + \frac{1}{\left(\sum_{j=1}^{J_i} \frac{x_{ji}}{EC50_{ji}}\right)^{\rho_{ji}}}} \right),$$

where N is the total number of cassettes in the mixture, J_i is the total number of compounds in the i^{th} cassette, subscript “ ji ” refers to the j^{th} compound in the i^{th} cassette, and μ , x 's, $EC50$'s and ρ 's are defined similarly as before.

Rider and LeBlanc (2005) also proposed an interaction model under the assumptions that “cassette” information is available and the ability of one chemical, A , in the mixture to modify the effective concentrations of other chemicals has been experimentally determined. A Z -function (referred to as K -function by Rider and LeBlanc) is introduced to the integrated additivity model such that their integrated-additivity-based interaction model for a mixture is

$$\mu = 1 - \prod_{i=1}^N \left(1 - \frac{1}{1 + \frac{1}{\left(\sum_{j=1}^{J_i} \frac{Z_{Aji} x_{ji}}{EC50_{ji}}\right)^{\rho_{ji}}}} \right),$$

where the interaction effect of A on the j^{th} chemical in the i^{th} cassette is captured by

$$Z_{Aji} = [1 + \exp(-\lambda_{0Aji} - \lambda_{1Aji} \log x_A)]^{-1},$$

which can be rewritten as

$$Z_{Aji} = \left(1 + \left(\frac{x_A}{L_{Aji}} \right)^{s_{Aji}} \right)^{-1},$$

where $L_{Aji} = \exp(-\lambda_{0Aji}/\lambda_{1Aji})$ and $s_{Aji} = \lambda_{1Aji}$.

This Z -function is modeled as a function of concentration x_A of A and its inverse has the same mathematical form as the factor $1 + ([B]/C_B)^s$ proposed by Lazareno and Birdsall (2003). In other words, Rider and LeBlanc's interaction model is consistent with the discussion surrounding (6.2) where presence of chemical A alters the dose-response curve of the j^{th} chemical in the i^{th} cassette by reducing its concentration x_{ij} through division by the Lazareno and Birdsall factor $1 + (x_A/L_{Aji})^{s_{Aji}}$.

The Rider and LeBlanc (2005) interaction model is scientifically sound and well justified, but it assumes knowledge that is typically not available. Only in very special cases will cassette placement (concentration versus response addition) be known and it is even more rare for Z -functions to be available since these require large data sets. In addition, it is unclear how one can determine that the chemical A modifies other chemicals prior to collecting the data. On the other hand, the Brunden et al. (1983) generalized linear model for capturing interaction is simple and does not require knowledge of activity mechanisms or details of how interaction occurs. Its simplicity is also its downfall, however, in that the model is not able to capture complicated dependencies between dose level and type or strength of interaction.

Several authors argue, however, that interactions may occur only at a subset of combination points or be restricted to a specific dose combination area (Tallarida, Porreca, and Cowan 1989; Dawson, Carter, and Gennings 1995; Hamm, Carter, and Gennings 2005). This situation is subsequently referred to as dose-dependent interaction. Hamm et. al. (2005) introduced a concept of interaction threshold boundary

and proposed a threshold model. This model recognizes that the interaction might occur depending on the dose combination level. Consequently, the dose space could be separated into interactive regions and additive regions by the interaction threshold boundary. Quasi-likelihood ratio test statistics developed from the model can answer the following two questions. First, is there an interaction occurring somewhere in the experimental region? Second, given the fact that interaction occurs somewhere in the experimental region, does an interaction threshold boundary, i.e., a dose-dependent interaction, exist?

While the Hamm et al. (2005) model is an advance, it also has several limitations, some of which have already been identified by the authors. One major limitation acknowledged by the authors is that the interaction boundary may be difficult to specify without input from subject scientists. Hamm et al. (2005) suggested four different forms of threshold boundary for a binary mixture and mentioned that additional forms could be generated from chemists. These functions generally contain five unknown parameters and represent highly different shapes of interaction boundaries. The application results in the paper show that the model fit can be bad with an inappropriate form of the threshold boundary, even for a binary mixture. It can be expected that forms will become more complicated and more diverse for mixtures of M chemicals, $M > 2$. Consequently, it is difficult to even start writing candidate models if there is no prior scientific or reasonable guess of the shape of the threshold boundary.

Another limitation mentioned by Hamm et al. (2005) is that model fitting might be difficult for mixtures of M chemicals when M is large. The total number of

parameters to be estimated for an M -component-mixture interaction threshold model is $1 + M + \binom{M}{2} + \dots + \binom{M}{M} + R = 2^M + R$, where R is the number of parameters needed to define the threshold boundary. Hamm et al. (2005) admitted that it is difficult to collect enough data for estimation when M increases.

Hamm et al. (2005) formulation is also limited in that it accommodates only one kind of interaction; it cannot simultaneously incorporate both synergism and antagonism within the experimental region or even synergism in two disjoint areas of the experimental region. True, this deficiency can be easily overcome by including additional threshold boundaries, but this is not prudent in light of previous comments. Finally, no measure of uncertainty is provided for labeling areas of the experimental region as being synergistic or antagonistic. Hamm et al. (2005) declare an area as being interactive if it falls inside (for their elliptical threshold boundary) or on one side of their threshold boundary, but no confidence region is reported for this threshold boundary.

We propose a simple interaction model that can flexibly describe overall additivity, overall synergism, overall antagonism, and dose-dependent interaction. It assumes no knowledge of activity mechanism for each chemical and no prior knowledge of interactive regions. The model can be readily extended to a mixture of M components, requiring estimation of only $2M + 1$ parameters. Three separate testing procedures—Wald, score, and likelihood ratio—are computed with respect to their abilities to identify interaction type as either additive, synergistic, antagonistic, or dose-dependent. Finally, as an alternative to specifying interaction threshold boundaries that can cause a myriad of complications, we identify interactive dose combinations using

multiplicity-adjusted hypothesis testing. In this way we directly incorporate uncertainty in our method for declaring a dose combination to be interactive. The following section describes our model for binary mixtures.

6.3 A New Nonlinear Model for Binary Mixtures

6.3.1 The Model

Most interaction models of Section 6.2 can be repeated by the simple formula

$$\mu = g_{inter}(x_1, x_2) = g_{add}(x_1h_1, x_2h_2), \quad (6.3)$$

where x_1 and x_2 are doses of the two components, $g_{add}(\cdot)$ is an additivity function for a binary mixture, $g_{inter}(\cdot)$ is an interaction function of the binary mixture, and h_1 and h_2 are two multipliers that modify the effective doses of the individual components. This model has several desirable features. First, functional forms of $g_{inter}(\cdot)$ and $g_{add}(\cdot)$ can be adjusted to achieve compliance with published literature that interactions effectively alter the concentrations of some mixture components. Second, dose-dependent types and strengths of interaction can be captured by this model. And third, the interaction effect can be easily tested by determining whether $h_1 = h_2 = 1$.

We begin by considering an agonist-antagonist mixture, which is the scenario surrounding (6.1). In this case, the additivity model in (6.3) simply becomes the dose-response model of the agonist. More specifically, (6.3) is even more simply represented as

$$\mu = g_{inter}(x_1, x_2) = g_{add}(x_1h_1, x_2h_2) = g_1(x_1h_1), \quad (6.4)$$

where $g_1(\cdot)$ is the dose-response model for the agonist, x_1 is concentration of the agonist, x_2 is concentration of the antagonist, and $h_1 = (1 + x_2/C_{d2})^{-1} \leq 1$ multiplies the concentration of agonist to shift the dose-response curve of agonist to the right without changing its shape.

Suppose the dose-response model for the agonist alone follows the commonly-used logistic model

$$\mu = g_1(x_1) = \frac{1}{1 + \left(\frac{ED50_1}{x_1}\right)^{\rho_1}} = \frac{1}{1 + \left(\frac{EC50_1}{\exp[\log(x_1)]}\right)^{\rho_1}}. \quad (6.5)$$

Then (6.4) and (6.5) define the interaction model as

$$\begin{aligned} \mu &= g_{inter}(x_1, x_2) = g_1(x_1 h_1) \\ &= \frac{1}{1 + \left(\frac{EC50_1}{\exp[\log(x_1) + \log(h_1)]}\right)^{\rho_1}} = \frac{1}{1 + \left(\frac{EC50_1}{\exp[\log x_1 - \log(1 + x_2/C_{d2})]}\right)^{\rho_1}} \end{aligned} \quad (6.6)$$

This interaction model does not change the shape of the dose-response relationship when dose is plotted on the log scale. The only change from (6.5) to (6.6) with respect to plotting as a function of $\log(x_1)$ to achieve the same level of response is a shift of the curve to the right. The magnitude of the shift varies depending on the concentration of the agonist.

Table 6.1: Some (6.3) Binary Interaction Models

Mixture Type	Additivity Model $g_{add}(x_1, x_2)$	Multipliers h 's	Interaction Model $g_{inter}(x_1, x_2)$
Agonist-Antagonist	$\frac{1}{1 + \left(\frac{EC50_1}{x_1}\right)^{\rho_1}}$	$h_1 = \left(1 + \frac{x_2}{K_{d2}}\right)^{-1}$	$\frac{1}{1 + \left(\frac{EC50_1 \left(1 + \frac{x_2}{C_{d2}}\right)}{x_1}\right)^{\rho_1}}$
		$h_1 = \left(1 + \frac{x_2^s}{K_{d2}}\right)^{-1}$	$\frac{1}{1 + \left(\frac{EC50_1 \left(1 + \frac{x_2^s}{C_2}\right)}{x_1}\right)^{\rho_1}}$
		$h_1 = \left(1 + \left(\frac{x_2}{K_{d2}}\right)^s\right)^{-1}$	$\frac{1}{1 + \left(\frac{EC50_1 \left(1 + \left(\frac{x_2}{C_2}\right)^s\right)}{x_1}\right)^{\rho_1}}$
Agonist-Agonist (Brunden et al.)	$q(\beta_1 + \beta_2 x_1 + \beta_3 x_2)$	$h_1 = 1 + \kappa_2 x_2$ $h_2 = 1 + \kappa_1 x_1$	$q(\beta_1 + \beta_2 x_1 + \beta_3 x_2 + \beta_4 x_1 x_2)$ $(\beta_4 = \beta_2 \kappa_2 + \beta_3 \kappa_1)$
Agonist-Agonist (new model)	$q(\beta_1 + \beta_2 x_1 + \beta_3 x_2)$	$h_1 = (1 + x_2)^{s_1}$ $h_2 = (1 + x_1)^{s_2}$	$q(\beta_1 + \beta_2 x_1 (1 + x_2)^{s_1} + \beta_3 x_2 (1 + x_1)^{s_2})$
Agonist "Cassette" - Agonist "Cassette"	$a \frac{1}{1 + \frac{1}{\left(\frac{x_1}{EC50_1} + \frac{x_2}{EC50_2}\right)^{\frac{\rho_1 + \rho_2}{2}}}}$	$h_{Aji} = Z_{Aji}$ $= \left(1 + \left(\frac{x_A}{L_{Aji}}\right)^{s_{Aji}}\right)^{-1}$	$c \ 1 - \prod_{i=1}^N \left(1 - \frac{1}{1 + \frac{1}{\left(\sum_{j=1}^{J_i} \frac{Z_{Aji} x_{ji}}{EC50_{ji}}\right)^{\rho_i}}}\right)$
(Rider and LeBlanc)	$b \ 1 - \prod_{i=1}^2 \left(1 - \frac{1}{1 + \left(\frac{x_i}{EC50_i}\right)^{\rho_i}}\right)$		
	$d \ 1 - \prod_{i=1}^N \left(1 - \frac{1}{1 + \frac{1}{\left(\sum_{j=1}^{J_i} \frac{x_{ji}}{EC50_{ji}}\right)^{\rho_i}}}\right)$		

^aConcentration addition

^bResponse addition

^cInteraction model based on integrated addition

^dIntegrated addition designed for a mixture of more than two chemicals with each chemical classified into one of N "cassettes"

Using the framework specified in (6.3), several binary interaction models are listed in Table 6.1. These include the Brunden et al. and Rider and LeBlanc models as special cases. An important mathematical property of the multipliers of Table 6.1 is that if concentration of the modifying component goes to zero the multipliers become one. In other words, ability of the modifying component in the mixture to modify the other component does not exist when the modifying component is absent. Notice that most multipliers, except those in the fourth and fifth rows, are always less than one, and so they focus only on modeling an antagonistic effect in the system. The model in the fourth row, which is implemented by Brunden et al. (1983) and Meadows et al. (2002), can describe both synergism and antagonism in the mixtures. It unfortunately does not allow you to simultaneously capture both synergism and antagonism. A conclusion of overall synergism, antagonism or additivity over the entire dose combination range can be drawn from the model by testing the sign of β_4 (Meadow et al. 2002). Testing the sign of β_4 is unable to identify dose-dependent interaction.

We focus on the model in row five of Table 6.1. The additivity model is a generalized linear model

$$E[Y] = \mu = q(\beta_1 + \beta_2x_1 + \beta_3x_2),$$

where Y is the response variable from an exponential family with a density function $f(y; \mu)$, and $q^{-1}(\cdot)$ is a link function. Gennings (1995) proved that this additivity model is mathematically equal to the interaction index or the general concept of additivity we proposed in Chapter 5 as long as a common intercept, i.e. β_1 is assumed. The multipliers we propose in our interaction model are $h_1 = (1 + x_2)^{\beta_3}$ and $h_2 = (1 +$

Table 6.2: Different Conclusions of Interaction from (6.7)

Interaction Effect	s_1	s_2	D'
Overall Synergism	> 0	> 0	> 0
	$= 0$	> 0	> 0
	> 0	$= 0$	> 0
Overall Antagonism	< 0	< 0	< 0
	$= 0$	< 0	< 0
	< 0	$= 0$	< 0
Dose-Dependent Interaction	> 0	< 0	$> 0, = 0, \text{ or } < 0$
	< 0	> 0	$> 0, = 0, \text{ or } < 0$
Overall Additivity	$= 0$	$= 0$	$= 0$

$x_1)^{s_2}$, where s_1 and s_2 are two parameters to be estimated. Therefore the interaction model can be presented as

$$E[Y] = \mu = q(\beta_1 + \beta_2 x_1(1 + x_2)^{s_1} + \beta_3 x_2(1 + x_1)^{s_2}). \quad (6.7)$$

The multipliers can be one, less than one, or greater than one and can be controlled by the signs of s_1 and s_2 . Additivity is implied by $s_1 = s_2 = 0$, overall antagonism by $s_1 < 0$ and $s_2 < 0$, overall synergism by $s_1 > 0$ and $s_2 > 0$, and dose-dependent interaction by $s_1 s_2 < 0$. Additional cases of overall antagonism and synergism are shown in Table 6.2.

Given an arbitrary dose combination point, interaction can also be indicated by the difference between the interaction and additivity models, and this can be measured as

$$D = q(\beta_1 + \beta_2 x_1(1 + x_2)^{s_1} + \beta_3 x_2(1 + x_1)^{s_2}) - q(\beta_1 + \beta_2 x_1 + \beta_3 x_2).$$

Under the assumption that each individual component has a monotonically increasing dose-response curve, testing the sign of D is the same as testing the difference between

$\beta_1 + \beta_2 x_1(1 + x_2)^{s_1} + \beta_3 x_2(1 + x_1)^{s_2}$ and $\beta_1 + \beta_2 x_1 + \beta_3 x_2$, i.e., testing the sign of

$$D' = \beta_2 x_1 [(1 + x_2)^{s_1} - 1] + \beta_3 x_2 [(1 + x_1)^{s_2} - 1].$$

Table 6.2 relates D' to different types of interaction. An obvious and important assumption here is that β_1 , β_2 and β_3 have the same interpretation and share the same values in both additivity and interaction models for a given binary mixture. In the real applications, we fit our interaction model with the complete data including the combination data and single-compound data to get the estimates of β_1 , β_2 , and β_3 for both models and test the signs of D' under the assumption that the values of β_1 , β_2 , and β_3 estimated from the complete data are very close to those estimated from additive data. Please note that this assumption is implemented by a number of models in practice. For example, Brunden et al. (1983) fitted the model in row four of table 6.1 with the combination data as well as single-compound data to get the estimates of β_1 , β_2 , and β_3 and tested the sign of β_3 to determine the interaction.

Our new interaction model does change the shape of the dose-response relationship of one drug when the other drug is set at a fixed dose level in the mixture. This is different from the agonist-antagonist models since we are modeling two agonists, i.e., two active compounds, in the mixture. A simple example can be used to further illustrate this difference. Given that the dose-response function of drug 1 is $E[y] = q(\beta_1 + \beta_2 x_1)$, a general form of agonist-antagonist model with x_2 antagonist (i.e., drug 2) is $E[y] = q(\beta_1 + \beta_2 x_1 h_1(x_2))$, where $h_1(x_2)$ is a function of (x_2) , and a general form of our agonist-agonist new model is $E[y] = q(\beta_1 + \beta_2 x_1(1 + x_2)^{s_1} + \beta_3 x_2(1 + x_1)^{s_2})$. The dose-response curves of drug 1 with different doses of drug 2 are plotted on a log-dose scale in Figure 6.1 from two models by arbitrarily setting $\beta_1 = -3$, $\beta_2 = 2$, $\beta_3 = 0.5$,

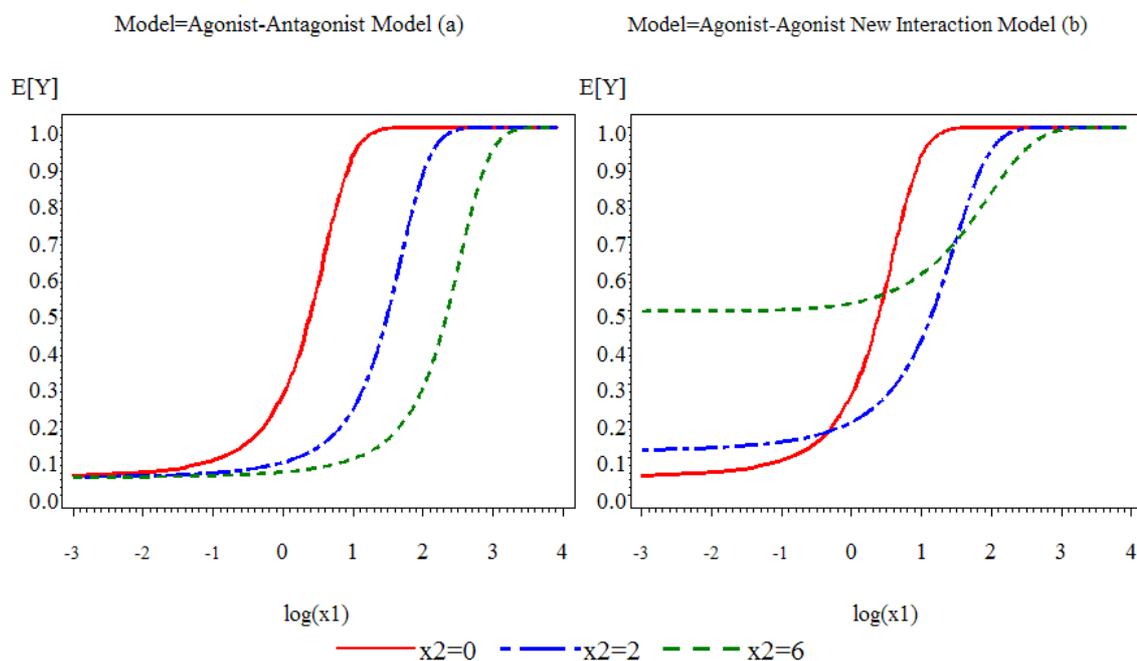


Figure 6.1: Dose-response curves of drug 1 with dose of drug 2=0, 2, 6, from two different models.

$s_1 = -1$, $s_2 = -0.1$, $q(x) = 1/(1 + \exp(-x))$, and $h_1(x_2) = 1/(1 + x_2)$. It can be shown that a fixed dose of drug 2 causes a parallel shift of the dose-response curve of drug 1 to the right in the agonist-antagonist model, whereas it changes the shape of the dose-response curve of drug 1 in our agonist-agonist new interaction model.

6.3.2 Testing for Interaction

Setup

Suppose data are available for K mixture dose points as responses y_1, \dots, y_K and dose combinations (x_{1k}, x_{2k}) , $k = 1, \dots, K$. Assuming the response of K points are

independently distributed, then the likelihood based on (6.7) is

$$L(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_K) = \prod_{k=1}^K f(y_k; \mu_k),$$

where

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}).$$

In order to determine the interaction situation we wish to conduct the following four one-sided tests:

- (1) $H_0^1 : s_1 = 0$ vs. $H_1^1 : s_1 > 0$ (6.8)
- (2) $H_0^2 : s_1 = 0$ vs. $H_1^2 : s_1 < 0$
- (3) $H_0^3 : s_2 = 0$ vs. $H_1^3 : s_2 > 0$
- (4) $H_0^4 : s_2 = 0$ vs. $H_1^4 : s_2 < 0$.

Outcomes from the tests in (6.8) can be combined with interpretation of (6.7) as depicted in Table 6.2 to indicate the type of interaction (if any) present in the samples. Because four tests are conducted in (6.8), at most $2^4 = 16$ four-tuples of outcomes are possible. Let (O_1, O_2, O_3, O_4) be such a four-tuple, where O_j takes value one if H_0^j is rejected in favor of H_1^j and zero otherwise. The four-tuple $(0, 0, 1, 1)$, $(0, 1, 1, 1)$, $(1, 0, 1, 1)$, $(1, 1, 0, 0)$, $(1, 1, 0, 1)$, $(1, 1, 1, 0)$, $(1, 1, 1, 1)$ are not possible. Of the remaining nine possibilities,

- | | |
|--|-----------------------------------|
| $(0, 0, 0, 0)$ | implies additivity |
| $(0, 0, 0, 1), (0, 1, 0, 0), (0, 1, 0, 1)$ | imply antagonism |
| $(0, 0, 1, 0), (1, 0, 0, 0), (1, 0, 1, 0)$ | imply synergism |
| $(0, 1, 1, 0), (1, 0, 0, 1)$ | imply dose-dependent interaction. |

A simple Bonferroni adjustment (Miller 1981, pp. 67-70) for multiple testing can be applied to control the family-wise error rate, i.e., the probability of falsely rejecting at least one null hypothesis, at an α test level. If all null hypotheses are true, controlling the family-wise error rate is referred to as the weak control of family-wise error at level α (Hochberg 1988 and Hommel 1989). The Bonferroni adjustment actually guarantees an even stronger property. Let $H_0 = \{H_0^1, \dots, H_0^4\}$ and $H'_0 \subseteq H_0$ be an arbitrary subset of H_0 . A property is referred to as strong control of family-wise error rate at level α (Hochberg 1988; Hommel 1989) if

$$\Pr_{H'_0}(\text{rejecting at least one } H_0^j; H_0^j \in H'_0) \leq \alpha.$$

In other words, the probability of a false rejection of any $H_0^i, i = 1, \dots, 4$, regardless of which and how many of the H_0^i 's are really true, is always less than or equal to α . Specifically, Bonferroni adjustment in our case rejects an H_0^i when its p-value P_i is less than $\alpha/4$, where $i = 1, \dots, 4$. If we define a Q-value(j) = $P_j N$, $j = 1, \dots, N$, a test is rejected if its Q-value is less than or equal to α . The Bonferroni inequality ensures that the family-wise error rate is no greater than α in a strong way.

The modified Bonferroni test procedure proposed by Hochberg (1988) that increases the power and still strongly controls the family-wise error rate α can also be conducted. Given n null hypotheses, H_0^1, \dots, H_0^n , with observed p-values, P_1, \dots, P_n , the ordered p-values, $P_{(1)} \leq \dots \leq P_{(n)}$, can be determined. Consequently, $J = \max\{j \in 1, \dots, n : P_{(j)} \leq \frac{\alpha}{n+1-j}\}$ can be found and all the $H_0^i, P_i \leq P_{(J)}$, are rejected at level α . If we define a Q-value(j) = $P_{(j)}(n+1-j)$, $j = 1, \dots, N$, J can be simply found using $J = \max\{j \in 1, \dots, N : \text{Q-value}(j) \leq \alpha\}$. This method is subsequently referred to as Hochberg adjustment.

Testing H_0^j Versus H_1^j

We consider three types of one-sided test statistics: Wald, score and likelihood ratio. The one-sided alternatives in (6.8) complicate the testing scenario because asymptotic null distributions for our test statistics are no longer chi-squared as in the standard case. In what follows, we first describe the test statistics then we explain how to obtain asymptotically relevant p-values from the chi-bar-squared distributions under the null hypothesis.

We illustrate all steps using H_0^1 versus H_1^1 , where $\boldsymbol{\theta} = (s_1, s_2, \beta_1, \beta_2, \beta_3)^T$ is partitioned as $\boldsymbol{\theta}^T = (\theta_1, \boldsymbol{\theta}_2^T)$ with $\theta_1 = s_1$ being the parameter of interest and $\boldsymbol{\theta}_2^T = (s_2, \beta_1, \beta_2, \beta_3, \beta_4)^T$ being a vector of nuisance parameters. The Wald test statistic is

$$A_1^W = (\widehat{s}_1 - 0) \left\{ K^{-1} [I_K(\widehat{\boldsymbol{\theta}})^{-1}]_{11} \right\}^{-1} (\widehat{s}_1 - 0),$$

where \widehat{s}_1 and $\widehat{\boldsymbol{\theta}}$ are maximum likelihood estimators under $H_0^1 \cup H_1^1$, and $I_K(\widehat{\boldsymbol{\theta}})$ is the average expected Fisher information matrix evaluated at $\widehat{\boldsymbol{\theta}}$, i.e.,

$$I_K(\widehat{\boldsymbol{\theta}}) = \frac{1}{K} \sum_{k=1}^K E \left\{ -\frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \log f(y_k; x_{1k}, x_{2k}, \boldsymbol{\theta}) \right\} \Big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}},$$

and $[I_K(\widehat{\boldsymbol{\theta}})^{-1}]_{11}$ is the upper (1,1) element of the inverse of $I_K(\widehat{\boldsymbol{\theta}})$. If we partition the corresponding information matrix based on $\boldsymbol{\theta}^T = (\theta_1, \boldsymbol{\theta}_2^T)$ as

$$I_K(\boldsymbol{\theta}) = \begin{pmatrix} I_{11} & I_{12} \\ I_{21} & I_{22} \end{pmatrix},$$

the upper (1,1) element of $I_K(\boldsymbol{\theta})^{-1}$ is

$$[I_K^{-1}]_{11} = (I_{11} - I_{12}I_{22}^{-1}I_{21})^{-1}.$$

Consequently, using the “hat” notation to indicate evaluation at $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}}$, we could write A_1^W as

$$A_1^W = K(\widehat{s}_1 - 0) \left(\widehat{I}_{11} - \widehat{I}_{12} \widehat{I}_{22}^{-1} \widehat{I}_{21} \right) (\widehat{s}_1 - 0) = K \widehat{s}_1^2 \left(\widehat{I}_{11} - \widehat{I}_{12} \widehat{I}_{22}^{-1} \widehat{I}_{21} \right),$$

where \widehat{s}_1 is a maximum likelihood estimator under the one-sided constraint $s_1 \geq 0$, instead of an unconstrained maximum likelihood estimator as would be used in the usual two-sided Wald statistic.

The score test statistic is

$$A_1^S = S_{s_1}(\widetilde{\boldsymbol{\theta}}) \left\{ K(\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21}) \right\}^{-1} S_{s_1}(\widetilde{\boldsymbol{\theta}}) - \inf \left\{ (S_{s_1}(\widetilde{\boldsymbol{\theta}}) - b) \left\{ K(\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21}) \right\}^{-1} (S_{s_1}(\widetilde{\boldsymbol{\theta}}) - b) : b \in H_0^1 \cup H_1^1, \text{i.e., } b \geq 0 \right\},$$

where $\widetilde{\boldsymbol{\theta}}$ is the maximum likelihood estimator under the null hypothesis, $S_{s_1}(\widetilde{\boldsymbol{\theta}})$ is the score function of s_1 evaluated at $\widetilde{\boldsymbol{\theta}}$, and \widetilde{I}_{11} , \widetilde{I}_{12} , \widetilde{I}_{22} , and \widetilde{I}_{21} are partitioned components of the information matrix evaluated at $\widetilde{\boldsymbol{\theta}}$. This test statistic is a special case of score type one-sided test statistics proposed by Silvapulle and Silvapulle (1995, eq. (7)). The general form of their score statistic for testing $H_0 : \boldsymbol{\psi} = 0$ vs. $H_1 : \boldsymbol{\psi} \geq 0$ is

$$T_s = U^T A_{\boldsymbol{\psi}\boldsymbol{\psi}}^{-1} U - \inf \{ (U - b)^T A_{\boldsymbol{\psi}\boldsymbol{\psi}}^{-1} (U - b) : b \geq 0 \},$$

where

$$\boldsymbol{\psi} = s_1,$$

$$U = \left[(I_K(\widetilde{\boldsymbol{\theta}}))^{-1} \right]_{11} (K^{-\frac{1}{2}} S_{s_1}(\widetilde{\boldsymbol{\theta}})) = K^{-\frac{1}{2}} \left(\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21} \right)^{-1} S_{s_1}(\widetilde{\boldsymbol{\theta}}),$$

and

$$A_{\boldsymbol{\psi}\boldsymbol{\psi}} = \left[(I_K(\widetilde{\boldsymbol{\theta}}))^{-1} \right]_{11} = \left(\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21} \right)^{-1}$$

in our case.

It is obvious that the first part of A_1^S is exactly the same as the score statistic for a two-sided test; see Silvapulle and Silvapulle (1995). The second part that involves minimizing a quadratic form can be very easy to calculate. It is easy to show that when $S_{s_1}(\tilde{\boldsymbol{\theta}}) \in H_0^1 \cup H_1^1$, i.e., $S_{s_1}(\tilde{\boldsymbol{\theta}}) \geq 0$ the second part achieves minimum 0 when $b = S_{s_1}(\tilde{\boldsymbol{\theta}})$ and so A_1^S is equal to the two-sided score test statistic. If $-S_{s_1}(\tilde{\boldsymbol{\theta}}) \in H_0^1 \cup H_1^1$, i.e., $S_{s_1}(\tilde{\boldsymbol{\theta}}) < 0$ the second part will be minimized at $b = 0$. Consequently, the minimum of the second part is equal to the first part and A_1^S is zero. In other words,

$$A_1^S = \frac{[S_{s_1}(\tilde{\boldsymbol{\theta}})]^2}{K(\tilde{I}_{11} - \tilde{I}_{12}\tilde{I}_{22}^{-1}\tilde{I}_{21})} I \{S_{s_1}(\tilde{\boldsymbol{\theta}}) \geq 0\},$$

where $I(A) = 1$ if A is true and 0 otherwise.

Last, a likelihood ratio test can be constructed as

$$A_1^L = -2 \log \left\{ \frac{\sup_{\boldsymbol{\theta} \in H_0^1} L(\boldsymbol{\theta}; y_1, \dots, y_K)}{\sup_{\boldsymbol{\theta} \in (H_0^1 \cup H_1^1)} L(\boldsymbol{\theta}; y_1, \dots, y_K)} \right\} = -2 \log L(\tilde{\boldsymbol{\theta}}) + 2 \log L(\hat{\boldsymbol{\theta}}).$$

Several authors (see, for example, Silvapulle and Silvapulle 1995; Robertson, Wright, and Dykstra 1988; Paula 1999; Neto and Paula 2001; Shapiro 1985) have investigated the asymptotic properties of these three types of test statistics under the null hypothesis. All three test statistics, A_1^W , A_1^S , A_1^L have the same asymptotic distribution under the null hypothesis, namely a chi-bar-squared distribution. A chi-bar-squared distribution is a mixture of chi-squared distributions. Since we only test one parameter here, the distribution function for the relevant variable, A , can be expressed as

$$\Pr\{A \geq c\} = \omega \left(1, 0, \left[(KI(\boldsymbol{\theta}; H_0^1))^{-1} \right]_{11} \right) \Pr\{\chi_0^2 \geq c\} \quad (6.9)$$

$$+\omega \left(1, 1, \left[(KI(\boldsymbol{\theta}; H_0^1))^{-1} \right]_{11} \right) \Pr\{\chi_1^2 \geq c\},$$

where χ_0^2 is simply a point mass at zero, χ_1^2 is a chi-squared variable with one degree of freedom, $[I(\boldsymbol{\theta}; H_0^1)^{-1}]_{11}$ is the upper (1,1) component of the inverse information matrix evaluated at H_0^1 , i.e., $s_1 = 0$, and the two ω 's are known as level probabilities (see definition, for instance, in Shapiro 1985).

The expression in (6.9) possibly depends on $\boldsymbol{\theta}_2$, so that the p-value corresponding to test statistic A must be obtained as $\sup_{\boldsymbol{\theta}_2} \Pr(A \geq c)$. Following equation (4.7) of Shapiro (1985), the weights in (6.9) are simply

$$\omega(1, 0, \Lambda) = \omega(1, 1, \Lambda) = 0.5$$

for any positive Λ . Hence, the asymptotic p-value for any of the one-sided Wald, score, or likelihood ratio test is obtained as

$$\text{p-value} = \sup_{\boldsymbol{\theta}_2} \Pr(A \geq c) = \sup_{\boldsymbol{\theta}_2} \left[\frac{1}{2} I(c = 0) + \frac{1}{2} \Pr(\chi_1^2 \geq c) \right].$$

In other words, the p-value from the asymptotic chi-bar-squared distribution is

$$\text{p-value} = \begin{cases} \frac{1}{2} \Pr\{\chi_1^2 \geq c\} & c > 0 \\ 1 & c = 0. \end{cases}$$

The three types of statistics—Wald, score and likelihood ratio—can be constructed for other one-sided tests in (6.8). These three types of statistics share similar properties with their counterparts in two-sided tests (Silvapulle and Silvapulle 1995). Therefore, the Wald statistic is expected to be variant to reparameterization whereas the score statistic and the likelihood ratio statistic are invariant. In addition, the score statistic is generally better than the Wald statistic since its null sampling distribution is closer to the chi-bar-squared asymptotic distribution than that of the Wald

statistic. The likelihood ratio test is also generally better than the Wald statistic in terms of control of type I error probability.

6.3.3 How to Classify Individual Combinations?

Setup

Given that some type of interaction has been identified by at least one of the tests in Section 6.3.2, it is desirable to individually identify all potentially interactive dose combinations, or, even better, to tell if the interaction is synergistic or antagonistic. This is especially important when a conclusion of dose-dependent interaction is drawn from testing the signs of s_1 and s_2 . Depending on the different objectives, we propose different test statistics to solve the problem. This can also be viewed as an alternative method for determining the regions of additivity and interaction without any prior specification on interaction threshold boundaries.

Suppose K mixture dose points $(x_{11}, x_{12}), \dots, (x_{1K}, x_{2K})$ with corresponding responses y_1, \dots, y_K are observed, and we are interested in identifying interaction status for $i = 1, \dots, N (N \leq K)$ of these dose combinations. The basic question is whether an additive model adequately predicts the expected response of combination i , i.e., is $\mu_i = E(y_i)$ well represented by μ_i^A , where μ_i^A represents μ_i obtained from an additive model. Additionally, we may want to detect synergism as $\mu_i > \mu_i^A$ or antagonism as $\mu_i < \mu_i^A$. These goals naturally lead to tests of hypotheses, with the general comparison to additivity requiring two-sided alternatives while synergism and antagonism require one-sided alternatives. Once again we develop Wald, score, and likelihood

ratio tests for these situations, as detailed later in the section.

The issue of multiple testing is again encountered because a series of tests will be pursued for each of N combination dose points. Because Section 6.3.2 was concerned with only four tests in need of multiple adjustment at any given time (see, e.g., (6.8)), the Bonferroni and Hochberg adjustment procedures that both strongly control family-wise error rate were ideal. Unfortunately, the cost of strong control of family-wise error rate is quite high for large N (as will usually be the case) and this can result in dramatic loss of power. To counteract this, Benjamini and Hochberg (1995) proposed another Bonferroni-type procedure (subsequently referred to as the B-H procedure) that aimed to control the false discovery rate rather than family-wise error rate and consequently resulted in increased power. False discovery rate is defined as the expected proportion of rejected null hypotheses that are incorrectly rejected. This rate is the same as family-wise error rate when all null hypotheses are true. Thus, the B-H procedure still weakly controls the family-wise error rate. The increase in power from B-H procedure is large when N increases and the number of false null hypotheses increases. The tradeoff is that the family-wise error rate will be greater than false discovery rate using such a procedure. The B-H procedure rejects all null hypotheses H_0^i 's, where $P_i \leq P_J$, $J = \max\{j \in 1, \dots, N : P_{(j)} \leq \frac{j\alpha}{N}\}$, and $P_{(1)} \leq P_{(2)} \leq P_{(3)} \leq \dots$ are the ordered p-value for all null hypotheses. Similarly, if we define a Q-value(j) = $\frac{P_{(j)}N}{j}$, $j = 1, \dots, N$, J can be simply found by $J = \max\{j \in 1, \dots, N : \text{Q-value}(j) \leq \alpha\}$.

Depending on different objectives, we can select different adjustments (Bonferroni, Hochberg, or B-H) for multiple testing. A scenario where B-H procedure would be preferred is screening of potential dose combinations for future drug development.

We might want to detect as many interactive dose combinations as possible for a combination drug treatment such that we have a high probability of finding an optimal dose combination in terms of both efficacy and safety in the following confirmatory studies. At the same time, we still want to control the proportion of misses at level α , i.e., to control false discovery rate.

The next issue to be addressed is how to obtain μ_i^A . The natural thing to do is to set

$$\mu_i^A = q(\beta_1 + \beta_2 x_{1i} + \beta_3 x_{2i}),$$

where $q^{-1}(\cdot)$ is the link function discussed in Section 6.3.1. Dawson et al. took this approach with the fact that interaction index is equivalent to the additivity model in a generalized linear model form, where $\beta_1, \beta_2, \beta_3$ are estimated from fitting their single-agent dose-response models: $\mu_{1i} = q(\beta_1 + \beta_2 r(x_{1i}))$ and $\mu_{2i} = q(\beta_1 + \beta_3 r(x_{2i}))$ with individual dose-response data based on dose concentrations that are transformed using function $r(\cdot)$. We, on the other hand, estimate $\beta_1, \beta_2, \beta_3$ using the interaction model of (6.7) with individual and combination data. This interaction model will accommodate non-additive behavior for other dose combinations, when these accommodations are necessary. As such, we expect greater accuracy from the interaction model. To avoid overfitting when estimating μ_i^A , our parametric interaction model is implemented to describe all the data except combination i . Moreover, rather than treating μ_i as known without error after substituting observed value y_i , we recognize the variability of y_i and still assume that y_i has a density function $f(y_i; \mu_i)$ with $\mu_i = E[y_i]$. Please note that we do not assume our interaction model on μ_i .

Two-sided Tests

A Wald-type test statistic could be naturally developed for:

$$H_{0,i,two} : \mu_i - \mu_i^A = 0 \quad \text{vs.} \quad H_{1,i,two} : \mu_i - \mu_i^A \neq 0 \quad (6.10)$$

for $i = 1, \dots, N$. Under the assumption that all dose combinations are independent, the likelihood function for estimating $\boldsymbol{\theta} = (s_1, s_2, \beta_1, \beta_2, \beta_3)^T$ is

$$L_i(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_K) = \prod_{k \neq i, k \in K} f(y_k; \mu_k), \quad (6.11)$$

where

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}).$$

We leave y_i out from (6.11) since the probability density of y_i is not related with our interaction model and is free of $\boldsymbol{\theta}$. Under the usual regularity conditions, the maximum likelihood estimator of $\boldsymbol{\theta}$ satisfies

$$\hat{\boldsymbol{\theta}} \sim AN_5 \left(\boldsymbol{\theta}, \frac{I_{K-1}(\boldsymbol{\theta})^{-1}}{K-1} \right),$$

where

$$I_{K-1}(\boldsymbol{\theta}) = \frac{1}{K-1} \sum_{k=1, k \neq i}^K E \left\{ -\frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \log f(y_k; \boldsymbol{\theta}) \right\}.$$

The maximum likelihood estimator of μ_i , $\hat{\mu}_i$, can be obtained from the density function of y_i . Consequently, $\hat{\mu}_i$ has an asymptotical normal distribution

$$\hat{\mu}_i \sim AN_1 \left(\mu_i, \frac{1}{E \left\{ -\frac{\partial^2}{\partial \mu_i \partial \mu_i} \log f(y_i; \mu_i) \right\}} \right).$$

Let $\boldsymbol{\lambda} = (\mu_i, \beta_1, \beta_2, \beta_3, s_1, s_2)^T$. Since $\hat{\mu}_i$ is independent with $\hat{\boldsymbol{\theta}}$, the maximum likelihood estimator of $\boldsymbol{\lambda}$, $\hat{\boldsymbol{\lambda}}$, satisfies

$$\hat{\boldsymbol{\lambda}} \sim AN_6 \left(\boldsymbol{\lambda}, \Gamma_{\boldsymbol{\lambda}} = \begin{bmatrix} \frac{1}{E \left\{ -\frac{\partial^2}{\partial \mu_i \partial \mu_i} \log f(y_i; \mu_i) \right\}} & \mathbf{0}_5^T \\ \mathbf{0}_5 & \frac{I_{K-1}(\boldsymbol{\theta})^{-1}}{K-1} \end{bmatrix} \right),$$

where $\mathbf{0}_5 = (0, 0, 0, 0, 0)^T$. Define $J_i(\boldsymbol{\lambda}) = \mu_i - q(\beta_1 + \beta_2 x_{i1} + \beta_3 x_{i2})$ and $J'_i(\boldsymbol{\lambda}) = \frac{\partial J_i(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}}$.

Then by Delta method

$$J_i(\widehat{\boldsymbol{\lambda}}) \sim AN_1 \left(J_i(\boldsymbol{\lambda}), J'_i(\boldsymbol{\lambda})^T \Gamma_{\boldsymbol{\lambda}} J'_i(\boldsymbol{\lambda}) \right).$$

Therefore, a Wald type statistic

$$T_i^W = \frac{(J_i(\widehat{\boldsymbol{\lambda}}) - 0)^2}{J'_i(\widehat{\boldsymbol{\lambda}})^T \widehat{\Gamma}_{\boldsymbol{\lambda}} J'_i(\widehat{\boldsymbol{\lambda}})}$$

has an asymptotic chi-squared χ_1^2 distribution under $H_{0,i,two}$.

The above Wald-type test statistic can also be derived by simply writing down the likelihood function for $\boldsymbol{\lambda}$ as

$$L_i(\mu_i, \beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_K) = f(y_i; \mu_i) \prod_{k \neq i, k \in K} f(y_k; \mu_k), \quad (6.12)$$

where

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}).$$

Under the regularity conditions $\widehat{\boldsymbol{\lambda}}$ satisfies

$$\widehat{\boldsymbol{\lambda}} \sim AN_6 \left(\boldsymbol{\lambda}, \frac{I_K(\boldsymbol{\lambda})^{-1}}{K} \right),$$

where

$$I_K(\boldsymbol{\lambda}) = \frac{1}{K} \sum_{k=1}^K E \left\{ -\frac{\partial^2}{\partial \boldsymbol{\lambda} \partial \boldsymbol{\lambda}^T} \log f(y_i; \boldsymbol{\lambda}) \right\}.$$

With some simple algebraic manipulation, it can be shown that

$$\frac{I_K(\boldsymbol{\lambda})^{-1}}{K} = \Gamma_{\boldsymbol{\lambda}}.$$

Therefore, the Wald test statistic can be re-written as

$$T_i^W = \frac{(J_i(\widehat{\boldsymbol{\lambda}}) - 0)^2}{J'_i(\widehat{\boldsymbol{\lambda}})^T (K I_K(\widehat{\boldsymbol{\lambda}}))^{-1} J'_i(\widehat{\boldsymbol{\lambda}})}.$$

The score and likelihood ratio test statistics can be easily developed from likelihood function (6.12). A score type statistic using ML estimator of $\boldsymbol{\lambda}$, $\tilde{\boldsymbol{\lambda}}$, subject to the constraint $\mu_i - q(\beta_1 + \beta_2 x_{i1} + \beta_3 x_{i2}) = 0$ is

$$T_i^S = S(\tilde{\boldsymbol{\lambda}})^T \{KI_K(\tilde{\boldsymbol{\lambda}})\}^{-1} S(\tilde{\boldsymbol{\lambda}}),$$

where

$$S(\tilde{\boldsymbol{\lambda}}) = \frac{\partial L_i(\boldsymbol{\lambda}; y_1, \dots, y_K)}{\partial \boldsymbol{\lambda}} \Big|_{\boldsymbol{\lambda}=\tilde{\boldsymbol{\lambda}}}.$$

The likelihood ratio test has the form

$$T_i^L = -2 \log \left\{ \frac{\sup_{\boldsymbol{\lambda} \in H_{0,i,two}} L_i(\boldsymbol{\lambda}; y_1, \dots, y_K)}{\sup_{\boldsymbol{\lambda} \in (H_{0,i,two} \cup H_{1,i,two})} L_i(\boldsymbol{\lambda}; y_1, \dots, y_K)} \right\}$$

One-sided Tests

One-sided Wald, score and likelihood ratio test statistics can be developed for testing synergism and antagonism, i.e., testing

$$H_{0,i} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i} : \mu_i > \mu_i^A$$

and

$$H_{0,i'} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i'} : \mu_i < \mu_i^A.$$

The model needs to be reparameterized before we implement a one-sided score statistic. The one-sided score statistic is designed particularly for testing $H_0 : \boldsymbol{\psi} = 0$ vs. $H_1 : \boldsymbol{\psi} \geq 0$ or $H_0 : \boldsymbol{\psi} = 0$ vs. $H_1 : \boldsymbol{\psi} \leq 0$, where $\boldsymbol{\psi}$ is a vector of parameters or a scalar parameter (Silvapulle and Silvapulle 1995). Therefore, define $\mu_i = q(\beta_1 + \beta_2 x_{1k} + \beta_3 x_{2k} + \delta_i)$, where $-\infty < \delta_i < \infty$. Consequently, testing $H_{0,i}$ vs. $H_{1,i}$ and

$H_{0,i'}$ vs. $H_{1,i'}$ becomes testing

$$H_{0,i} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i} : \delta_i > 0$$

and

$$H_{0,i'} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i'} : \delta_i < 0 .$$

The likelihood function under the reparameterization can be written as

$$L_i(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_K) = f(y_i; \mu_i) \prod_{k \neq i, k \in K} f(y_k; \mu_k), \quad (6.13)$$

where

$$\mu_i = q(\beta_1 + \beta_2 x_{1k} + \beta_3 x_{2k} + \delta_i)$$

and

$$\mu_k = q(\beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2}).$$

Let $\gamma^T = (\gamma_1 = \delta_i, \gamma_2^T = (\beta_1, \beta_2, \beta_3, s_1, s_2))$. The form of the one-sided Wald statistic under the reparameterization for testing $H_{0,i}$ and $H_{1,i}$ is

$$T_i^{W_1} = (\widehat{\delta}_i - 0)^2 \left\{ K [I_K(\widehat{\gamma})^{-1}]_{11} \right\}^{-1} = K \widehat{\delta}_i^2 \left(\widehat{I}_{11} - \widehat{I}_{12} \widehat{I}_{22}^{-1} \widehat{I}_{21} \right),$$

where $\widehat{\gamma}$ is the maximum likelihood estimator of γ under $H_{0,i} \cup H_{1,i}$, \widehat{I}_{11} , \widehat{I}_{12} , \widehat{I}_{21} , and \widehat{I}_{22} are the partitioned components of the information matrix evaluated at $\widehat{\gamma}$.

The one-sided score test has the form

$$T_i^{S_1} = S_{\delta_i}(\widetilde{\gamma}) \left\{ K (\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21}) \right\}^{-1} S_{\delta_i}(\widetilde{\gamma}) - \inf \left\{ (S_{\delta_i}(\widetilde{\gamma}) - b) \left\{ K (\widetilde{I}_{11} - \widetilde{I}_{12} \widetilde{I}_{22}^{-1} \widetilde{I}_{21}) \right\}^{-1} (S_{\delta_i}(\widetilde{\gamma}) - b) : b \in H_{0,i} \cup H_{1,i}, \text{i.e., } b \geq 0 \right\},$$

where $\widetilde{\gamma}$ is the maximum likelihood estimator under null hypothesis, $S_{\delta_i}(\widetilde{\gamma})$ is the score function of δ_i evaluated at $\widetilde{\gamma}$, and \widetilde{I}_{11} , \widetilde{I}_{12} , \widetilde{I}_{22} , and \widetilde{I}_{21} are partitioned components of

the information matrix evaluated at $\tilde{\gamma}$. Since the first part has the same form as a two-sided score test statistic and the score statistic is invariant to reparameterization, the above statistic can also be written as

$$T_i^{S_1} = S(\tilde{\lambda})^T \{K I_K(\tilde{\lambda})\}^{-1} S(\tilde{\lambda}) - \inf \left\{ (S_{\delta_i}(\tilde{\gamma}) - b) \left\{ K(\tilde{I}_{11} - \tilde{I}_{12} \tilde{I}_{22}^{-1} \tilde{I}_{21}) \right\}^{-1} (S_{\delta_i}(\tilde{\gamma}) - b) : b \in H_{0,i} \cup H_{1,i}, \text{i.e., } b \geq 0 \right\}.$$

The computational advantage of this form is that the calculation of $T_i^{S_1}$ only involves the minimization part if the two-sided score test statistic with original parameterization has been calculated. Based on the same argument in Section 6.3.2 we could re-write the one-sided score test statistic as

$$T_i^{S_1} = \frac{[S_{\delta_i}(\tilde{\gamma})]^2}{K(\tilde{I}_{11} - \tilde{I}_{12} \tilde{I}_{22}^{-1} \tilde{I}_{21})} I \{S_{\delta_i}(\tilde{\gamma}) \geq 0\},$$

where $I(A) = 1$ if A is true and 0 otherwise.

A one-sided likelihood ratio test statistic can be easily derived

$$T_i^{L_1} = -2 \log \left\{ \frac{\sup_{\gamma \in H_{0,i}} L_i(\gamma; y_1, \dots, y_K)}{\sup_{\gamma \in (H_{0,i} \cup H_{1,i})} L_i(\gamma; y_1, \dots, y_K)} \right\}$$

Again, the three statistics have the same asymptotic, chi-bar-squared distribution. The approximate p-value's based on asymptotic distribution can be determined using a weighted sum of two chi-squared statistics with weights being 0.5 for each. After the p-value for each test is calculated, multiple testing procedures can be applied to determine the significant individual test(s), i.e., synergistic/antagonistic dose combinations.

6.4 A Nonlinear Model for Mixtures of M Components and Hypotheses Tests

The model and the testing procedures proposed in Section 6.3 can be generalized to a mixture of M components under the assumption that each individual component has a monotonically increasing dose-response curve.

The additivity model for a mixture of M components has the general form:

$$E[y] = \mu = q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_m\right),$$

where x_1, \dots, x_M are the doses of the M components, y is the response variable from an exponential family with a density function $f(y; \mu)$, μ is the expected response and $q^{-1}(\cdot)$ is a link function in Generalized Linear Models (GLM's). Gennings (1995) proved that this additivity model is mathematically equal to the interaction index as long as a common intercept, i.e. β_1 , is assumed. The multipliers we propose in our interaction model have a general form as

$$h_m = \left(1 + \sum_{j=1, j \neq m}^M x_j\right)^{s_m}, m = 1, \dots, M,$$

where s_1, \dots, s_M are parameters to be estimated. Therefore the interaction model can be presented as

$$E[y] = \mu = q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_m h_m\right).$$

The difference between the interaction model and the additivity model over entire dose range can be measured as

$$D = q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_m h_m\right) - q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_m\right).$$

Under the assumption that each individual component has a monotonically increasing dose-response curve, testing the sign of D is equivalent to testing the difference between $\beta_1 + \sum_{m=1}^M \beta_{m+1} x_m h_m$ and $\beta_1 + \sum_{m=1}^M \beta_{m+1} x_m$ i.e., testing the sign of

$$D' = \sum_{m=1}^M \beta_{m+1} x_m (h_m - 1).$$

Some trivial algebraic deductions yield

$$\begin{array}{lll} D' = 0 & \text{(Overall Additivity)} & \text{if } \sum_{m=1}^M s_m = 0; \\ D' > 0 & \text{(Overall Synergism)} & \text{if } s_i > 0, i \in \{m : s_m \neq 0\}; \\ D' < 0 & \text{(Overall Antagonism)} & \text{if } s_i < 0, i \in \{m : s_m \neq 0\}; \\ D' > 0, = 0, < 0 & \text{(Dose-Dependent Interaction)} & \text{if } \{(i, j) : s_i s_j < 0, \\ & & i, j \in \{m : s_m \neq 0\} \neq \emptyset. \end{array}$$

In order to determine the interaction situation we test $2M$ hypotheses simultaneously

$$\begin{array}{ll} (1) & H_0^1 : s_1 = 0 \quad H_1^1 : s_1 > 0 \\ (2) & H_0^2 : s_1 = 0 \quad H_1^2 : s_1 < 0 \\ & \dots \quad \dots \\ (2M-1) & H_0^{2M-1} : s_M = 0 \quad H_1^{2M-1} : s_M > 0 \\ (2M) & H_0^{2M} : s_M = 0 \quad H_1^{2M} : s_M < 0 \end{array}$$

Suppose K mixture dose points $(x_{11}, \dots, x_{M1}), \dots, (x_{1K}, \dots, x_{MK})$ and the corresponding responses y_1, \dots, y_K are observed independently; consequently, the likelihood function for the sample is

$$L(\beta_1, \dots, \beta_{M+1}, s_1, \dots, s_M; y_1, \dots, y_K) = \prod_{k=1}^K f(y_k; \mu_k),$$

where

$$\mu_k = q\left(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}\right).$$

Three one-sided test statistics (i.e., Wald, score and likelihood ratio) against one-sided alternatives can be constructed in the same way as we propose in Section 6.3.2. They are asymptotically equivalent to a chi-bar-squared distribution under the null hypothesis. Again, simple Bonferroni adjustment or Hochberg adjustment can be applied to control the family-wise error rate of multiple testing.

In order to identify the interaction status of individual dose combinations, two-sided or one-sided tests for each mixture dose point of interest can be simultaneously conducted. Suppose K mixture dose points $(x_{11}, \dots, x_{M1}), \dots, (x_{1K}, x_{MK})$ and the corresponding independent responses y_1, \dots, y_K are observed. If we are interested in identifying interactive dose combinations among N candidates, N two-sided hypothesis tests

$$H_{0,i,two} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i,two} : \mu_i \neq \mu_i^A,$$

where $\mu_i^A = q(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mi})$, can be conducted simultaneously. Three two-sided test statistics will be developed following the same method proposed in Section 6.3.3. They all have an asymptotic chi-squared distribution under the null hypothesis. If detection of synergistic or antagonistic combinations is of interest, two one-sided tests can be formed for each dose combination as

$$H_{0,i} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i} : \mu_i > \mu_i^A$$

or

$$H_{0,i'} : \mu_i = \mu_i^A \quad \text{vs.} \quad H_{1,i'} : \mu_i < \mu_i^A.$$

Using the same reparameterization method described in Section 6.3.3, we define

$$\mu_i = q\left(\beta_1 + \left(\sum_{m=1}^M \beta_{m+1} x_{mi}\right) + \delta_i\right),$$

where $-\infty < \delta_i < \infty$. Consequently, testing $H_{0,i}$ vs. $H_{1,i}$ and $H_{0,i'}$ vs. $H_{1,i'}$ becomes testing

$$H_{0,i} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i} : \delta_i > 0$$

and

$$H_{0,i'} : \delta_i = 0 \quad \text{vs.} \quad H_{1,i'} : \delta_i < 0.$$

Three one-sided test statistics will be developed in the same way as Section 6.3.3. They have the same asymptotic distribution, chi-bar-squared distribution. Multiple testing procedures including Hochberg adjustment and B-H procedure can be applied.

6.5 Real Data Sets with Quantal Response

6.5.1 Amethopterin and 6-mercaptopurine Data Set

The data set comes from Brunden et al. (1988, pp. 103-104). Goldin, Venditti, Humphreys, and Dennis (1955) first investigated the data set to determine if there is increased (i.e., synergistic) toxicity of amethopterin and 6-mercaptopurine when the two drugs are administered in combination to treat leukemia in mice. Brunden et al. used the generalized linear model displayed as the fourth row of Table 6.1 to analyze the data. After rejecting $H_0 : \beta_4 = 0$ versus $H_1 : \beta_4 \neq 0$ and observing a positive estimate of β_4 , Brunden et al. declared an overall synergistic effect between the two drugs.

The data set includes single-component testing for each chemical and mixture testing for various dose combinations, as shown in Appendix C. All of the dose com-

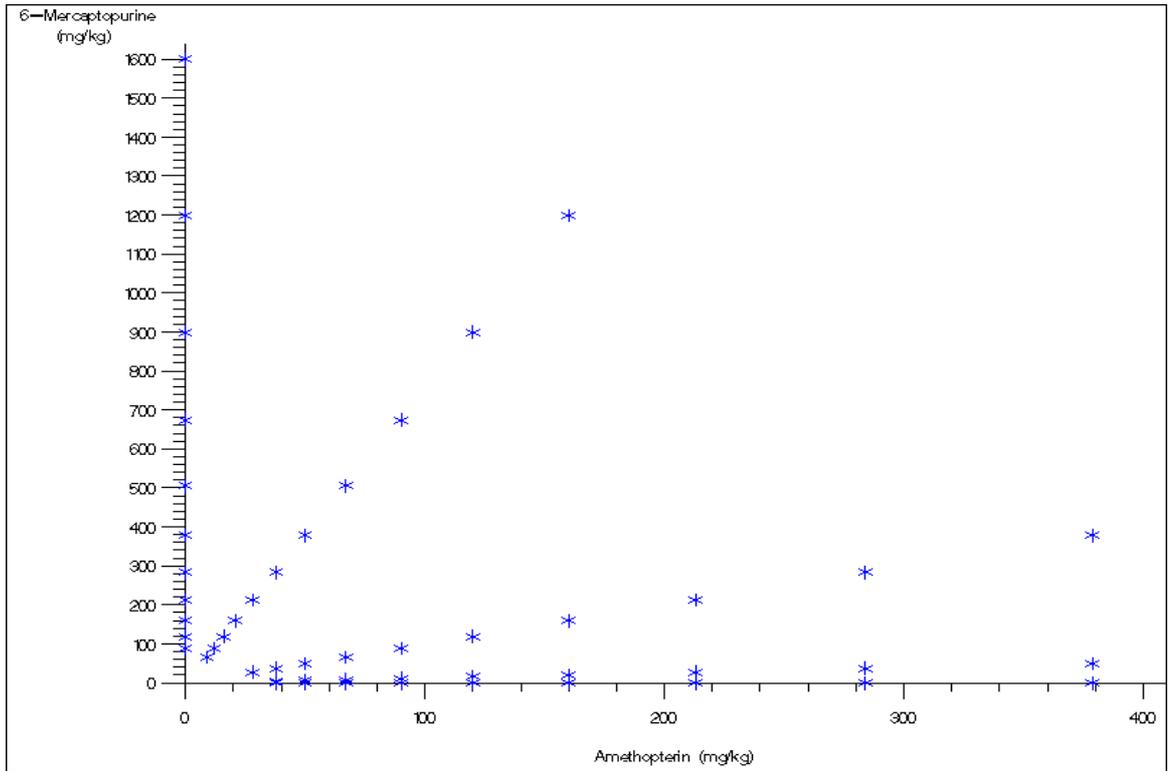


Figure 6.2: The scatter plot of the dose of 6-mercaptopurine versus the dose of Amethopterin.

binations in the study are shown in Figure 6.2. Let y_k denote the sample proportion of dead animals in batch k , i.e. mortality proportion in batch k where dose point k was applied to. Assuming that the w_k animals in a given batch can reasonably be treated as independent trials, the number $w_k y_k$ of dead animals in group k , has a binomial distribution, $\text{Bin}(w_k, \mu_k)$, where μ_k is the expected value of y_k , i.e., the probability of an animal in group k being dead.

Our interaction model (fifth row of Table 6.1) with a logit link can be implemented to model all μ_k 's as

$$\log \frac{\mu_k}{1 - \mu_k} = \beta_1 + \beta_2 x_{1k} (1 + x_{2k})^{s_1} + \beta_3 x_{2k} (1 + x_{1k})^{s_2},$$

where x_{1k} , x_{2k} are the doses of amethopterin and 6-mercaptopurine, respectively, for the k^{th} batch. Consequently, the likelihood function is,

$$L(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_{57}) \propto \prod_{k=1}^{57} \mu_k^{w_k y_k} (1 - \mu_k)^{w_k - w_k y_k}.$$

Two goodness-of-fit statistics, the deviance (Agresti 2002, pp. 141-142) and the Pearson statistic (Agresti 2002, p. 220), were calculated to check the model fit. The results suggests that the model provides an acceptable fit.

	Statistic	Degrees of freedom	p-value
Deviance	43.79	52	0.787
Pearson	40.33	52	0.880

The estimated parameter values and the Wald standard errors are

Parameter	Estimate	Standard Error
β_1	-3.90	0.36
β_2	0.02	0.002
β_3	0.006	0.001
s_1	-0.05	0.04
s_2	0.22	0.04

The four one-sided tests of s_1 and s_2 in (6.8) were conducted. Wald, score and likelihood ratio test statistics were calculated as described in Section 6.3.2 using PROC NLMIXED of SAS 8.2 (SAS Inc., 2000) and SAS/IML (SAS Inc., 1999b); see Appendix A for technical details and Appendix B for SAS code. We used asymptotic p-values for all three types of tests. The Hochberg (1988) adjustment and Bonferroni adjustment were implemented separately to adjust for multiplicity and to control the total family-wise error rate at $\alpha = 0.05$. Results are summarized in Table 6.3.

Table 6.3: Test results of amethopterin and 6-mercaptopurine data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.

Hypothesis Test	Raw P-value		Q-Value & Significance	
			Hochberg Adjustment	Bonferroni Adjustment
$H_0^1 : s_1 = 0$ vs. $H_1^1 : s_1 > 0$	Wald	1	2 (N)	4 (N)
	Score	1	2 (N)	4 (N)
	LRT	1	2 (N)	4 (N)
$H_0^2 : s_1 = 0$ vs. $H_1^2 : s_1 < 0$	Wald	0.140	0.420 (N)	0.560 (N)
	Score	0.120	0.360 (N)	0.480 (N)
	LRT	0.129	0.387 (N)	0.516 (N)
$H_0^3 : s_2 = 0$ vs. $H_1^3 : s_2 > 0$	Wald	1.60×10^{-8}	6.40×10^{-8} (Y)	6.40×10^{-8} (Y)
	Score	2.82×10^{-6}	1.13×10^{-5} (Y)	1.13×10^{-5} (Y)
	LRT	3.64×10^{-7}	1.46×10^{-6} (Y)	1.46×10^{-6} (Y)
$H_0^4 : s_2 = 0$ vs. $H_1^4 : s_2 < 0$	Wald	1	1 (N)	4 (N)
	Score	1	1 (N)	4 (N)
	LRT	1	1 (N)	4 (N)

It can be seen that all three types of test statistics give consistent test results no matter which multiplicity adjustment method is used. Only s_2 is significantly different from zero and it is positive. Therefore, an overall synergistic effect exists between amethopterin and 6-mercaptopurine based on our model. Our results agree with Brunden et al. and Goldin et al. Traditional isobologram graph is shown in Figure 6.3, which includes two isoboles based on two response levels: 50 percent (EC_{50}) and 95 percent (EC_{95}). We show isoboles obtained from the fitted interaction model as well as the additive isoboles that result from connecting single-agent fitted values using a straight line. It can be seen that most points of two isoboles are less than the corresponding additive isoboles, clearly indicating a synergistic effect. Both of the two isoboles have a very small segment above the additive isoboles when

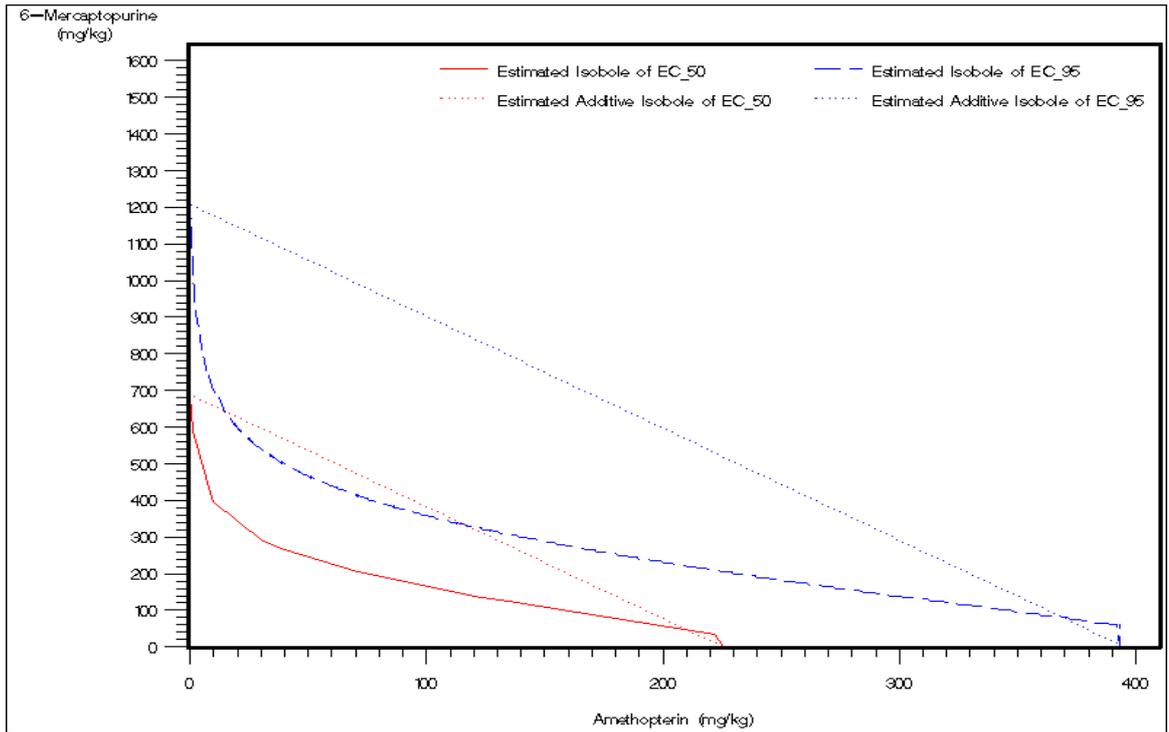


Figure 6.3: Isobologram of amethopterin and 6-mercaptopurine.

amethopterin is dominating the mixture, i.e., the dose of 6-mercaptopurine is close to zero. These segments indicate a weak antagonistic effect. The isobologram, however, is created by the interaction model only using point estimates of each parameter. Since the point estimates of s_1 and s_2 have different signs, we expect to see some antagonistic region in the isobologram graph. All three test statistics conclude that s_1 is not significantly different from zero whereas s_2 is significantly greater than zero, i.e., an overall synergism is implied by test results. Therefore, the small antagonistic segments of the two isoboles are not significantly different from the corresponding additive isoboles whereas the synergistic regions have strong evidence of existing synergism when variability is considered.

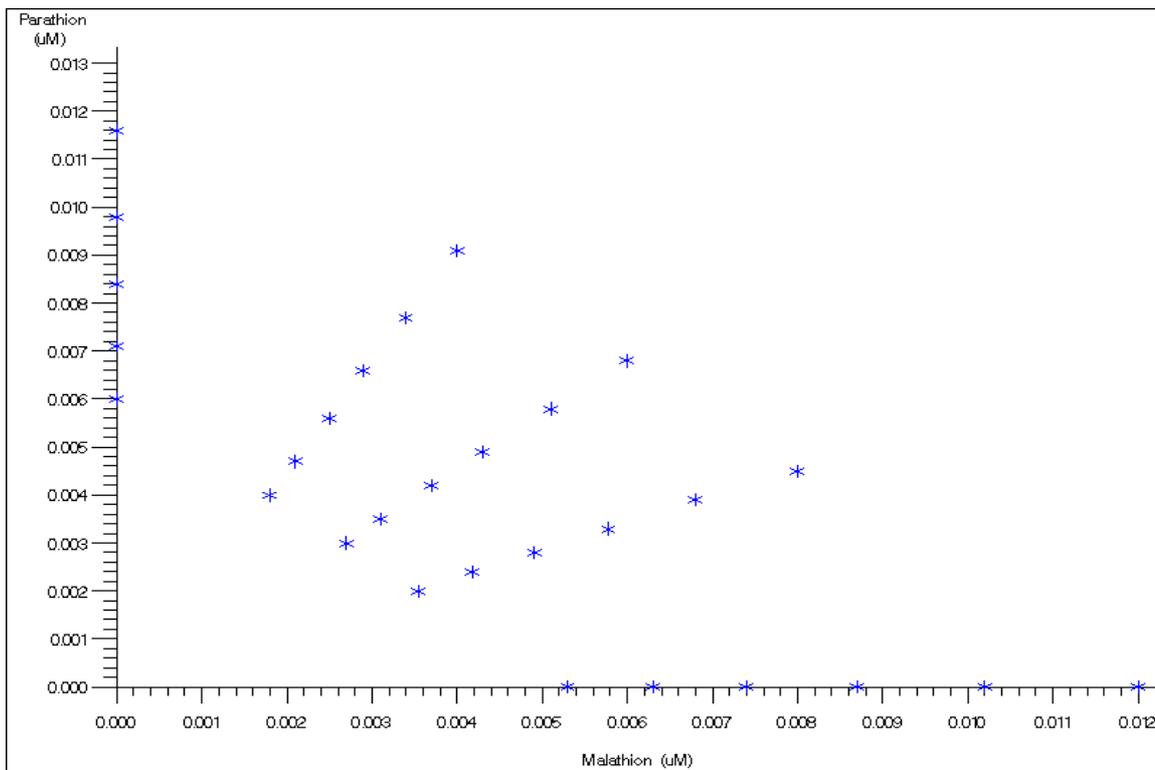


Figure 6.4: The scatter plot of the dose of malathion versus the dose of parathion.

6.5.2 Malathion and Parathion Data Set

The data set comes from Rider and LeBlanc (2005). Two acetylcholinesterase inhibiting organophosphates, malathion and parathion, were combined at five fixed mixing ratios (including 1:0 and 0:1, i.e., single-component mixtures) to demonstrate that the two components are concentration additive, i.e., non-interactive. The common mode of action for malathion and parathion was experimentally confirmed in the same paper.

Figure 6.4 shows all of the dose combinations in the study.

Every dose combination or a single-component dose was used to treat a group of 20 neonatal daphnids. The number of immobilized daphnids after treatment was recorded to form our quantal response. The data is given in Appendix C. Denote the sample proportion of immobilized daphnids in group k as y_k . Again we assume that $20y_k$, the number of immobilized daphnids in group k , has a binomial distribution, $\text{Bin}(20, \mu_k)$, where μ_k is the expected value of y_k , i.e., the probability that an individual daphnid treated in group k is immobilized. Our interaction model (fifth row of Table 6.1 with a logit link can be implemented to model all μ_k 's as

$$\log \frac{\mu_k}{1 - \mu_k} = \beta_1 + \beta_2 x_{1k} (1 + x_{2k})^{s_1} + \beta_3 x_{2k} (1 + x_{1k})^{s_2},$$

where x_{1k} and x_{2k} are the doses of malathion and parathion, respectively, administered in group k . The likelihood function can be written as,

$$L(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_{30}) \propto \prod_{k=1}^{30} \mu_k^{20y_k} (1 - \mu_k)^{20-20y_k}.$$

Again, the deviance and the Pearson statistic were calculated to check the model fit. The results suggest that the model provides a weak fit.

	Statistic	Degrees of freedom	p-value
Deviance	32.21	25	0.152
Pearson	29.83	25	0.231

The estimated parameter values and the Wald standard errors are

Parameter	Estimate	Standard Error
β_1	-15.49	1.67
β_2	1737.59	197.23
β_3	1612.46	176.80
s_1	-284.44	116.27
s_2	103.50	20.19

Table 6.4: Test results of marathion and parathion data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.

Hypothesis Test	Raw P-value		Q-value & Significance	
			Hochberg Adjustment	Bonferroni Adjustment
$H_0^1 : s_1 = 0$ vs. $H_1^1 : s_1 > 0$	Wald	0.0175	0.0525 (N)	0.0700 (N)
	Score	0.0161	0.0644 (N)	0.0644 (N)
	LRT	0.0219	0.0876 (N)	0.0876 (N)
$H_0^2 : s_1 = 0$ vs. $H_1^2 : s_1 < 0$	Wald	1	2 (N)	4 (N)
	Score	1	2 (N)	4 (N)
	LRT	1	2 (N)	4 (N)
$H_0^3 : s_2 = 0$ vs. $H_1^3 : s_2 > 0$	Wald	1	1 (N)	4 (N)
	Score	1	1 (N)	4 (N)
	LRT	1	1 (N)	4 (N)
$H_0^4 : s_2 = 0$ vs. $H_1^4 : s_2 < 0$	Wald	0.0074	0.0296 (Y)	0.0296 (Y)
	Score	0.0418	0.1254 (N)	0.1672 (N)
	LRT	0.0203	0.0609 (N)	0.0812 (N)

The four one-sided tests of s_1 and s_2 were conducted. Wald, score and likelihood ratio test statistics were calculated (see Appendices A and B) and asymptotic p-values were used for all three tests. The Hochberg and Bonferroni adjustments were implemented separately to adjust for multiplicity and to control the total family error rate at $\alpha = 0.05$. The test results are shown in Table 6.4.

Both score and likelihood ratio test statistics state that neither s_1 nor s_2 is significantly different from zero under Bonferroni adjustment or under Hochberg adjustment. Therefore, we have a weak evidence that an interaction exists between two chemicals. This conclusion is supported by Rider and LeBlanc's finding (2005) that the two compounds have a similar action mechanism and do not interact with each other.

The Wald test statistics under both adjustments, however, say that s_2 is signif-

icantly less than zero whereas s_1 is not significantly different from zero. Therefore, an overall antagonism is found based on Wald test results. It seems that the score and likelihood ratio test results are more consistent with experimental findings and Rider and LeBlanc's conclusion than Wald test results. Indeed, much evidence exists in the literature that suggest very liberal properties of the Wald test; see for example, Berndt and Savin (1977) and Boos and Stefanski (2005 pp. 57-58).

Again, isobologram graph is shown in Figure 6.5, which includes two isoboles based on two response levels: 50 percent (EC_{50}) and 95 percent (EC_{95}). Since the point estimates of s_1 and s_2 have different signs, it can be expected that the isoboles will present the features of dose-dependent interaction, i.e., both synergistic region and antagonistic region can exist. The two isoboles, however, are very close to the corresponding additive isoboles by visual comparison. The score and likelihood ratio test statistics both show no strong evidence that s_1 and s_2 are different from zero. Therefore, the isoboles of EC_{50} and EC_{95} are not statistically different from the corresponding additive isoboles at 0.05 level.

6.5.3 Tramadol and Acetaminophen Data Set

A patent of co-administration of tramadol (TRAM) and acetaminophen (APAP) was granted to McNeilab, Inc. for synergistic effect generated at certain dose combinations (U.S. patent 5,336,691).

Dawson et al. (2000) investigated interaction between TRAM and APAP on a study in which each single-chemical dose or a dose combination was applied to a

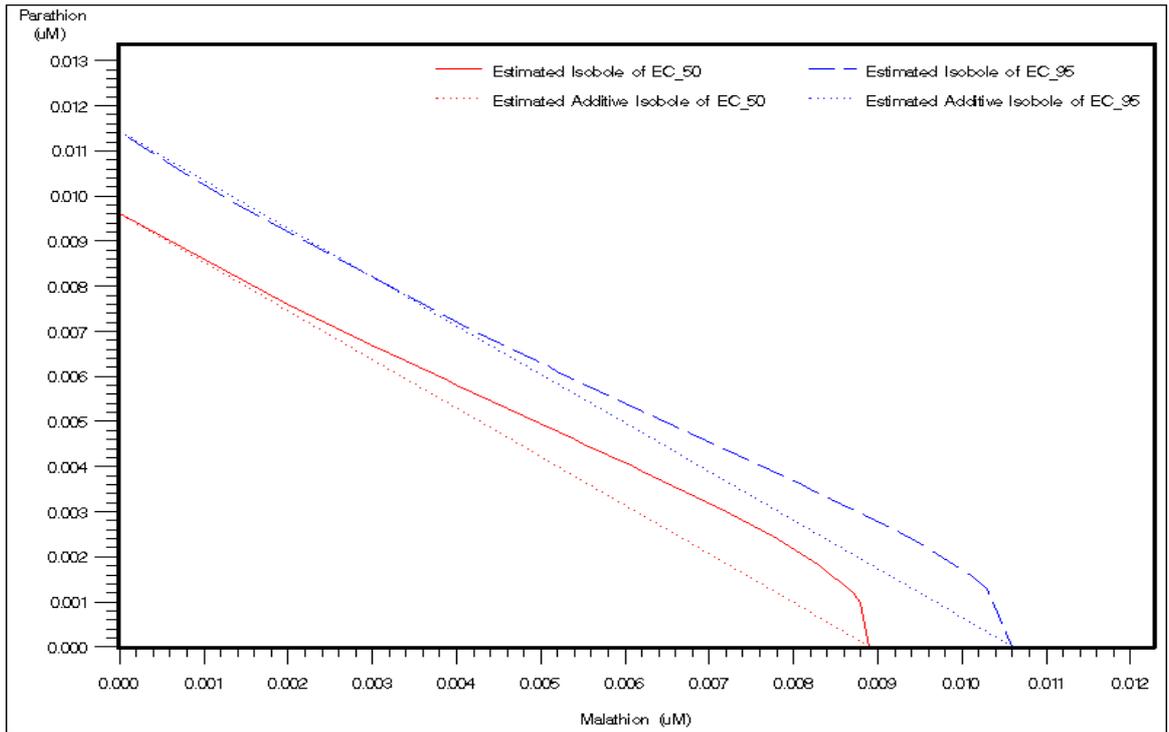


Figure 6.5: Isobologram of malathion and parathion.

group of several mice and the presence/absence of analgesia was recorded as 1/0 for each mouse. Dawson et al.(2000) used their model with a particular transformational form of doses to demonstrate that a few dose combinations are interactive. Although they eliminated a large portion of combination data from their analysis, their results seemed to suggest that interaction can only occur at certain binary mixture dose points.

The data includes 72 different dose combinations as well as single-compound data (see Appendix C and Figure 6.12). Letting y_k represent the sample proportion of the w_k mice treated at the k^{th} dose point that have analgesia, we assume $w_k y_k$ is distributed as $\text{Bin}(w_k, \mu_k)$ where μ_k , the probability of having analgesia, is modeled

as

$$\log \frac{\mu_{kj}}{1 - \mu_{kj}} = \beta_1 + \beta_2 x_{1k}(1 + x_{2k})^{s_1} + \beta_3 x_{2k}(1 + x_{1k})^{s_2},$$

where x_{1k} and x_{2k} are doses of TRAM and APAP, respectively, applied to group k , $k = 1, \dots, 91$.

The likelihood function of (y_1, \dots, y_{91}) can be written as,

$$L(\beta_1, \beta_2, \beta_3, s_1, s_2; y_1, \dots, y_{91}) \propto \prod_{k=1}^{91} \mu_k^{w_k y_k} (1 - \mu_k)^{w_k - w_k y_k}.$$

The estimated parameter values and the Wald standard errors are

Parameter	Estimate	Standard Error
β_1	-2.11	0.01
β_2	0.31	0.02
β_3	0.01	0.001
s_1	0.12	0.02
s_2	-0.40	0.10

One-sided tests of the signs of s_1 and s_2 were conducted using the Wald, score and likelihood ratio test statistics (see Appendix A) and asymptotic p-values were used to approximate p-values of three tests. Hochberg (1988) adjustment and Bonferroni adjustment were implemented separately to adjust for multiplicity and to control the total family-wise error rates at $\alpha = 0.05$. The test results are shown in Table 6.5.

All three test statistics give the same conclusion under both adjustments for all but one case. The broad conclusion is that s_1 is significantly greater than zero and s_2 is significantly less than zero. This implies that interaction exists but the type depends on the dose combination; that is, an interactive effect is dose-dependent.

Table 6.5: Test results of tramadol and acetaminophen data. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is four.

Hypothesis Test	Raw P-value		Q-Value & Significance	
			Hochberg Adjustment	Bonferroni Adjustment
$H_0^1 : s_1 = 0$ vs. $H_1^1 : s_1 > 0$	Wald	2.33×10^{-6}	9.32×10^{-6} (Y)	9.32×10^{-6} (Y)
	Score	1.67×10^{-5}	6.68×10^{-5} (Y)	6.68×10^{-5} (Y)
	LRT	8.33×10^{-6}	2.53×10^{-5} (Y)	2.53×10^{-5} (Y)
$H_0^2 : s_1 = 0$ vs. $H_1^2 : s_1 < 0$	Wald	1	2 (N)	4 (N)
	Score	1	2 (N)	4 (N)
	LRT	1	2 (N)	4 (N)
$H_0^3 : s_2 = 0$ vs. $H_1^3 : s_2 > 0$	Wald	1	1 (N)	4 (N)
	Score	1	1 (N)	4 (N)
	LRT	1	1 (N)	4 (N)
$H_0^4 : s_2 = 0$ vs. $H_1^4 : s_2 < 0$	Wald	3.25×10^{-3}	9.75×10^{-3} (Y)	0.13×10^{-2} (Y)
	Score	3.36×10^{-4}	3.25×10^{-2} (Y)	1.34×10^{-3} (Y)
	LRT	1.03×10^{-3}	3.09×10^{-3} (Y)	4.12×10^{-3} (Y)

Suppose we are also interested in individually testing all 72 combinations (combination ID, i.e., k in our model, goes from 20 to 91) to determine their interaction status. 72 two-sided tests were conducted under the multiplicity adjustment. Three types of test statistics were calculated (see Appendix A) for each of the 72 two-sided alternative hypotheses and adjusted for multiplicity using both the Hochberg adjustment and the B-H procedure to achieve family-wise error rate $\alpha = 0.05$ or false discovery rate $\alpha = 0.05$. P-values were approximated using asymptotic chi-squared distribution. Significant test results found by at least one type of statistic under either Hochberg adjustment or B-H procedure are shown in Table 6.6.

Alternatively, we also tested two one-sided hypotheses for each of 72 combinations under multiplicity adjustment to determine if there are any synergistic or antagonistic

Table 6.6: Two-Sided test results of individual combinations of tramadol and acetaminophen. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is 72.

(x_1, x_2) mg/kg po $(w_k y_k)/w_k$	Hypothesis Test	Raw P-value		Q-value & Significance	
				Hochberg Adjustment	B-H Procedure
(4, 400) 55/60	$H_{0,68,two} : \delta_{68} = 0$ $H_{1,68,two} : \delta_{68} \neq 0$	Wald	0.29×10^{-2}	0.2048 (N)	0.1039 (N)
		Score	9.17×10^{-4}	0.0642 (N)	0.0220 (Y)
		LRT	7.10×10^{-3}	0.4583 (N)	0.0635 (N)
(10, 50) 30/30	$H_{0,45,two} : \delta_{45} = 0$ $H_{1,45,two} : \delta_{45} \neq 0$	Wald	9.94×10^{-1}	5.9585 (N)	1.0672 (N)
		Score	0.15×10^{-1}	0.9771 (N)	0.1353 (N)
		LRT	1.29×10^{-3}	0.0880 (N)	0.0187 (Y)
(0.125, 25) 1/60	$H_{0,70,two} : \delta_{70} = 0$ $H_{1,70,two} : \delta_{70} \neq 0$	Wald	0.15×10^{-1}	0.9860 (N)	0.1766 (N)
		Score	0.20×10^{-2}	0.1361 (N)	0.0355 (Y)
		LRT	1.46×10^{-4}	0.0103 (Y)	0.0035 (Y)
(0.25, 100) 18/30	$H_{0,78,two} : \delta_{78} = 0$ $H_{1,78,two} : \delta_{78} \neq 0$	Wald	0.33×10^{-2}	0.2302 (N)	0.0789 (N)
		Score	0.21×10^{-2}	0.1398 (N)	0.0296 (Y)
		LRT	2.92×10^{-3}	0.1955 (N)	0.0350 (Y)
(0.25, 200) 27/30	$H_{0,85,two} : \delta_{85} = 0$ $H_{1,85,two} : \delta_{85} \neq 0$	Wald	0.58×10^{-2}	0.3932 (N)	0.0833 (N)
		Score	0.21×10^{-2}	0.1407 (N)	0.0252 (Y)
		LRT	0.85×10^{-3}	0.0587 (N)	0.0153 (Y)
(7.5, 22.5) 29/30	$H_{0,42,two} : \delta_{42} = 0$ $H_{1,42,two} : \delta_{42} \neq 0$	Wald	0.50×10^{-2}	0.3478 (N)	0.0907 (N)
		Score	1.58×10^{-4}	0.0112 (Y)	0.0057 (Y)
		LRT	1.12×10^{-5}	0.0008 (Y)	0.0004 (Y)
(2, 400) 48/60	$H_{0,74,two} : \delta_{74} = 0$ $H_{1,74,two} : \delta_{74} \neq 0$	Wald	1.17×10^{-6}	0.0001 (Y)	0.0001 (Y)
		Score	1.89×10^{-8}	0.0000 (Y)	0.0000 (Y)
		LRT	2.65×10^{-6}	0.0002 (Y)	0.0002 (Y)

Table 6.7: One-Sided test results of individual combinations of tramadol and acetaminophen. Y=reject null hypothesis; N=accept null hypothesis. The total number of tests adjusted for multiplicity is 144.

(x_1, x_2) mg/kg po y_k/w_k	Hypothesis Test	Raw P-value		Significance	
				Hochberg Adjustment	B-H Procedure
(4, 400) 55/60	$H_{0,68'} : \delta_{68} = 0$ $H_{1,68'} : \delta_{68} < 0$	Wald	1.44×10^{-3}	0.2062 (N)	0.1038 (N)
		Score	4.59×10^{-4}	0.0651 (N)	0.0220 (Y)
		LRT	0.35×10^{-2}	0.9588 (N)	0.1128 (N)
(10, 50) 30/30	$H_{0,45} : \delta_{45} = 0$ $H_{1,45} : \delta_{45} > 0$	Wald	4.97×10^{-1}	35.751 (N)	0.9795 (N)
		Score	0.75×10^{-2}	1.0298 (N)	0.1353 (N)
		LRT	6.47×10^{-4}	0.0906 (N)	0.0186 (Y)
(0.125, 25) 1/60	$H_{0,70'} : \delta_{70} = 0$ $H_{1,70'} : \delta_{70} < 0$	Wald	0.74×10^{-2}	1.0228 (N)	0.1766 (N)
		Score	0.10×10^{-2}	0.1391 (N)	0.0355 (Y)
		LRT	7.32×10^{-5}	0.0208 (Y)	0.0070 (Y)
(0.25, 100) 18/30	$H_{0,78} : \delta_{78} = 0$ $H_{1,78} : \delta_{78} > 0$	Wald	1.64×10^{-3}	0.2335 (N)	0.0789 (N)
		Score	1.03×10^{-3}	0.1439 (N)	0.0252 (Y)
		LRT	0.15×10^{-2}	0.2028 (N)	0.0350 (Y)
(0.25, 200) 27/30	$H_{0,85} : \delta_{85} = 0$ $H_{1,85} : \delta_{85} > 0$	Wald	0.29×10^{-2}	0.4048 (N)	0.0833 (N)
		Score	1.05×10^{-3}	0.1459 (N)	0.0252 (Y)
		LRT	4.25×10^{-4}	0.0599 (N)	0.0153 (Y)
(7.5, 22.5) 29/30	$H_{0,42} : \delta_{42} = 0$ $H_{1,42} : \delta_{42} > 0$	Wald	0.25×10^{-2}	0.3554 (N)	0.0907 (N)
		Score	0.79×10^{-4}	0.0113 (Y)	0.0057 (Y)
		LRT	5.62×10^{-6}	0.0008 (Y)	0.0004 (Y)
(2, 400) 48/60	$H_{0,74'} : \delta_{74} = 0$ $H_{1,74'} : \delta_{74} < 0$	Wald	5.86×10^{-7}	0.0001 (Y)	0.0001 (Y)
		Score	9.44×10^{-9}	0.0000 (Y)	0.0000 (Y)
		LRT	1.32×10^{-6}	0.0004 (Y)	0.0004 (Y)

mixture points. Three types of test statistics were calculated (see Appendix A) under Hochberg adjustment and B-H procedure for family-wise error $\alpha = 0.05$ or false discovery rate $\alpha = 0.05$. P-values are approximated with the asymptotic chi-bar-squared distribution. Significant test results found by at least one type of statistic under either the Hochberg adjustment or the B-H procedure are shown in Table 6.7.

Summarizing the results of Tables 6.6 and 6.7, we make several observations. First, given a test statistic and a multiplicity adjustment method, interactive dose

combinations identified from two-sided tests are exactly the same as those identified from one-sided tests. This is not surprising but we are pleased with empirical evidence supporting the use of the chi-bar-squared distribution versus the use of chi-squared distributions. Second, the B-H procedure tends to find more significant results than the Hochberg adjustment. This is to be expected since the B-H procedure gains power by controlling the false discovery rate rather than the family-wise error rate. Third, a large portion of significant results are consistent between the score test and likelihood ratio test under the same multiplicity adjustment, whereas the Wald test only identifies one of these interactive combinations. The Wald test appears to be less powerful than the score and likelihood ratio tests when the observations are on the sample space boundary, i.e., $y_k = 0$ or 1 . And fourth, the finding that interaction is dose-dependent (obtained from Table 6.5) has been corroborated. Of the seven dose combinations listed in Tables 6.6 and 6.7, three are declared antagonistic and four are declared synergistic by at least one test statistic.

The testing results for the interaction status of each of 72 combinations with B-H procedure are shown in Figure 6.6. In addition, the interaction status of the combination i can simply be determined by the sign of $\mu_i - \mu_i^A$ without conducting any test. In other words, a positive sign implies synergistic effect and a negative sign implies antagonistic effect. This type of results can also be found in Figure 6.6.

We also implemented more traditional isobologram to illustrate our fitted model and some of findings in Table 6.7. Isoboles based on two response levels are shown in Figure 6.7: 80 percent (EC_{80}) and 96.7 percent ($EC_{96.7}$). We show isoboles obtained from the fitted interaction model as well as the additive isoboles that result from

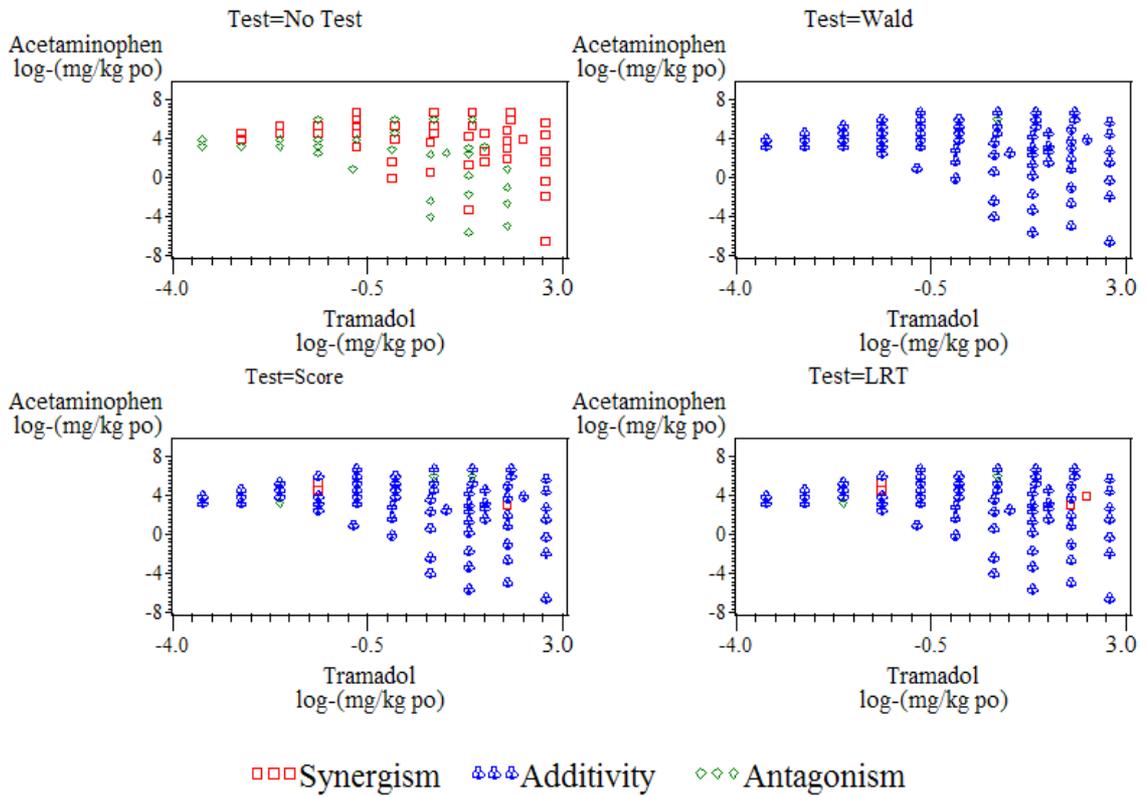


Figure 6.6: The interaction status of each of 72 combinations in TRAM and APAP data set. “Test=No Test” refers to the method of using sign of $\mu_i - \mu_i^A$. “Test=Wald” refers to the Wald one-sided tests under B-H procedure. “Test=Score” refers to the score one-sided tests under B-H procedure. “Test=LRT” refers to the likelihood ratio tests under B-H procedure. The doses are plotted in a log scale.

connecting single-agent fitted values using a straight line. It can be seen that each isobole has two segments. Antagonism is indicated by the fitted isobole exceeding the additive isobole and synergism is indicated by the fitted isobole being less than the additive isobole. Dose combinations (7.5, 22.5) and (2, 400) from Table 6.7 are also presented in the isobologram graph. The combination of (2, 400) that generates 80 percent response level is declared as an antagonistic combination by all three test statistics. In isobologram graph, this combination is far above additive isobole of EC_{80} , indicating strong antagonism. In contrast, the combination (7.5, 22.5) that generates 96.7 percent response level is far below the additive isobole of $EC_{96.7}$, implying strong synergism which is identified by both the score and likelihood ratio test statistics. Note that the two combinations do not fall on the isobles estimated from the interaction model, which indicate that our interaction model does not fit the data very well. In addition, these isoboles suggest that more dose combinations may be interactive. Unfortunately, the large number of tests under multiple adjustment may decrease the power of the test procedure to identify most of these interactive dose combinations.

Note that the Wald test results for $H_{0,45}$ vs. $H_{1,45}$ are very different from the score and likelihood ratio test. Actually, all observations that fall on the boundary of sample space, i.e., $y_k = 0$ or $y_k = 1$ will give completely unreliable Wald test results (see Appendix A).

Goodness-of-fit statistics, unfortunately indicate that our model provides a very poor fit to this data:

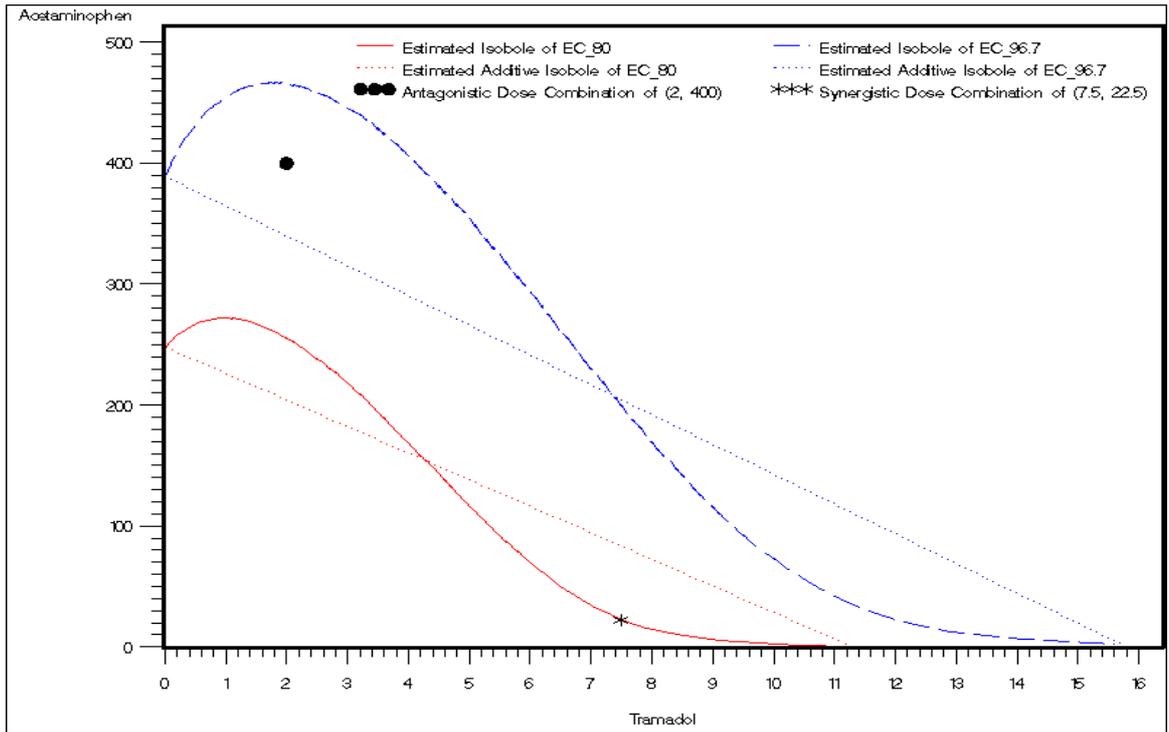


Figure 6.7: Isobologram of Tramadol and Acetaminophen.

	Statistic	Degrees of freedom	p-value
Deviance	247.79	86	< 0.001
Pearson	1116.41	86	< 0.001

A diagnostic Q-Q plot of the ordered Pearson residuals versus standard normal quantiles is given in Figure 6.8. There is a clear outlier that was further identified to be dose point (0, 800). Removing this dose point from the data reduces the Pearson statistic to 215.08 with 85 degrees of freedom and reduces the deviance to 224.98 with 85 degrees of freedom. The p-values, however, still suggest a weak model fit. Dawson et al. (2000) used two different transformations of doses to test several individual dose combinations. The results from two transformation forms were different, one of which detected no interactive dose combinations and conflicted with the patent statement that a synergistic effect occurs at certain combination points. It seems

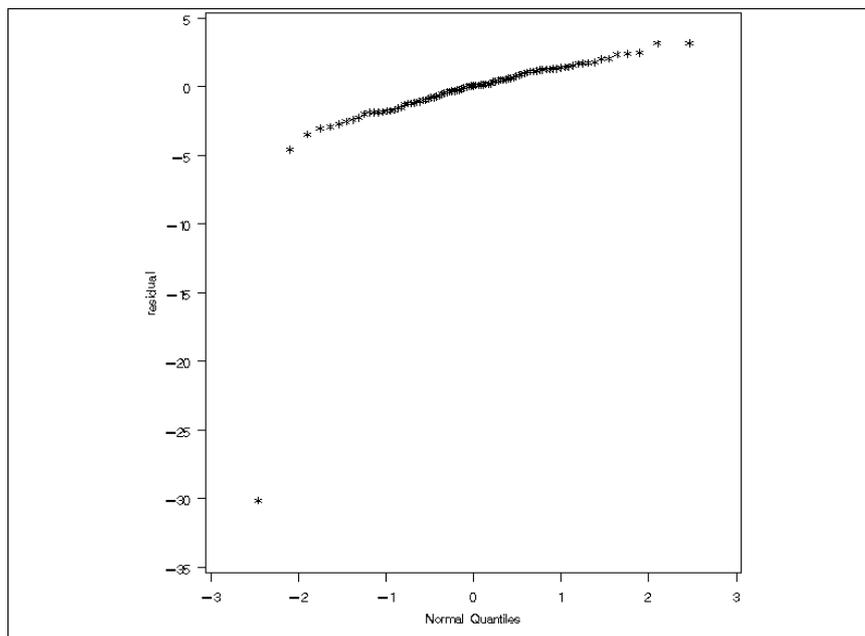


Figure 6.8: The Q-Q plot of the ordered Pearson residuals vs. the standard normal quantiles

that the model fit is a general problem for at least several dose-response models. The experimental design might play an important role here in data collection and data analysis. We briefly discuss the designs of experiment in compound mixture studies in Section 6.7. Finally, although the conclusion of dose-dependent interaction from our model seems to be consistent with the patent statement, the test results, especially those of interaction of individual dose combinations, need to be interpreted with great caution.

6.5.4 Marathion, Parathion and Piperonyl Butoxide Data Set

The data set comes from Rider and LeBlanc (2005). Two acetylcholinesterase inhibiting organophosphates, malathion and parathion, were combined with a P450 inhibitor, piperonyl butoxide, to model the toxicity. The authors declared an antagonistic effect from the tertiary mixture.

The data includes single-chemical data, binary mixture data for each pair of agents, and tertiary mixture data. All the dose combinations of the data set are shown in Figure 6.9.

Every dose point (single, binary or tertiary) was applied on a group of 20 neonatal daphnids. The number of immobilized daphnids after treatment were recorded to form the quantal response. Denote the sample proportion of immobilized daphnids in batch k as y_k . Consequently, $20y_k$ has a binomial distribution, $\text{Bin}(20, \mu_k)$, where μ_k is the expected value of y_k , i.e., the probability of an individual daphnid treated in group k being immobilized. Our interactive model with a logit link can be implemented to model all μ_k 's, i.e.,

$$\log \frac{\mu_k}{1 - \mu_k} = \beta_1 + \beta_2 x_{1k} (1 + x_{2k} + x_{3k})^{s_1} + \beta_3 x_{2k} (1 + x_{1k} + x_{3k})^{s_2} + \beta_4 x_{3k} (1 + x_{1k} + x_{2k})^{s_3},$$

where x_{1k} , x_{2k} and x_{3k} are the doses of malathion, parathion and piperonyl butoxide administered in group k , respectively. The likelihood function for the total sample of 290 observations is

$$L(\beta_1, \beta_2, \beta_3, \beta_4, s_1, s_2, s_3; y_1, \dots, y_{290}) \propto \prod_{k=1}^{290} \mu_k^{20y_k} (1 - \mu_k)^{20-20y_k}.$$

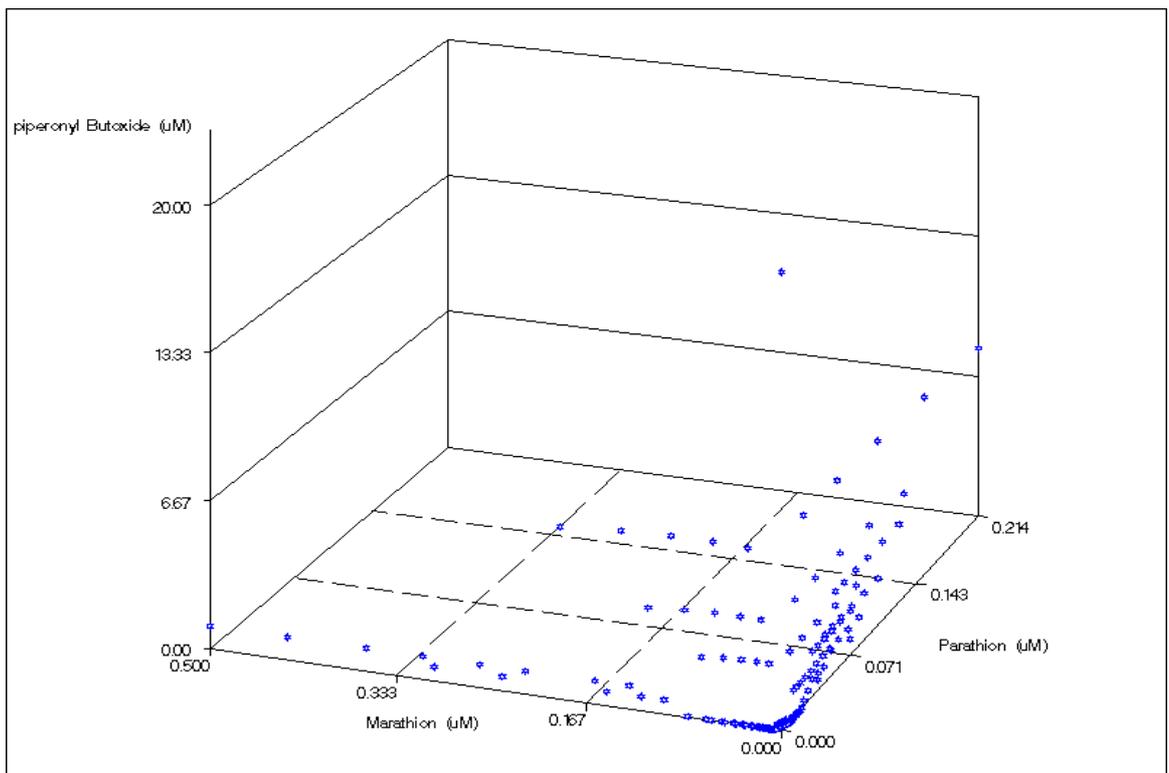


Figure 6.9: The 3-D plot of dose point included in matathion, parathion and piperonyl butoxide data set.

The estimated parameter values and the Wald standard errors are

Parameter	Estimate	Standard Error
β_1	-2.98	0.01
β_2	256.59	0.02
β_3	261.83	0.001
β_4	0.35	0.02
s_1	-92.49	4.18
s_2	-2.34	0.06
s_3	6.49	0.29

Six one-sided tests of s_1 , s_2 and s_3 were conducted simultaneously. Wald, score and likelihood ratio test statistics were calculated (see Appendix A) and asymptotic p-value's were used to approximate p-value's of three tests. Hochberg (1988) adjustment and Bonferroni adjustment were implemented separately to adjust multiplicity and to control the total family error rate at $\alpha = 0.05$. The test results are shown in Table 6.8. Note that since p-values are either 0 or 1 and it is easy to determine the rejection status of each test we do not include Q-value in Table 6.8 and 6.8.

All test results state that s_1 and s_2 are significantly less than zero and s_3 is significantly greater than zero. Therefore, uncertainty interaction is declared from testing signs of s_1 , s_2 and s_3 simultaneously.

Suppose we also have interest in testing 24 tertiary combinations (combination ID, i.e., k in our model, goes from 267 to 290) against non-additivity individually. 24 two-sided tests were conducted under the multiplicity adjustment. Three types of test statistics were calculated (see Appendix A) under Hochberg adjustment and B-H procedure separately. P-values were approximated using asymptotic chi-squared distribution.

15 tertiary combinations, whose observed y_k 's are not on the boundary of the

Table 6.8: Test results of marathion, parathion and piperonyl butoxide Data. The total number of tests adjusted for multiplicity is 6.

Hypothesis Test	Raw P-value		Significance	
			Hochberg Adjustment	Bonferroni Adjustment
$H_0^1 : s_1 = 0$ vs. $H_1^1 : s_1 > 0$	Wald	1	No	No
	Score	1	No	No
	LRT	1	No	No
$H_0^2 : s_1 = 0$ vs. $H_1^2 : s_1 < 0$	Wald	0	Yes	Yes
	Score	0	Yes	Yes
	LRT	0	Yes	Yes
$H_0^3 : s_2 = 0$ vs. $H_1^3 : s_2 > 0$	Wald	1	No	No
	Score	1	No	No
	LRT	1	No	No
$H_0^4 : s_2 = 0$ vs. $H_1^4 : s_2 < 0$	Wald	0	Yes	Yes
	Score	0	Yes	Yes
	LRT	0	Yes	Yes
$H_0^5 : s_3 = 0$ vs. $H_1^5 : s_3 > 0$	Wald	1	No	No
	Score	1	No	No
	LRT	1	No	No
$H_0^6 : s_3 = 0$ vs. $H_1^6 : s_3 < 0$	Wald	0	Yes	Yes
	Score	0	Yes	Yes
	LRT	0	Yes	Yes

sample space, i.e., $0 < y_k < 1$, are identified to be interactive by all three types of statistics under either Hochberg adjustment or under B-H procedure. All p-values of these 15 tertiary mixture points are very close to zero, which provide strong evidence of interactions. For the tertiary combinations with y_k 's on the boundary, i.e., $y_k = 0$ or $y_k = 1$, Wald statistics again provide non-sensible results that they are all non-significant. Score and Likelihood ratio tests give the same results that only combination 285 and combination 290 are non-interactive and the rest of combinations are highly interactive (p-value's are all close to 0). The test results for the tertiary combinations with y_k 's on the boundary of sample space are given in Table 6.9. Note that since p-values are either 0 or 1 and it is easy to determine the rejection status of each test we do not include Q-value in Table 6.9.

The results of one-sided tests of 24 tertiary combinations show high consistency with two-sided tests. All interactive combinations points identified by score and likelihood ratio two-sided tests are rejected in one-sided tests against $\delta < 0$ with p-values all close to 0, presenting strong evidence that they all have antagonistic effect. Only the null hypotheses of combination point 285 and combination point 290 cannot be rejected at either direction by score and likelihood tests, thus, being non-interactive. The one-sided Wald test results, again, identified the same antagonistic combination points as score and likelihood ratio tests when y_k is not on the sample space boundary. It fails to identify any antagonistic or synergistic combinations when $y_k = 0$ or $y_k = 1$ (See Appendix A). All conclusions of one-sided tests are drawn under Hochberg's adjustment and B-H procedure separately. The two multiplicity adjustment methods yield no different conclusions of significant results. The testing

Table 6.9: Two-Sided test results of tertiary combinations on sample space boundary. The total number of tests adjusted for multiplicity is 24.

(x_1, x_2, x_3) uM y_k/w_k	Hypothesis Test	Raw P-value		Significance	
				Hochberg Adjustment	B-H Procedure
(0.0181, 0.1738, 7.22) 20/20	$H_0 : \delta_{285} = 0$ $H_1 : \delta_{285} \neq 0$	Wald	1	No	No
		Score	1	No	No
		LRT	1	No	No
(0.0326, 0.1395, 6.52) 20/20	$H_0 : \delta_{290} = 0$ $H_1 : \delta_{290} \neq 0$	Wald	1	No	No
		Score	1	No	No
		LRT	1	No	No
(0.0553, 0.0066, 2.46) 0/20	$H_0 : \delta_{267} = 0$ $H_1 : \delta_{267} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0271, 0.0193, 1.81) 0/20	$H_0 : \delta_{268} = 0$ $H_1 : \delta_{268} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0190, 0.0305, 1.90) 0/20	$H_0 : \delta_{270} = 0$ $H_1 : \delta_{268} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0350, 0.0161, 2.00) 0/20	$H_0 : \delta_{271} = 0$ $H_1 : \delta_{271} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0444, 0.0119, 2.22) 0/20	$H_0 : \delta_{272} = 0$ $H_1 : \delta_{272} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0158, 0.0395, 2.11) 0/20	$H_0 : \delta_{273} = 0$ $H_1 : \delta_{273} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes
(0.0453 0.0323 3.02) 0/20	$H_0 : \delta_{276} = 0$ $H_1 : \delta_{276} \neq 0$	Wald	1	No	No
		Score	0	Yes	Yes
		LRT	0	Yes	Yes

results for the interaction status of each of 24 combinations with B-H procedure are shown in Figure 6.10. Again, the results of interaction status determined by the sign of $\mu_i - \mu_i^A$ without conducting any test are included in Figure 6.10.

The goodness-of-fit statistics, however, do not support a good fit (see test results below).

	Statistic	Degrees of freedom	p-value
Deviance	4227.92	283	< 0.001
Pearson	5512.63	283	< 0.001

A diagnostic Q-Q plot of the ordered Pearson residuals versus standard normal quantiles is given in Figure 6.11. The plot is not linear, especially at smallest and largest values of Pearson residuals. Again, our method is not adequate to model these data. A possible reason is that the data were collected from at least three different experiments. Some exploratory analysis (not presented here) shows that the data from different experiments yields significantly different estimates of EC50's of the same drug. Therefore, it seems that the impact of experimental design on data quality and data analysis results is non-negligible. We briefly discuss the designs of experiment in compound mixture studies in Section 6.7. Again, the test results of tertiary data need to be interpreted with great caution.

6.6 Simulation Study

In order to check the power of our proposed testing procedures, we conducted a simulation study for the binary mixtures with quantal responses.

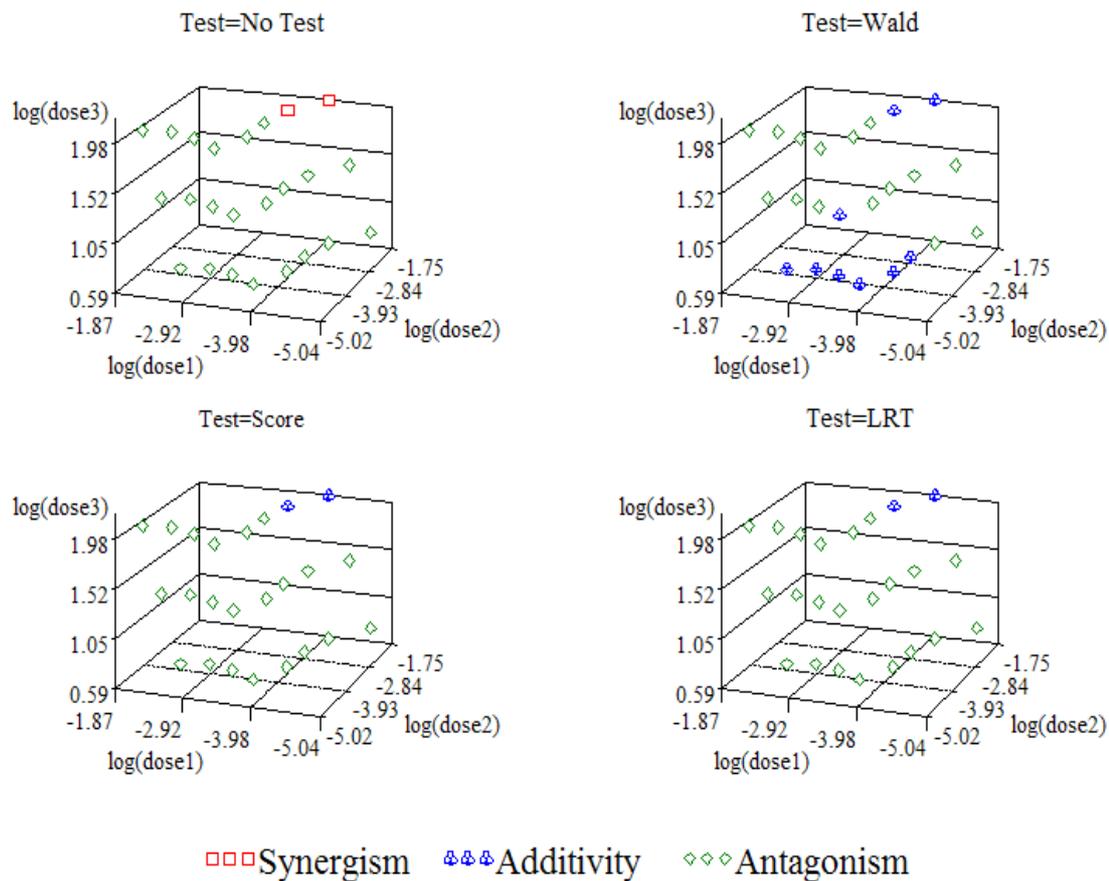


Figure 6.10: The interaction status of each of 24 combinations in marathion, parathion and piperonyl butoxide data set. “Test=No Test” refers to the method of using sign of $\mu_i - \mu_i^A$. “Test=Wald” refers to the Wald one-sided tests under B-H procedure. “Test=Score” refers to the score one-sided tests under B-H procedure. “Test=LRT” refers to the likelihood ratio tests under B-H procedure. Dose1 refers to the dose of marathion; dose2 refers to the dose of parathion; dose3 refers to the dose of piperonyl butoxide. The unit of doses is μM . The doses are plotted in a log scale.

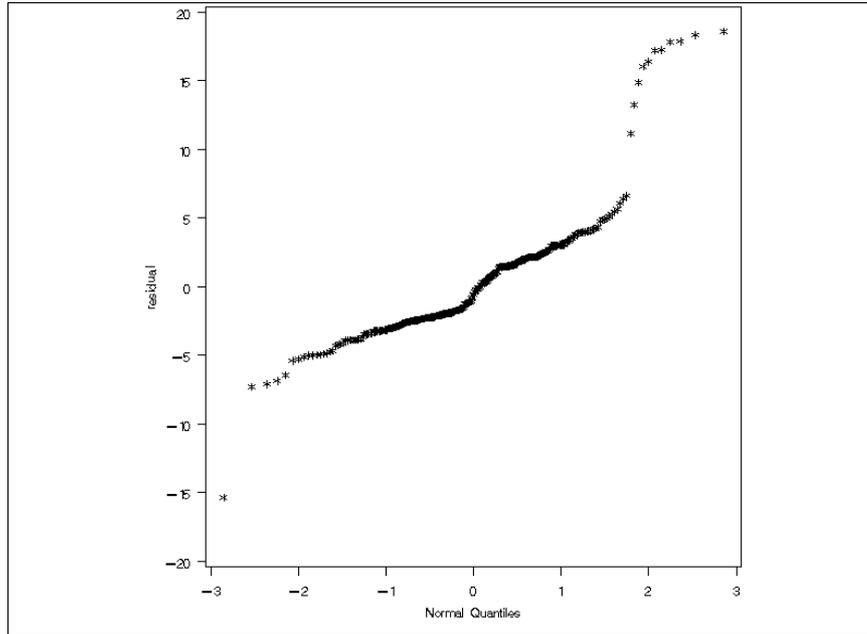


Figure 6.11: The Q-Q plot of the ordered Pearson residuals vs. the standard normal quantiles

The parameter estimates of β_1 , β_2 , β_3 , s_1 , and s_2 (Table 6.10) using the tramadol (TRAM) and acetaminophen (APAP) data set were obtained from PROC NLMIXED (SAS, Inc., 2001). Using estimates of β_1 , β_2 , and β_3 and different combinations of (s_1, s_2) as the truth, we simulated overall synergistic effects, overall antagonistic effects and dose-dependent interactions.

Table 6.10: Estimates of Interactive Model for Tramadol and Acetaminophen Data Set

Parameter	Estimates
β_1	-2.02
β_2	0.30
β_3	0.01
s_1	0.11
s_2	-0.32

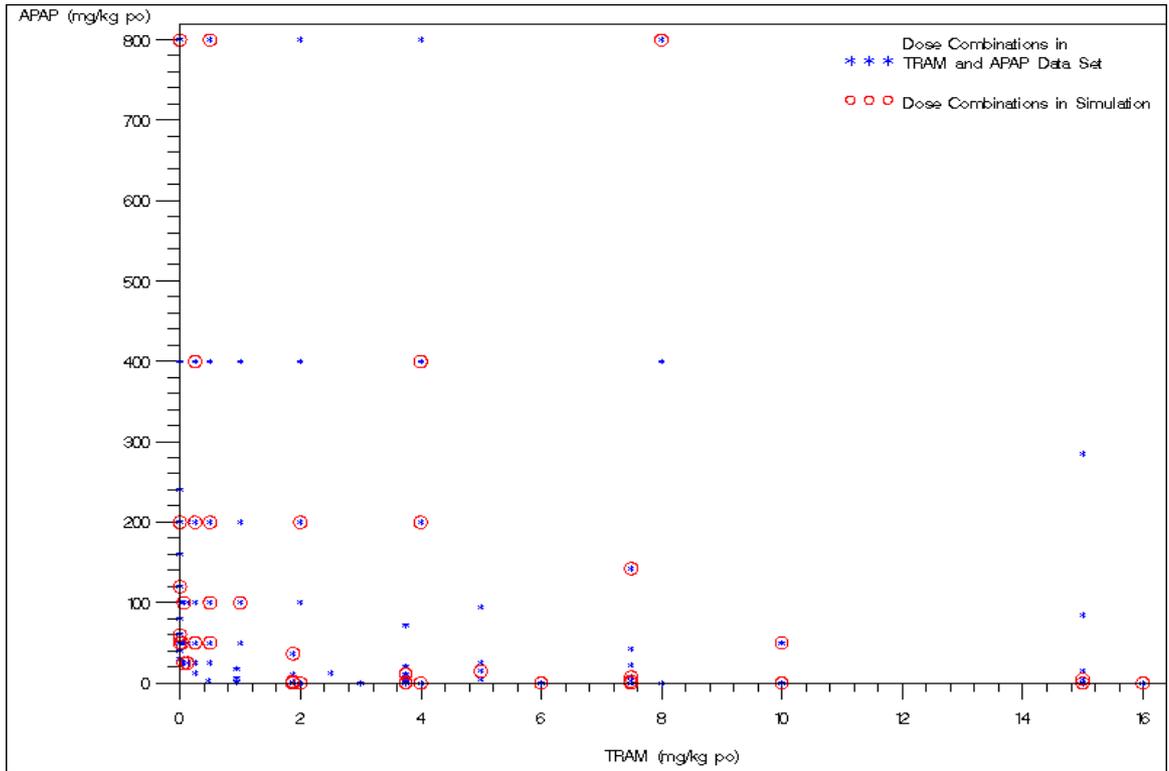


Figure 6.12: The scatter plot of the dose of APAP versus the dose of TRAM.

More specifically, five single-agent doses of TRAM, five single-agent doses of APAP and 30 binary dose combinations of TRAM and APAP were randomly selected from the TRAM and APAP data set (See Figure 6.12). The interaction model, that generated the expected proportion of responders at each dose point k , $k = 1, \dots, 40$, is

$$\mu_k = \frac{1}{1 + \exp(-(-2.0208 + 0.2978x_{1k}(1 + x_{2k})^{s_1} + 0.0127x_{2k}(1 + x_{1k})^{s_2}))}.$$

Consequently, the responses at dose point k , $w_k y_k$, could be generated from a binomial distribution, $\text{Bin}(w_k, \mu_k)$, where y_k is the sample proportion of responders at dose point k and w_k is the total number of independent replicates at dose point k . In our

simulation study, w_k is assumed to be the same over all dose points.

A Monte Carlo (MC) method was implemented to estimate the power of each test statistic for a fixed combination of (s_1, s_2) and type I error of the tests. Given a combination of (s_1, s_2) , (y_1, \dots, y_{40}) at 40 dose points were simulated as described above. Three types of statistics testing the signs of s_1 and s_2 were calculated after our interaction model was fitted with the simulated data. The testing conclusion of overall effect from each test statistic under Hochberg adjustment was recorded, separately. The same process of simulating (y_1, \dots, y_{40}) , fitting our model, calculating test statistics and recording test results, was repeated for $N_{MC} = 1000$ times. For a given pair of (s_1, s_2) , the proportion of times when correct conclusions were drawn is the MC estimated power. The MC-estimated type I error is calculated in the same way with $(s_1, s_2) = (0, 0)$. In addition, a Wald-type 95% confidence interval of each MC estimated power was calculated. The interval has the general form

$$\left(\widehat{P}_{MC} - 1.96 \sqrt{\frac{\widehat{P}_{MC}(1 - \widehat{P}_{MC})}{N_{MC}}}, \widehat{P}_{MC} + 1.96 \sqrt{\frac{\widehat{P}_{MC}(1 - \widehat{P}_{MC})}{N_{MC}}} \right),$$

where \widehat{P}_{MC} is the MC estimated power.

We checked the type I error first, i.e., the probability that at least one hypothesis of sign of s_1 or s_2 was rejected when $s_1 = 0$ and $s_2 = 0$, using MC method with $N_{MC} = 1000$. In the real TRAP and APAP data, w_k varies from 15 to 60 over 91 different dose points. Here, we set w_k to 60 for $k = 1, \dots, 40$. The results are shown in Table 6.11.

The overall synergistic effect was then simulated by setting $s_1 > 0$ and $s_2 > 0$. The value of s_1 was fixed at 0.1 and the value of s_2 was changed from 0.1 to 0.9.

Table 6.11: Estimated Type I Error with $N_{MC} = 1000$

w_k ($k = 1, \dots, 40$)	Type I Error		
	Score	Wald	LRT
60	0.041	0.046	0.043
50	0.043	0.043	0.045
40	0.039	0.051	0.039
30	0.043	0.046	0.045

Consequently, the value of s_2 reflected the magnitude of synergism. In the real TRAP and APAP data, w_k varies from 15 to 60 over 91 different dose points. In order to study the relationship between the power and the magnitude of synergism, we set w_k to 60 for $k = 1, \dots, 40$. The power curves of three types of statistics are overlaid in the Figure 6.13. The corresponding confidence intervals of power estimates are presented as vertical lines with tops and bottoms.

Similarly, the overall antagonistic effect was simulated by setting $s_1 < 0$ and $s_2 < 0$. The value of s_2 was fixed at -0.3 and the value of s_1 varied within $[-0.1, -0.001]$ to reflect different magnitudes of antagonism of the mixture. Again, w_k was assumed to be 60 for $k = 1, \dots, 40$ and N_{MC} was set to 1000. The MC estimated power curves of three statistics with corresponding confidence intervals are given in Figure 6.14.

Dose-dependent interaction was also investigated for three types of statistics. Let $w_k = 60$ for $k = 1, \dots, 40$. The combinations of (s_1, s_2) have a general form $(0.1 + 0.1\Delta, -0.3 + \Delta)$ with Δ changing between 0 and 0.5. The increment rates of s_1 and s_2 are set different to be consistent the difference between β_2 and β_3 . The corresponding MC power estimates were calculated after 1000 iterations. Figure 6.15 presents the power curves for the dose-dependent interaction case.

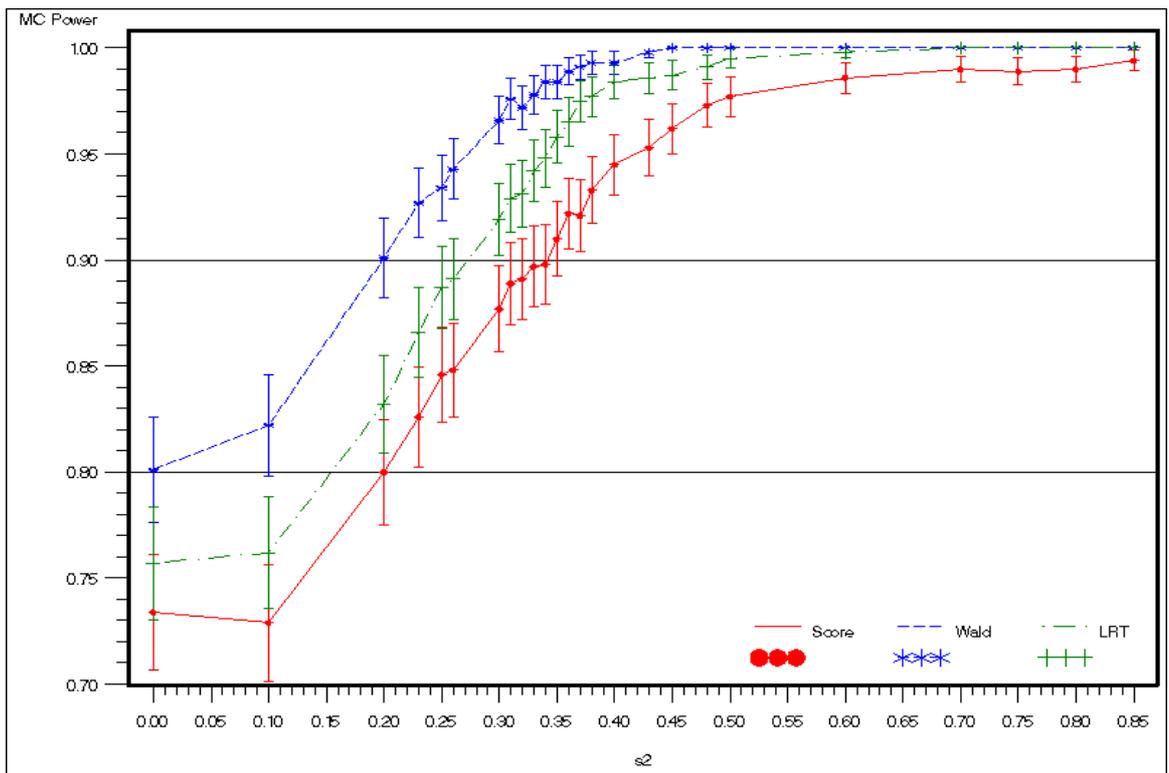


Figure 6.13: The plot of MC estimated power versus s_2 with $N_{MC} = 1000$, $s_1 = 0.1$ and $w_k = 60$.

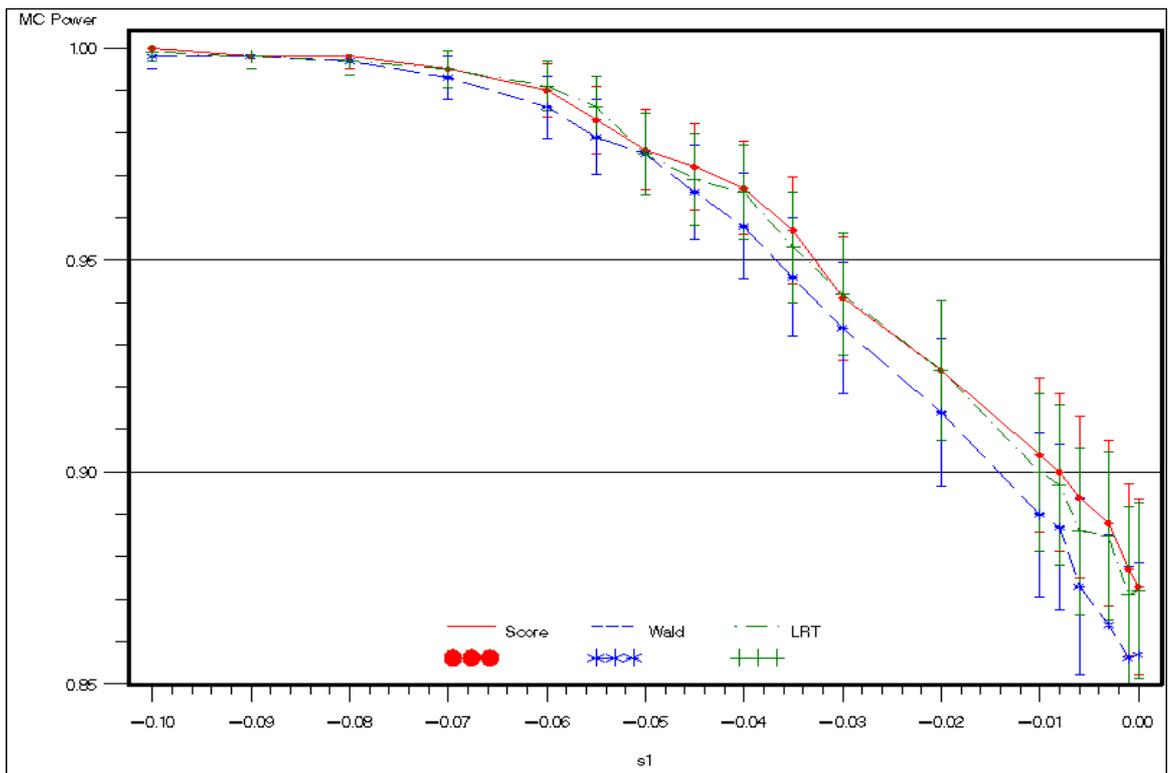


Figure 6.14: The plot of MC estimated power versus s_1 with $N_{MC} = 1000$, $s_2 = -0.3$ and $w_k = 60$.

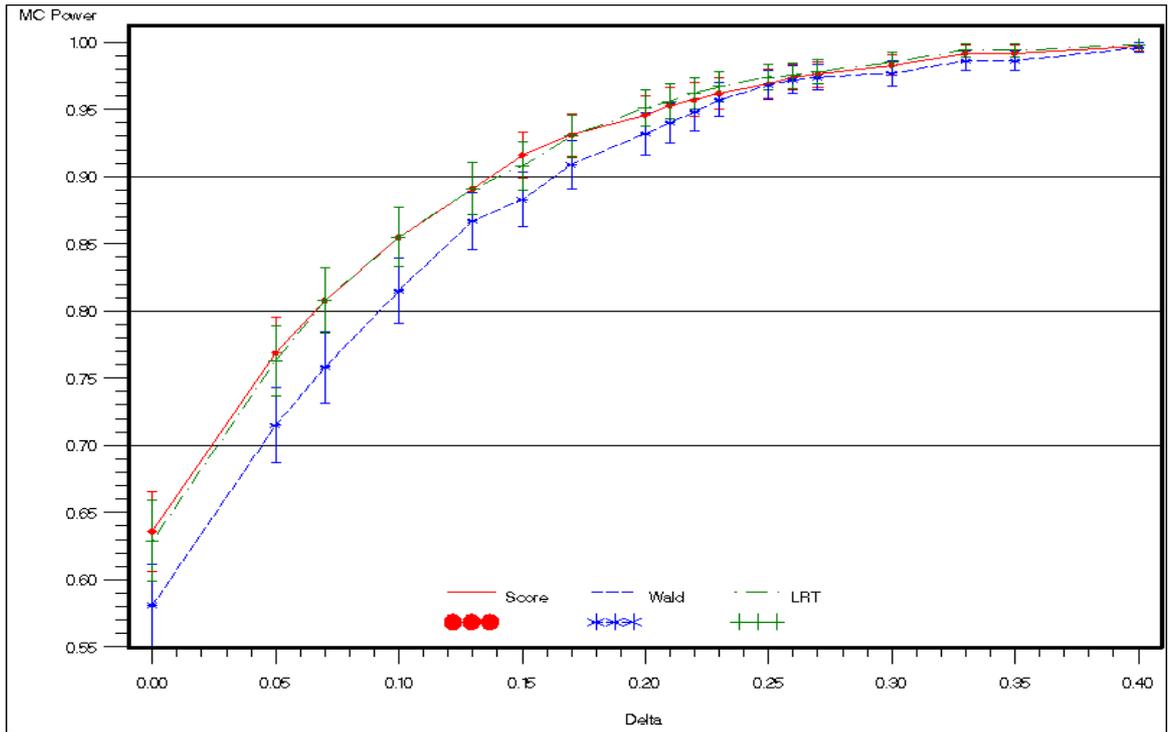


Figure 6.15: The plot of MC estimated power versus Δ with $N_{MC} = 1000$, and $w_k = 60$.

It can be seen that the power of three types of statistics are quite close to each other for overall antagonism and dose-dependent interaction. The confidence intervals of the three statistics are overlap each other. The likelihood ratio test statistic has a larger power than score test statistics and a smaller power than Wald statistic for the overall synergism case. The type I error of the three statistics are generally less than 0.05 with score statistic slightly more conservative than the other two. Our results are very consistent with the findings (Boos and Stefanski, pp. 56-57 and p. 61) of these three types of statistics for two-sided tests. In general, the two-sided score test statistic is smaller and more conservative than the other two statistics when asymptotic critical values are used. Also, Brendt and Savin (1977) proved that the score statistic is no greater than the likelihood ratio statistic, and the Wald statistic

is no smaller than the likelihood ratio statistic for the normal linear model.

The relationship between sample size at each dose point, i.e., w_k and MC estimated power was examined for overall synergism, overall antagonism, and dose-dependent interaction. The (s_1, s_2) was fixed at $(0.1, 0.8)$ to simulate overall synergism since the estimated power of three test statistics are all close to 1 when $w_k = 60$. (s_1, s_2) was specified as $(-0.06, -0.3)$ and $(0.135, -0.65)$ to generate overall antagonism and dose-dependent interaction, respectively, for a similar reason. The results are shown in Figure 6.16, Figure 6.17 and Figure 6.18.

As expected, the power of tests decreases when sample size decreases. For the overall synergism in our simulation, the Wald statistic is more robust to sample size decrement. The likelihood ratio test statistic performs worse than the Wald statistic but better than the score test statistic. For the overall antagonism and dose-dependent interaction, three types of statistics yielded quite similar power curves with Wald statistic being slightly less powerful.

Another interesting question is that how powerful our testing procedure would be to determine a synergistic/antagonistic effect for an individual combination. An obvious fact is that the power of all methods will decrease dramatically, when the number of hypotheses is relatively large. This is the cost of multiplicity control. In practice, the main interest might be focused on detecting one or a few combinations that are most significantly synergistic or antagonistic. For example, the optimal dose combination to be used in a large confirmatory clinical trail might be selected only from several of the most synergistic combinations otherwise it might be difficult to

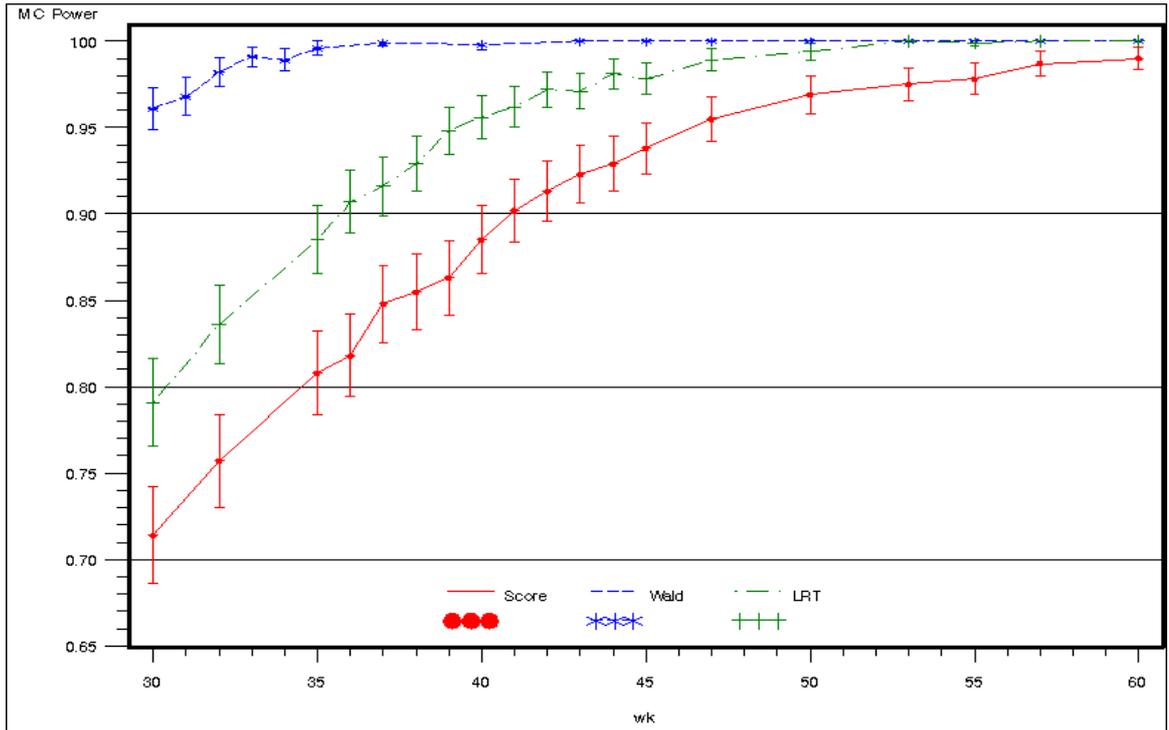


Figure 6.16: The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = 0.1$ and $s_2 = 0.8$.

commercialize.

Therefore, the issue of power could be alternatively addressed by studying the power of detecting the most synergistic/antagonistic combination(s). In drug development, clinicians generally have a good idea of a clinically meaningful synergistic/antagonistic effect such that statisticians could use this quantity to define the most interactive combination(s) to be detected and calculate the power based on a fixed sample size. In our simulation study, we borrowed the idea from the suggestions of the assessment of bioequivalence from U.S. FDA (Chow and Liu 2000, pp. 20-22). U.S. FDA suggests that “bioequivalence should be concluded if the average bioavailability of the test formulation is within (80%, 125%) of that the reference formulation with a certain assurance” (Chow and Liu 2000, pp.20). Although the response in

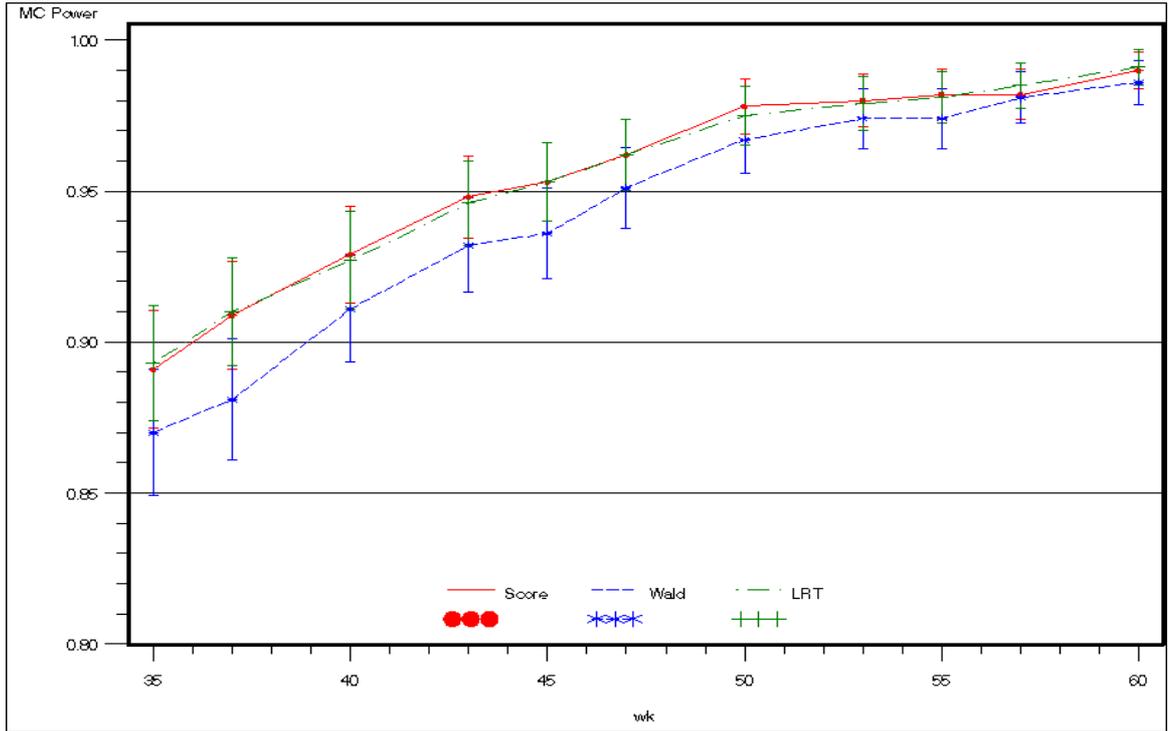


Figure 6.17: The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = -0.06$ and $s_2 = -0.3$.

our simulation is not a parameter that can assess bioavailability of the drug we still used this so-called “80/125 rule” to define our significantly antagonistic/synergistic combinations.

More specifically, the additive μ_k^a calculated by setting $s_1 = 0$ and $s_2 = 0$ serves as the reference response of dose combination k in “80/125 rule”. The interactive μ_k defined from our interactive model with a given combination of (s_1, s_2) is the true response of dose combination k . Under the fact μ_k and μ_k^a are both bounded by $(0,1)$ if the ratio of μ_k to μ_k^a is less than 80% or greater than 125% the dose combination k is defined to be significantly antagonistic or significantly synergistic, respectively. Consequently, these dose combinations are the most interactive combinations in our simulations and we will focus on examining the power of tests to detect those combi-

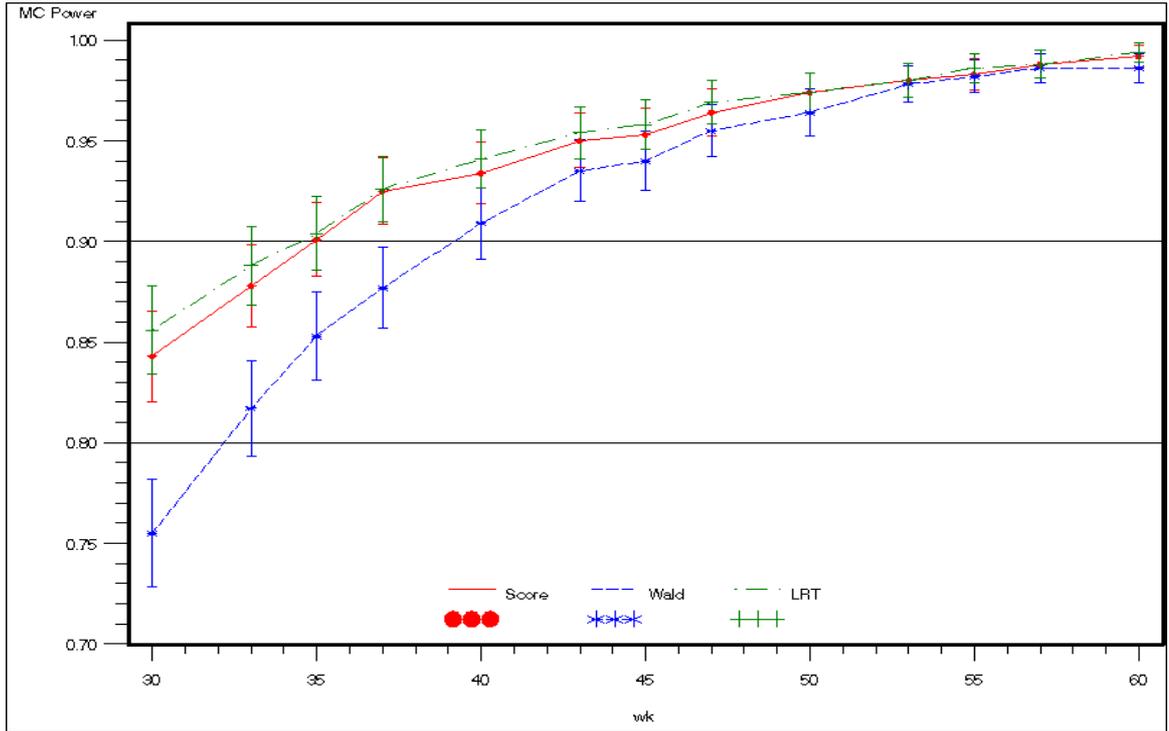


Figure 6.18: The plot of MC estimated power versus w_k with $N_{MC} = 1000$, $s_1 = 0.135$ and $s_2 = -0.65$.

nations.

Given a fixed pair of (s_1, s_2) 30 binary combinations from 40 dose points selected before could be classified into three categories with “80/125 rule”: significantly synergistic combinations, significantly antagonistic combinations, and non-significantly interactive combinations. $w_k = 60$ was set for $k = 1, \dots, 40$. The MC estimated power of three one-sided test statistics was checked with $N_{MC} = 100$ due to the long computational time and limited memory of the PC. Please note that the one-sided tests of individual combinations will be conducted in each iteration of our MC simulation without first testing the signs of s_1 and s_2 since we treat these two testing procedures as two independent methods in our methodology development. Two multiplicity adjustment methods were checked separately. The results of estimated MC power with

Table 6.12: Estimated MC power for one-sided tests of the most synergistic/antagonistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.123$ and $s_2 = -0.53$. The total number of tests adjusted for multiplicity is 30.

(x_{1k}, x_{2k})	μ_k/μ_k^a	Test Statistic	Multiple Adjustment	
			Hochberg	B-H
(5,15)	1.29	Wald	0.14	0.49
		Score	0.16	0.20
		LRT	0.16	0.50

different (s_1, s_2) are given in Table 6.12, Table 6.13 and Table 6.14.

Some general conclusions could be summarized from three tables. First, the power of testing individual combinations can be fairly small due to the multiplicity adjustment for the relatively large number of tests. For example, the setting of (s_1, s_2) and w_k in Table 6.12, under which all three statistics for testing the signs of s_1 and s_2 have power greater than 0.95, only yield the power less than 0.5 of detecting the most synergistic combination from 30 binary combinations. Second, the B-H procedure can dramatically increase the power compared to Hochberg adjustment since it controls the false discovery error rate. Third, the power of the score test statistic in our example is less than that of the other two statistics. This could be a general situation since the score test statistic is generally smaller and more conservative in controlling type I error than the other two. In our simulation, the observed values of the significantly synergistic/antagonistic combinations are generally not on the boundary of the sample space. If the observed value is on the boundary of sample space (see Appendix A) the Wald statistic tends to be unreliable. Therefore, the LRT test statistic combined with the B-H procedure is recommended to use for testing individual combinations according to our simulation.

Table 6.13: Estimated MC power for one-sided tests of the most synergistic/antagonistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.2$ and $s_2 = -1.4$. The total number of tests adjusted for multiplicity is 30.

(x_1k, x_2k)	μ_k/μ_k^a	Test Statistic	Multiple Adjustment	
			Hochberg	B-H
(5,15)	1.55	Wald	0.64	0.97
		Score	0.68	0.86
		LRT	0.67	0.97
(3.75,11.25)	1.44	Wald	0.17	0.60
		Score	0.23	0.40
		LRT	0.17	0.60
(7.5,7.5)	1.40	Wald	0.57	0.95
		Score	0.63	0.82
		LRT	0.67	0.95
(0.25,200)	0.80	Wald	0.12	0.43
		Score	0.14	0.28
		LRT	0.12	0.44
(2,200)	0.75	Wald	0.32	0.79
		Score	0.37	0.55
		LRT	0.32	0.79
(0.5,200)	0.70	Wald	0.24	0.77
		Score	0.28	0.58
		LRT	0.27	0.77

Table 6.14: Estimated MC power for one-sided tests of the most synergistic/antagonistic individual combination(s) with $N_{MC} = 100$, $s_1 = 0.2$ and $s_2 = -2$. The total number of tests adjusted for multiplicity is 30.

(x_1k, x_2k)	μ_k/μ_k^a	Test Statistic	Multiple Adjustment	
			Hochberg	B-H
(0.5,200)	1.54	Wald	0.62	0.96
		Score	0.67	0.88
		LRT	0.66	0.96
(0.5,200)	1.44	Wald	0.19	0.60
		Score	0.20	0.42
		LRT	0.19	0.60
(0.5,200)	1.39	Wald	0.57	0.95
		Score	0.62	0.87
		LRT	0.64	0.95
(0.25,200)	0.72	Wald	0.03	0.42
		Score	0.04	0.22
		LRT	0.05	0.43
(3.75,11.25)	0.72	Wald	0.07	0.41
		Score	0.08	0.15
		LRT	0.09	0.42
(5,15)	0.71	Wald	0.24	0.73
		Score	0.27	0.48
		LRT	0.27	0.74
(2,200)	0.65	Wald	0.62	0.95
		Score	0.64	0.84
		LRT	0.63	0.95
(7.5,7.5)	0.58	Wald	0.66	0.94
		Score	0.68	0.85
		LRT	0.67	0.94

Finally, we set $(\beta_1, \beta_2, \beta_3, s_1, s_2)$ as the estimated values in Table 6.10. (x_{1k}, x_{2k}) , and w_k , $k = 1, \dots, 91$ in TRAM and APAP data set were all used in simulation. Again, μ_k was calculated from our interaction model and y_k was generated from binomial distributions, $\text{Bin}(w_k, \mu_k)$, $k = 1, \dots, 91$. The estimated power for testing signs of s_1 and s_2 with $N_{MC} = 1000$ MC samples is given as

Statistic	Power (%)
Wald	75.7
Score	73.2
LRT	75.3

In addition, the power for testing individual combinations was estimated with $N_{MC} = 100$. Four individual combinations could be classified as significantly synergistic dose points using “80/125 rule”. Unfortunately, the estimated power of detecting any of them is less than 0.05. The weak power may be explained by small magnitudes of synergism and multiplicity adjustment. Note that three of four synergistic dose points in our simulation are not identified as synergistic dose combinations in the results of TRAM and APAP data (See Table 6.6 and Table 6.7). This can be explained by the weak fit of our interaction model in TRAM and APAP data set. In other words, the simulated “true” response data in simulation is based on the fitted interaction model which does not accurately describe the TRAM and APAP data. Consequently, the interactive combinations determined in real TRAM and APAP data set cannot be confirmed in simulation study. Note that this does not necessarily mean that the model fit in simulated data is also poor. In contrast, the model generally fits the simulated data very well. For example, the goodness-of-fit statistics for the first MC sample are shown below.

	Statistic	Degrees of freedom	p-value
Deviance	63.92	86	0.96
Pearson	60.96	86	0.98

This suggests an excellent model fit.

6.7 Design of Experiments

Though the design of experiments is not the main interest in our research, it is worth briefly discussing the important role of experimental designs in compound mixture studies. A lot of literature can be found on mixture design examples (e.g., Greco et al. 1995; Gennings 1995; Meadow et al. 2002; Price et al., 2002). Several factors, such as selection of dose combinations, the number of experimental animals, and different sources of variation other than dose combinations should be considered when planning a mixture study.

Selection of dose combinations gains most statistical consideration since it is impossible to test all dose combinations in a study. The possible designs include full and fractional factorial designs, ray designs, and multi-stage designs.

The simplest strategy to determine the interaction effects between two particular compounds is using a 2×2 factorial design, where four combinations of two drugs are administered to four groups of experimental animals. This strategy focuses on whether the combination of two exposures yields greater or lesser than additive effects compared with the single exposures. This simple factorial design can allow the model in row four of Table 6.1 to test the sign of β_3 , i.e., overall interaction of the binary

mixture. Brunden et al. (Chapter 2, 1988) discussed how to obtain four optimal dose combinations in this design under different optimality criteria such as minimizing the variance of the estimator of β_3 or minimizing the determinant of variance-covariance matrix of estimator of $(\beta_1, \beta_2, \beta_3)^T$. In addition, 3×3 and 4×4 factorial designs can also be commonly seen in binary mixture studies. The majority of full factorial designs focuses on 2^M factorial designs when a mixture contains M ($M > 3$) compounds, where 2^M means each of M compound will be tested at two dose levels and in total 2^M possible combinations will be tested. Even though only two dose levels are selected for each compound in the 2^M factorial design, this type of design can still be infeasible with M increasing due to a large number of dose combinations required by the design. For example, if the mixture contains six compounds, a 2^M factorial design implies 64 different dose combinations as well as several replications at each combination. When the number of compounds is large, the fractional factorial designs, e.g., a 2^{k-1} design where a half fraction of dose combinations is selected, can be used as an alternative starting point to explore the possible interaction (see Groten et al. 1997 for an example).

Ray designs, initially proposed by Mantel (1958) and then discussed by Brunden et al. (1988) and Meadows et al. (2002), are widely used to study a mixture of more than three components. For a mixture of M components, a “ray” consists of all dose combinations with a fixed mixing ratio, $r_1 : r_2 : \dots : r_M$, where $\sum_{i=1}^M r_i = 1$. For a binary mixture, each ray is just a straight line emanating from the dose point (0,0) with a specific slope. Figures 6.2 and 6.4 clearly shows that both of the studies implement ray designs. The main advantage of ray designs is that the dose

combinations along a particular ray can be regarded as a unique, single chemical with only total dose varying. Consequently, any dose-response modeling strategy for a single compound can be applied to analyze the data collected from the same ray. This will also allow us to test overall interaction along a particular ray (see Meadows et al. 2002 for further details). If a design contains several rays, a conclusion of interaction over entire dose range can be drawn by testing interaction of different rays simultaneously.

A design (subsequently referred to as multi-stage design) that combines the factorial and ray designs takes several steps to detect the interaction. Tajima et al. (2002) proposed multi-stage design to detect interactive effects among several classes of mycotoxins. Three stages were included in the design. In the first stage (detection stage), the whole mixture of five mycotoxins along a fixed-ratio ray was used with the total dose increasing. In addition, the dose-response relationship of each individual component was measured. The interaction among five mycotoxins was tested along the ray and the antagonism was detected at the high total dose level. Consequently, the stage two (screening stage) involved a fractional factorial design to screen interaction of possible pairs of mycotoxins. Finally, two pairs of mycotoxins that were found with synergistic effects in the stage two were separately studied in the stage three (confirmation stage) using a full factorial design.

In addition to dose selection, the number of experimental animals or sometimes referred to as sample size is of critical importance due to the ethical implications of using animals in research as well as the constraints of the cost and the resource. As we have seen in our simulation studies, sample size is highly related with the statistical

hypothesis test you will conduct after the data collection. Given a formal statistical hypothesis and a well-defined test statistic, the sample size can be determined by the desired power and the magnitude of interaction of interest. In general, simulation studies are needed to determine the sample size.

Some common issues in experimental designs should also be taken care of in compound mixture studies. For example, inclusion of all relevant factors as well as some uncontrollable systematic variation through the measurement of covariates is important. For the tertiary data set in real application we got different estimates of EC50's of the same drug in the different experiments. This might be explained by the protocol difference or different moving ability of neonate daphnids at baseline. These factors should be measured and analyzed in the model. Sometimes there are unknown or unmeasured sources of variation that can bias the results as well. Therefore, randomly assigning animals to different dose combinations is essential since the randomization can help ensure that all dose combinations have equal chance of being affected by the unknown or unmeasured sources. Different estimates of EC50's of the same drug in different experiments in tertiary study might also be explained by some unknown or unmeasured sources of variation, such as the edge effects in plates or different local environment of the incubators. A good randomization schedule before running the whole study can help solve this problem. In addition, blocking is also a common design technique that can help reduce excess variation and group more homogeneous experimental units together. For example, we might conduct all the experimental measurements during one week if the week-to-week variation is known. Or we might want to use the same laboratory to do the study and repeat the study in different

laboratories if we think there is variation of different laboratories.

Good experimental designs help investigators well understand the interaction nature of the mixtures. The design issues discussed here are not intended to be comprehensive. They represent a starting guidance of planning a compound mixture study. Different issues might be considered under different objectives.

6.8 Discussion and Summary

We propose an interaction model that introduces a new multiplier to a GLM-type additive model. The mathematical properties of the multiplier enable the model to flexibly accommodate additivity as well as different types of interactions that include overall synergism, overall antagonism, and dose-dependent interaction. Different testing procedures are developed to either test the interaction over the entire dose range or to test the interaction status at a particular dose combinations. The model requires no prior knowledge of action mechanisms of the mixtures or activity mechanisms of individual chemicals. The model and the testing procedures is initially developed for a binary mixture and can be easily generalized to a mixture of M components, where $M > 2$. The results from the real applications and the simulation studies suggest that the model is generally applicable to a wide range of mixtures with different types of interactions. The testing conclusions of overall interaction are highly consistent with the previous experiments or other analysis results. The weak model fits, however, were observed for some real data sets. As we discussed before, it might be worth considering other critical explanatory variables (e.g., protocol difference) in

our dose-response interaction model to improve the model fit.

As expected, the power of testing procedures is affected by the magnitude of interaction, sample size, the method of multiplicity adjustment, and the number of tests in a multiple testing context. In general, testing the overall interaction is more powerful than testing the interaction of individual dose combinations given the same data. According to the results from simulation studies, the tests of overall interaction is recommended when the number of individual combinations is large and the magnitude of interaction on each combination is expected to be moderate. In general, our interaction model combined with the testing procedures can serve as a good exploratory tool for compound mixture studies.

Bibliography

- [1] Aarts, E., and Korst, J. (1989), *Simulated Annealing and Boltzmann Machine*, Chichester: Wiley.
- [2] Ashford, J. R. and Cobby, J. M. (1979), "A system of models for the action of drugs applied singly and jointly to biological organisms." *Biometrics*, 30, 11.
- [3] Bedrick, E. J., Christensen, R., and Johnson W. (1996), "A new perspective on priors for generalized linear models." *Journal of the American Statistical Association*, 91, 1450-1460.
- [4] Berenbaum, M. C. (1989), "What is synergy?" *Pharmacological Reviews*, 41, 93-141.
- [5] Bliss, C. I. (1939), "The toxicity of poisons applied jointly." *Annual Applied Biology*, 26, 585-615.
- [6] Blower, P., Fligner, M., Verducci J., and Bjoraker, J. (2002), "On Combining Recursive Partitioning and Simulated Annealing to Detect Groups of Biologically Active Compounds," *Journal of Chemical Information and Computer Sciences*, 42, 393-404.
- [7] Boos, D. D. and Stefanski, L. A. (2004), *Modern Statistical Inference: Theory*

and Methods (Course Notes), Raleigh.

- [8] Brieman, L. (1998), "Arcing Classifiers (with discussion and a rejoinder by the author)," *Annals of Statistics*, 26, 801-849.
- [9] Breiman, L. (2001), "Random Forests," *Machine Learning*, 45, 5-32.
- [10] Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and Regression Trees*, New York: Chapman and Hall.
- [11] Berndt, E. and Savin, N. E., (1977), "Conflict among criteria for testing hypotheses in the multivariate linear regression model," *Econometrica*, 45, 1263-177.
- [12] Brown, R. D., and Martin, Y. C. (1996), "Use of Structure-Activity Data to Compare Structure-Based Clustering Methods and Descriptors For Use in Compound Selection," *Journal of Chemical Information and Computer Sciences*, 36, 572-584.
- [13] Brunden, M. N., Vidmar, T. J., and McKean, J. W. (1988), *Drug Interaction and Lethality Analysis*, Boca Raton: CRC Press, Inc.
- [14] Carhart, R. E., Smith, D. H., and Venkataraghavan, R. (1985), "Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications," *Journal of Chemical Information and Computer Sciences*, 25, 64-73.
- [15] Casella, G., and Berger, R. L. (2002), *Statistical Inference*, 2nd ed., Belmont: Duxbury Press.
- [16] Chen, X., Rusinko, A., Tropsha, A., and Young, S. S. (1999), "Automated Pharmacophore Identification for Large Chemical Data Sets," *Journal of Chemical Information and Computer Sciences*, 38, 1054-1062.

- [17] Chib, S., and Greenberg, E. (1995), "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, 49, 327-335.
- [18] Cho, S. J., Shen, C. F., and Hermsmeier, M. A. (2000), "Binary Formal Inference-Based Recursive Modeling Using Multiple Atom and Physicochemical Property Class Pair and Torsion Descriptors as Decision Criteria," *Journal of Chemical Information and Computer Sciences*, 41, 168-175.
- [19] Chou, T-C. and Talalay, P. (1981), "Generalized equations for the analysis of Michaelis-Menten and higher-order kinetic systems with two or more mutually exclusive and non-exclusive inhibitors." *European Journal of Biochemistry*, 115, 207-216.
- [20] Chou, T-C. and Talalay, P. (1984), "Quantitative analysis of dose-effect relationships: the combined effects of multiple drugs or enzyme inhibitors." *Advances in Enzyme Regulation*, 22, 27-55.
- [21] Dawson, K. S., Carter, W. H., and Gennings, C. (2000), "A Statistical Test for Detecting and Characterizing Departures from Additivity in Drug/Chemical Combinations." *Journal of Agriculture, Biological, and Environmental Statistics*, 5, 342-359.
- [22] Finney, D. J. (1942), "The analysis of toxicity tests on mixtures of poisons." *Annual Applied Biology*, 29,82-94.
- [23] Finney, D. J. (1952), *Probit Analysis (third edition)*, Cambridge: Cambridge University Press.
- [24] Gaddum, J. H. (1957), "Theories of drug antagonism." *Pharmacological Reviews*,

9(2), 211-218.

- [25] Gennings, C. (1995), "An efficient experimental design for detecting departure from additivity in mixtures of many chemicals." *Toxicology* 105: 189-197.
- [26] Gessner, P. K. (1988), "A straightforward method for the study of drug interactions: an isobolographic analysis primer." *Journal of the American College of Toxicology*, 7, 987-1012.
- [27] Gessner, P. K. (1995), "Isobolographic analysis of interactions: an update on applications and utility." *Toxicology*, 105(2-3), 161-179.
- [28] Goldin, A., Venditti, J. M., Humphreys, S.R., and Dennis, D. (1955), "An investigation into the synergism antileukemic action of amethopterin and 6-mercaptopurine in mice." *Journal of National Cancer Institute*, 16, 132.
- [29] Grabovsky, Y. and Tallarida, R. J. (2004) "Isobolographic analysis for combinations of a full and partial agonist: curved isoboles." *The Journal of Pharmacology and Experimental Therapeutics*, 310, 981-986.
- [30] Greco, W., Unkelbach, H. D., Pösch, G., Sühnel, J., Kundi, M., and Boedeker, W. (1992), "Consensus on concepts and terminology for combined action assessment: The Saariselkä agreement." *Archives of Complex Environmental Studies*, 4, 65-69.
- [31] Greco, W., Bravo, G., and Parsons J. G. (1995), "The search for synergy: a critical review from a response surface perspective." *Pharmacological Reviews*, 47, 331-385.
- [32] Groten, J. P., Schoen, E. D., VanBladeren, P. J., Kuper, C. F., VanZorge, J. A.,

- and Feron, V. J. (1997), "Subacute toxicity of a mixture of nine chemicals in rats: detecting interactive effects with a fractionated two-level factorial design." *Fundamental and Applied Toxicology*, 36, 1529.
- [33] Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning Data Mining, Inference, and Prediction*, New York: Springer.
- [34] Hawkins, D. M. and Kass, G. V. (1982), "Automatic Interaction Detection," In *Topics in Applied Multivariate Analysis*, ed. D. M. Hawkins, Cambridge, U.K.: Cambridge University Press. pp. 269-302.
- [35] Hewlett, P. S. and Plackett, R. L. (1959), "A unified theory for quantal responses to mixtures of drugs: non-interactive action" *Biometrics* 20, 591-610.
- [36] Hochberg, Y. (1988), "A Sharper Bonferroni Procedure for Multiple Tests of Significance," *Biometrika*, 75, 800-802.
- [37] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983), "Optimization by Simulated Annealing." *Science*, 220, 67-78.
- [38] Kudo, A. (1963), "A Multivariate Analogue of the One-Sided Test." *Biometrika*, 50, 403-418.
- [39] Lambert, C. G. (2004), Partitionator Analysis Software, Golden Helix, Inc., <http://www.goldenhelix.com> (last accessed December 9, 2004).
- [40] Lazareno S. and Birdsall, N. J. M. (1993), "Estimation of competitive antagonist affinity from functional inhibition curves using the Gaddum, Schild and Cheng-Prusoff equations." *Journal of British Pharmacology*, 109, 1110-1119.
- [41] Loewe, S. (1953), "The problem of synergism and antagonism of combined

- drugs.” *Arzneimittel Forschung*, 3, 285-290.
- [42] Loewe, S. (1957), “Antagonisms and antagonists.” *Pharmacological Reviews*, 9, 237-243.
- [43] Liu, J., Feng, J., and Young, S. S. (2004), “PowerMV: A Software Environment for Statistical Analysis, Molecular Viewing, Descriptor Generation, and Similarity Search,” <http://www.niss.org/PowerMV> (last accessed December 9, 2004).
- [44] Mantel N. (1958), “An experimental design in combination chemotherapy.” *Annals of the New York Academy of Sciences*, 76, 909.
- [45] Maycock, A. L., Abeles, R. H., Salach, J. I., and Singer, T. P. (1976), “The Structure of the Covalent Adduct Formed by the Interaction of 3-(dimethylamino)-1-propyne And the Flavine of Mitochondrial Amine Oxidase.” *Biochemistry*, 15, 114-125.
- [46] Meadows, S. L., Gennings, C., Carter, W. H., and Bae, D-S. (2002), “Experimental Designs for Mixtures of Chemicals along Fixed Ratio Rays.” *Environmental Health Perspectives*, 110, 979-983.
- [47] McFarland, J. W. and Gans, D. J. (1986), “On the Significance of Clusters in the Graphical Display of Structure-Activity Data.” *Journal of Medicinal Chemistry*, 29, 505-514.
- [48] Miller, R. G., Jr. (1981), *Simultaneous Statistical Inference*, 2nd edition, New York: Springer-Verlag.
- [49] Motulsky H. and Christopoulos A. (2004), *Fitting Models to Biological Data*

Using Linear and Nonlinear Regression: A Practical Guide to Curve Fitting,
New York: Oxford University Press.

- [50] Muska, S. D. and Weber, L. J. (1977), *An approach for studying the effects of mixtures of environmental toxicants in whole organism performance*. In: *Recent advances in fish toxicology*. Taub, RA, ed. EPA/600/3-77/085, pp. 71-87.
- [51] Neto, J. C. and Paula, G. A. (2001), “Wald One-Sided Test Using Generalized Estimating Equations Approach.” *Computational Statistics and Data Analysis*, 36, 475-495.
- [52] Paula, G. A. (1999), “One-Sided Tests in Generalized Linear Dose-Response Models.” *Computational Statistics and Data Analysis*, 30, 413-427.
- [53] Plackett, R. L. and Hewlett, P. S. (1951), “Quantal responses to mixtures of poisons.” *Journal of the Royal Statistical Society: Series B*, 14, 141-163.
- [54] Plackett, R. L. and Hewlett, P. S. (1967), “A comparison of two approaches to the construction of models for quantal responses to mixtures of drugs.” *Biometrics* 23, 27-43.
- [55] Price, B., Borgert, C. J., Wells, C. S. and Simon, G. S. (2002), “Assessing toxicity of mixtures: the search for economical study designs.” *Human and Ecological Risk Assessment*, 8, 305-326.
- [56] R Development Core Team. (2003), *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-00-3, URL [http:// www.R-project.org](http://www.R-project.org). (last accessed De-

cember 9, 2004)

- [57] Raffa, R. B., Stone, D. J., and Tallarida, R. J. (1999) “Antinociceptive self synergy between spinal and supraspinal acetaminophen (paracetamol).” *International Association for the Study of Pain*, Austria.
- [58] Raffa, R. B., Stone, D. J., and Tallarida, R. J. (2000) “Discovery of self-synergistic spinal/supraspinal antinociception produced by acetaminophen (paracetamol).” *The Journal of Pharmacology and Experimental Therapeutics*, 295, 291-294.
- [59] Rider, C. V. and LeBlanc, G. A. (2005), “ An integrated addition and interaction model for assessing toxicity of chemical mixtures.” *Toxicological Sciences*, 87, 520 - 528.
- [60] Robertson, T., Wright, F. T., and Dykstra, R. L. (1988), *Order Restricted Statistical Inference*, New York: John Wiley.
- [61] Rusinko, A., Farmen, M. W., Lambert, C. G., Brown, P. L., and Young, S. S. (1999), “Analysis of a Large Structure/Biological Activity Data Set Using Recursive Partitioning.” *Journal of Chemical Information and Computer Sciences*, 39, 1017-1026.
- [62] SAS Institute, Inc. (1999a), *Getting Started with Enterprise Miner Software, Version 3.0*, Cary: SAS Institute, Inc.
- [63] SAS Institute, Inc. (1999b), *SAS/IML User’s Guide, Version 8*, Cary: SAS Institute, Inc.
- [64] SAS Institute, Inc. (2000), *SAS/STAT User’s Guide, Version 8*, Cary: SAS

Institute, Inc.

- [65] Schild, H. O. (1957), "Drug antagonism and pax." *Pharmacological Reviews*, 9(2), 242-246.
- [66] Shapiro, A. (1985), "Asymptotic Distribution of Test Statistics in the Analysis of Moment Structures Under Inequality Constraints." *Biometrika*, 72(1), 133-144.
- [67] Shapiro, A. (1988), "Towards a Unified Theory of Inequality Constrained Testing in Multivariate Analysis." *Interactional Statistical Review*, 56, 49-62.
- [68] Silvapulle, M. J., and Silvapulle, P. (1995), "A score Test Against One-Sided Alternatives." *Journal of the American Statistical Association*, 90, 342-349.
- [69] Silverman, L., Campbell, R., and Broach, J. R. (1998), "New Assay Technologies for High Throughput Screening," *Current Opinion in Chemical Biology*, 2, 397-403.
- [70] Sittampalam, G. S., Kahl, S. D. and Janzen, W. P. (1997), "High-throughput Screening: Advances in Assay Technologies," *Current Opinion in Chemical Biology*, 1, 384-391.
- [71] Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R., and Feuston, B. P. (2003), "Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling," *Journal of Chemical Information and Computer Sciences*, 43, 1947-1958.
- [72] Tajima, O., Schoen, E. D., Feron, V. J., Groten, J. P. (2002), "Statistically designed experiments in a tiered approach to screen mixtures of Fusarium

- mycotoxins for possible interactions.” *Food and Chemical Toxicology*, 40, 685-695.
- [73] Tallarida, R. J. (2000), *Drug Synergism and Dose-Effect Data Analysis*, Boca Raton: Chapman & Hall/CRC Press.
- [74] Tallarida, R. J. (2001), “Drug synergism: its detection and applications.” *The Journal of Pharmacology and Experimental Therapeutics*, 298, 865-872.
- [75] Tallarida, R. J., Porreca, F., and Cowan, A., (1989) “Statistical Analysis of Drug-Drug and Site-Site Interactions with Isobolograms.” *Life Science*, 45, 947-961.
- [76] Wolfe, J. D., Tashkin, D. P., Calvarese, B., and Simmons, M. (1978), “Bronchodilator effects of terbutaline and aminophylline alone and in combination in asthmatic patients.” *New England Journal of Medicine*, 298, 363-367.
- [77] Young, S. S., and Hawkins, D. M. (1995), ”Analysis of a 2⁹ Full Factorial Chemical Library,” *Journal of Medicinal Chemistry*, 38, 2784-2788.

Appendices

Appendix A

Test Statistics in Studies with Quantal Responses

A.1 Likelihood Function

A general modeling framework can be set up for chemical mixture studies with quantal responses. Suppose a data set from a mixture of M components contains K different dose points (including single-agent doses, two-agent dose combinations, ..., M -agent dose combinations). x_{1k}, \dots, x_{Mk} are the doses of the M components in dose point k . Suppose $(w_k y_k)$ responders are observed from w_k , where w_k is the total number of replicates treated at dose point k and y_k is the proportion of responders at dose point k . Therefore, $w_k y_k$ has a binomial distribution, $\text{Bin}(w_k, \mu_k)$, where μ_k is the probability that a subject treated at dose point k will respond. Notice that the expected value of y_k , μ_k , is independent of w_k . Consequently, μ_k can be described by

our interaction model with a logit link; that is

$$\log\left(\frac{\mu_k}{1-\mu_k}\right) = \beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk},$$

where

$$h_{mk} = \left(1 + \sum_{j=1, j \neq m}^M x_{jk}\right)^{s_m}, m = 1, \dots, M, k = 1, \dots, K.$$

Under the assumption that all subjects are independent, the log-likelihood function is

$$\begin{aligned} & \log\{L(\beta_1, \dots, \beta_{M+1}, s_1, \dots, s_M; y_1, \dots, y_K)\} \\ &= \sum_{k=1}^K \left\{ w_k y_k \log\left(\frac{\mu_k}{1-\mu_k}\right) - w_k \log\left(\frac{1}{1-\mu_k}\right) \right\} + \text{Constant}, \end{aligned}$$

where

$$\mu_k = \frac{1}{1 + \exp(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}))}, k = 1, \dots, K.$$

A.2 Hypothesis Test of $H_0 : s_1 = 0$ vs. $H_1 : s_1 > 0$

Without loss of generality, we calculate three types of one-sided test statistics of $H_0 : s_1 = 0$ vs. $H_1 : s_1 > 0$. Define $\boldsymbol{\theta} = (s_1, \dots, s_M, \beta_1, \dots, \beta_{M+1})^T$. The score function is

$$S(\boldsymbol{\theta}) = \frac{\partial \log(L)}{\partial \boldsymbol{\theta}} = \begin{pmatrix} S_{s_1}(\boldsymbol{\theta}) \\ \dots \\ S_{s_M}(\boldsymbol{\theta}) \\ S_{\beta_1}(\boldsymbol{\theta}) \\ S_{\beta_2}(\boldsymbol{\theta}) \\ \dots \\ S_{\beta_{M+1}}(\boldsymbol{\theta}) \end{pmatrix} = \begin{pmatrix} \frac{\partial \log(L)}{\partial s_1} \\ \dots \\ \frac{\partial \log(L)}{\partial s_M} \\ \frac{\partial \log(L)}{\partial \beta_1} \\ \frac{\partial \log(L)}{\partial \beta_2} \\ \dots \\ \frac{\partial \log(L)}{\partial \beta_{M+1}} \end{pmatrix},$$

where

$$\frac{\partial \log(L)}{\partial s_m} = \sum_{k=1}^K U_k(\boldsymbol{\theta}) E_k(\boldsymbol{\theta}) \beta_{m+1} x_{mk} h_{mk} \log(h_{mk}^{-s_m}), m = 1, \dots, M;$$

$$\begin{aligned}
\frac{\partial \log(L)}{\partial \beta_1} &= \sum_{k=1}^K U_k(\boldsymbol{\theta}) E_k(\boldsymbol{\theta}); \\
\frac{\partial \log(L)}{\partial \beta_{m+1}} &= \sum_{k=1}^K U_k(\boldsymbol{\theta}) E_k(\boldsymbol{\theta}) x_{mk} h_{mk}, m = 1, \dots, M; \\
U_k(\boldsymbol{\theta}) &= \frac{w_k y_k - w_k \mu_k}{\mu_k (1 - \mu_k)}, k = 1, \dots, K; \\
E_k(\boldsymbol{\theta}) &= \frac{\exp\left\{-\left(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}\right)\right\}}{\left\{1 + \exp\left(-\left(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}\right)\right)\right\}^2} = \mu_k (1 - \mu_k), k = 1, \dots, K; \\
\mu_k &= \frac{1}{1 + \exp\left\{-\left(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}\right)\right\}}, k = 1, \dots, K; \\
h_{mk} &= \left(1 + \sum_{j=1, j \neq m}^M x_{jk}\right)^{s_m}, m = 1, \dots, M, k = 1, \dots, K.
\end{aligned}$$

The information matrix is

$$\begin{aligned}
I(\boldsymbol{\theta}) &= K I_K(\boldsymbol{\theta}) = E \left[-\frac{\partial^2 \log(L)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right] \\
&= -E \left[\begin{pmatrix} \frac{\partial^2 \log(L)}{\partial s_1 \partial s_1} & \cdots & \frac{\partial^2 \log(L)}{\partial s_1 \partial s_M} & \frac{\partial^2 \log(L)}{\partial \beta_1 \partial s_1} & \frac{\partial^2 \log(L)}{\partial s_1 \partial \beta_2} & \cdots & \frac{\partial^2 \log(L)}{\partial s_1 \partial \beta_{M+1}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial s_1} & \cdots & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial s_M} & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_1} & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_2} & \cdots & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_{M+1}} \end{pmatrix} \right]
\end{aligned}$$

where

$$E \left[-\frac{\partial^2 \log(L)}{\partial s_i \partial s_j} \right] = \sum_{k=1}^K \left\{ U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta}) \beta_{i+1} x_{ik} h_{ik} \log(h_{ik}^{-s_i}) \beta_{j+1} x_{jk} h_{jk} \log(h_{jk}^{-s_j}) \right\},$$

$i, j = 1, \dots, M;$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_{i+1} \partial \beta_{j+1}} \right] = \sum_{k=1}^K \left\{ U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta}) x_{ik} h_{ik} x_{jk} h_{jk} \right\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial s_i \partial \beta_{j+1}} \right] = \sum_{k=1}^K \left\{ U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta}) \beta_{i+1} x_{ik} h_{ik} \log(h_{ik}^{-s_i}) x_{jk} h_{jk} \right\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial \beta_1} \right] = \sum_{k=1}^K \{U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta})\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial \beta_{m+1}} \right] = \sum_{k=1}^K \{U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta}) x_{mk} h_{mk}\}, m = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial s_m} \right] = \sum_{k=1}^K \{U'_k(\boldsymbol{\theta}) E'_k(\boldsymbol{\theta}) \beta_{m+1} x_{mk} h_{mk} \log(h_{mk}^{-s_m})\}, m = 1, \dots, M;$$

$$U'_k(\boldsymbol{\theta}) = \frac{w_k}{\mu_k(1 - \mu_k)}, k = 1, \dots, K;$$

$$E'_k(\boldsymbol{\theta}) = \left\{ \frac{\exp \left\{ -(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}) \right\}}{\left[1 + \exp(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk})) \right]^2} \right\}^2 = (\mu_k(1 - \mu_k))^2, k = 1, \dots, K.$$

As described in Section 6.3.2 we could calculate three types of statistics based on the score function and information matrix. The maximum likelihood estimates under H_0 and under $H_0 \cup H_1$ can be obtained using PROC NLMIXED/SAS. Calculation of test statistics can also be programmed with IML/SAS (See Appendix B).

A.3 Hypothesis Tests of Individual Combinations

A.3.1 Likelihood function

If we are interested in determining the interaction status of a particular combination, say combination 1, the two-sided test can be constructed as

$$H_{0,1,two} : \mu_1 = \mu_1^A \quad \text{vs.} \quad H_{1,1,two} : \mu_1 < \mu_1^A,$$

where

$$\mu_1^A = \frac{1}{1 + \exp \left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{m1}) \right)}$$

In addition, two one-sided tests could be developed as

$$H_{0,1} : \mu_1 = \mu_1^A \quad \text{vs.} \quad H_{1,1} : \mu_1 > \mu_1^A$$

and

$$H_{0,1'} : \mu_1 = \mu_1^A \quad \text{vs.} \quad H_{1,1'} : \mu_1 < \mu_1^A$$

to test if the response is synergistic or antagonistic.

As described in Section 6.3.3, our interaction model with logit link is used to describe the expected responses of all dose points but combination 1. For combination 1, we only assume that $w_1 y_1$ has a binomial distribution, $\text{Bin}(w_1, \mu_1)$, where $0 \leq \mu_1 \leq 1$. Since all observations are independent, the log-likelihood function is

$$\begin{aligned} & \log\{L(\mu_1, \beta_1, \dots, \beta_{M+1}, s_1, \dots, s_M; y_1, \dots, y_K)\} \\ &= w_1 y_1 \log\left(\frac{\mu_1}{1-\mu_1}\right) - w_1 \log\left(\frac{1}{1-\mu_1}\right) \\ &+ \sum_{k=2}^K \left\{ w_k y_k \log\left(\frac{\mu_k}{1-\mu_k}\right) - w_k \log\left(\frac{1}{1-\mu_k}\right) \right\} + \text{Constant}, \end{aligned}$$

where

$$\mu_k = \frac{1}{1 + \exp(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk}))}, k = 2, \dots, K.$$

A.3.2 Reparameterization

The likelihood function is reparameterized before three test statistics are calculated. Define

$$\mu_1 = \frac{1}{1 + \exp(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{m1} + \delta_1))}.$$

Let $\boldsymbol{\gamma} = (\gamma_1 = \delta_1, \boldsymbol{\gamma}_2 = (\beta_1, \dots, \beta_{M+1}, s_1, \dots, s_M))^T$. The reparameterized likelihood function can be obtained as

$$\begin{aligned} & \log\{L(\boldsymbol{\gamma}; y_1, \dots, y_K)\} \\ &= w_1 y_1 \log\left(\frac{\mu_1}{1-\mu_1}\right) - w_1 \log\left(\frac{1}{1-\mu_1}\right) \\ &+ \sum_{k=2}^K \left\{ w_k y_k \log\left(\frac{\mu_k}{1-\mu_k}\right) - w_k \log\left(\frac{1}{1-\mu_k}\right) \right\} + \text{Constant}, \end{aligned}$$

where

$$\begin{aligned} \mu_1 &= \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{m1} + \delta_1)\right)}, \\ \mu_k &= \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1} x_{mk} h_{mk})\right)}, k = 2, \dots, K, \end{aligned}$$

and

$$h_{mk} = \left(1 + \sum_{j=1, j \neq m}^M x_{jk}\right)^{s_m}, m = 1, \dots, M, k = 1, \dots, K.$$

Under the reparameterization the two-sided hypothesis test for testing the interaction status of combination 1 is given as

$$H_{0,1,two} : \delta_1 = 0 \quad \text{vs.} \quad H_{1,1,two} : \delta_1 \neq 0.$$

In addition, two one-sided tests for testing the synergistic or antagonistic effect of combination 1 could be re-constructed as

$$H_{0,1} : \delta_1 = 0 \quad \text{vs.} \quad H_{1,1} : \delta_1 > 0$$

and

$$H_{0,1'} : \delta_1 = 0 \quad \text{vs.} \quad H_{1,1'} : \delta_1 < 0.$$

For reparameterized model, the score function is

$$S(\boldsymbol{\gamma}) = \frac{\partial \log(L)}{\partial \boldsymbol{\gamma}} = \begin{pmatrix} S_{\delta_1}(\boldsymbol{\gamma}) \\ S_{s_1}(\boldsymbol{\gamma}) \\ \dots \\ S_{s_M}(\boldsymbol{\gamma}) \\ S_{\beta_1}(\boldsymbol{\gamma}) \\ S_{\beta_2}(\boldsymbol{\gamma}) \\ \dots \\ S_{\beta_{M+1}}(\boldsymbol{\gamma}) \end{pmatrix} = \begin{pmatrix} \frac{\partial \log(L)}{\partial \delta_1} \\ \frac{\partial \log(L)}{\partial s_1} \\ \dots \\ \frac{\partial \log(L)}{\partial s_M} \\ \frac{\partial \log(L)}{\partial \beta_1} \\ \frac{\partial \log(L)}{\partial \beta_2} \\ \dots \\ \frac{\partial \log(L)}{\partial \beta_{M+1}} \end{pmatrix},$$

where

$$\begin{aligned} \frac{\partial \log(L)}{\partial \delta_1} &= U_1(\boldsymbol{\gamma})E_1(\boldsymbol{\gamma}); \\ \frac{\partial \log(L)}{\partial s_m} &= \sum_{k=2}^K U_k(\boldsymbol{\gamma})E_k(\boldsymbol{\gamma})\beta_{m+1}x_{mk}h_{mk} \log(h_{mk}^{-s_m}), m = 2, \dots, M; \\ \frac{\partial \log(L)}{\partial \beta_1} &= \sum_{k=1}^K U_k(\boldsymbol{\gamma})E_k(\boldsymbol{\gamma}); \\ \frac{\partial \log(L)}{\partial \beta_{m+1}} &= U_1(\boldsymbol{\gamma})E_1(\boldsymbol{\gamma})x_{m1} + \sum_{k=2}^K U_k(\boldsymbol{\gamma})E_k(\boldsymbol{\gamma})x_{mk}h_{mk}, m = 1, \dots, M; \\ U_k(\boldsymbol{\gamma}) &= \frac{w_k y_k - w_k \mu_k}{\mu_k(1 - \mu_k)}, k = 1, \dots, K; \\ E_k(\boldsymbol{\gamma}) &= \mu_k(1 - \mu_k), k = 1, \dots, K; \\ \mu_1 &= \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_{m1} + \delta_1)\right)}; \\ \mu_k &= \frac{1}{1 + \exp\left\{-\left(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_{mk}h_{mk}\right)\right\}}, k = 2, \dots, K; \\ h_{mk} &= \left(1 + \sum_{j=1, j \neq m}^M x_{jk}\right)^{s_m}, m = 1, \dots, M, k = 1, \dots, K. \end{aligned}$$

The information matrix is

$$I(\boldsymbol{\gamma}) = KI_K(\boldsymbol{\gamma}) = E \left[-\frac{\partial^2 \log(L)}{\partial \boldsymbol{\gamma} \partial \boldsymbol{\gamma}^T} \right]$$

$$= -E \left[\begin{pmatrix} \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \delta_1} & \cdots & \frac{\partial^2 \log(L)}{\partial \delta_1 \partial s_M} & \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \beta_1} & \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \beta_2} & \cdots & \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \beta_{M+1}} \\ \frac{\partial^2 \log(L)}{\partial s_1 \partial \delta_1} & \cdots & \frac{\partial^2 \log(L)}{\partial s_1 \partial s_M} & \frac{\partial^2 \log(L)}{\partial s_1 \partial \beta_1} & \frac{\partial^2 \log(L)}{\partial s_1 \partial \beta_2} & \cdots & \frac{\partial^2 \log(L)}{\partial s_1 \partial \beta_{M+1}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \delta_1} & \cdots & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial s_M} & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_1} & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_2} & \cdots & \frac{\partial^2 \log(L)}{\partial \beta_{M+1} \partial \beta_{M+1}} \end{pmatrix} \right]$$

where

$$E \left[-\frac{\partial^2 \log(L)}{\partial \delta_1 \partial \delta_1} \right] = U'_1(\boldsymbol{\gamma}) E'_1(\boldsymbol{\gamma});$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \delta_1 \partial s_m} \right] = 0, m = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \delta_1 \partial \beta_1} \right] = U'_1(\boldsymbol{\gamma}) E'_1(\boldsymbol{\gamma});$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \delta_1 \partial \beta_{m+1}} \right] = U'_1(\boldsymbol{\gamma}) E'_1(\boldsymbol{\gamma}) x_{m+1,1}, m = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial s_i \partial s_j} \right] = \sum_{k=2}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) \beta_{i+1} x_{ik} h_{ik} \log(h_{ik}^{-s_i}) \beta_{j+1} x_{jk} h_{jk} \log(h_{jk}^{-s_j}) \right\},$$

$i, j = 1, \dots, M;$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_{i+1} \partial \beta_{j+1}} \right] = U'_1(\boldsymbol{\gamma}) E'_1(\boldsymbol{\gamma}) x_{i1} x_{j1} + \sum_{k=2}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) x_{ik} h_{ik} x_{jk} h_{jk} \right\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial s_i \partial \beta_{j+1}} \right] = \sum_{k=2}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) \beta_{i+1} x_{ik} h_{ik} \log(h_{ik}^{-s_i}) x_{jk} h_{jk} \right\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial \beta_1} \right] = \sum_{k=1}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) \right\}, i, j = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial \beta_{m+1}} \right] = U'_1(\boldsymbol{\gamma}) E'_1(\boldsymbol{\gamma}) x_{m1} + \sum_{k=2}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) x_{mk} h_{mk} \right\}, m = 1, \dots, M;$$

$$E \left[-\frac{\partial^2 \log(L)}{\partial \beta_1 \partial s_m} \right] = \sum_{k=2}^K \left\{ U'_k(\boldsymbol{\gamma}) E'_k(\boldsymbol{\gamma}) \beta_{m+1} x_{mk} h_{mk} \log(h_{mk}^{-s_m}) \right\}, m = 1, \dots, M;$$

$$U'_k(\boldsymbol{\gamma}) = \frac{w_k}{\mu_k(1 - \mu_k)}, k = 1, \dots, K;$$

$$E'_k(\gamma) = (\mu_k(1 - \mu_k))^2, k = 1, \dots, K;$$

$$\mu_1 = \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_{m1} + \delta_1)\right)};$$

$$\mu_k = \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_{mk}h_{mk})\right)}, k = 2, \dots, K.$$

$$h_{mk} = \left(1 + \sum_{j=1, j \neq m}^M x_{jk}\right)^{s_m}, m = 1, \dots, M, k = 1, \dots, K.$$

As described in Section 6.3.3, we could calculate three test statistics for the two-sided test. They all have an asymptotic Chi-squared distribution. Similarly, we could calculate three test statistics for one-sided tests. They all have an asymptotic Chi-bar-squared distribution. The maximum likelihood estimates can be obtained by PROC NLMIXED/SAS and the calculation of different types of statistics can be programmed in IML/SAS (See Appendixe B).

A.3.3 Wald Statistics with Observations on Sample Space Boundary

Let $\boldsymbol{\lambda} = (\lambda_1 = \mu_1, \boldsymbol{\lambda}_2 = (\beta_1, \dots, \beta_{M+1}, s_1, \dots, s_M))^T$ and

$$J = \mu_1 - \frac{1}{1 + \exp\left(-(\beta_1 + \sum_{m=1}^M \beta_{m+1}x_{m1})\right)}.$$

We first focus on the two-sided tests. Without doing reparameterization, a general form of the Wald statistics based on the likelihood function (6.11) is

$$T^W = \frac{(J(\hat{\boldsymbol{\lambda}}) - 0)^2}{J'(\hat{\boldsymbol{\lambda}})^T (KI_K(\hat{\boldsymbol{\lambda}}))^{-1} J'(\hat{\boldsymbol{\lambda}})},$$

where $J'(\boldsymbol{\lambda}) = \frac{\partial J(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}}$, $\hat{\boldsymbol{\lambda}}$ is the maximum likelihood estimator. Since μ_1 is only related to y_1 's density and observations are independent, the $KI_K(\boldsymbol{\lambda})$ can be written as

$$KI_K(\boldsymbol{\lambda}) = -E \left[\begin{pmatrix} \frac{\partial^2 \log(L)}{\partial \mu_1 \partial \mu_1} & \mathbf{0}_{2M+1}^T \\ \mathbf{0}_{2M+1} & \frac{\partial^2 \log(L)}{\partial \boldsymbol{\lambda}_2 \partial \boldsymbol{\lambda}_2^T} \end{pmatrix} \right] = \begin{pmatrix} \frac{w_1(y_1 - \mu_1)}{\mu_1(1 - \mu_1)} & \mathbf{0}_{2M+1}^T \\ \mathbf{0}_{2M+1} & -E \left[\frac{\partial^2 \log(L)}{\partial \boldsymbol{\lambda}_2 \partial \boldsymbol{\lambda}_2^T} \right] \end{pmatrix},$$

where $\mathbf{0}_{2M+1}$ is a vector of $2M + 1$ zeroes. When the observation falls at the boundary of the sample space, i.e., $y_1 = 0$ or $y_1 = 1$, the maximum likelihood estimate of μ_1 is 0 or 1. Consequently, $\frac{w_1(y_1 - \mu_1)}{\mu_1(1 - \mu_1)}$ is undefined and causes a problem in calculating Wald test statistic with any software.

Under the reparameterized model the finite maximum likelihood estimate of δ_1 , however, does not exist in the strict sense when y_1 is zero or one since $-\infty < \delta_1 < +\infty$. Nevertheless, a program such as PROC NLMIXED/SAS will attempt to maximize the reparameterized likelihood function and identify an ‘‘approximate’’ maximum likelihood estimate of δ_1 in a sense that the likelihood function is approximately maximized, i.e.,

$$\frac{1}{1 + \exp\left(-(\widehat{\beta}_1 + \sum_{m=1}^M \widehat{\beta}_{m+1} x_{m1} + \widehat{\delta}_1)\right)} \approx 0$$

if $y_1 = 0$, or

$$\frac{1}{1 + \exp\left(-(\widehat{\beta}_1 + \sum_{m=1}^M \widehat{\beta}_{m+1} x_{m1} + \widehat{\delta}_1)\right)} \approx 1$$

if $y_1 = 1$. Please note that the maximum likelihood estimates identified in such a way can still approximately solve the log-likelihood equation

$$S(\boldsymbol{\gamma}) = \frac{\partial \log(L)}{\partial \boldsymbol{\gamma}} = 0.$$

Consequently, $U'_1(\boldsymbol{\gamma})E'_1(\boldsymbol{\gamma}) = w_1 \widehat{\mu}_1(1 - \widehat{\mu}_1) \approx 0$. Based on the information matrix

derived in Appendix A.3.2, we have

$$I(\hat{\gamma}) = KI_K(\hat{\gamma}) = \begin{pmatrix} K\hat{I}_{11} = -E \left\{ \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \delta_1} \right\} \Big|_{\gamma=\hat{\gamma}} & K\hat{I}_{12} = -E \left\{ \frac{\partial^2 \log(L)}{\partial \delta_1 \partial \gamma_2^T} \right\} \Big|_{\gamma=\hat{\gamma}} \\ K\hat{I}_{21} = -E \left\{ \frac{\partial^2 \log(L)}{\partial \gamma_2 \partial \delta_1} \right\} \Big|_{\gamma=\hat{\gamma}} & K\hat{I}_{22} = -E \left\{ \frac{\partial^2 \log(L)}{\partial \gamma_2 \partial \gamma_2^T} \right\} \Big|_{\gamma=\hat{\gamma}} \end{pmatrix} \\ \approx \begin{pmatrix} 0 & \mathbf{0}_{2M+1}^T \\ \mathbf{0}_{2M+1} & -E \left\{ \left[\frac{\partial^2 \log(L)}{\partial \gamma_2 \partial \gamma_2^T} \right] \right\} \Big|_{\gamma=\hat{\gamma}} \end{pmatrix}.$$

Therefore,

$$\left\{ \left[(KI(\hat{\gamma}))^{-1} \right]_{11} \right\}^{-1} = K(\hat{I}_{11} - \hat{I}_{12} \hat{I}_{22}^{-1} \hat{I}_{21}) \approx 0.$$

In real applications this quantity is generally less than 10^{-5} for tramadol and acetaminophen data set and less than 10^{-13} for marathion, parathion and piperonyl butoxide Data Set.

A two-sided Wald test statistic for δ based on reparameterized model has a general form as

$$T^W = \hat{\delta}_1^2 K(\hat{I}_{11} - \hat{I}_{12} \hat{I}_{22}^{-1} \hat{I}_{21}).$$

The estimates of δ_1 is generally between ± 20 for tramadol and acetaminophen data set and between ± 200 with a reasonable range of starting values; therefore, $T^W \approx 0$ when observations fall on sample space boundary. Please note that the estimates of δ_1 could be affected dramatically by specifying an extreme starting value in PROC NLMIXED/SAS. For example, when the starting value of δ_1 is set as 10^8 for $y_1 = 1$ in tramadol and acetaminophen data set, the final estimate of δ_1 will stick to the starting value, i.e., 10^8 . Consequently, T^W could be a big value even though

$$\left\{ \left[(KI(\hat{\gamma}))^{-1} \right]_{11} \right\}^{-1} = K(\hat{I}_{11} - \hat{I}_{12} \hat{I}_{22}^{-1} \hat{I}_{21}) \approx 0.$$

This is because $\hat{\delta}_1^2$ is about 10^{16} and is much larger than $\left[(KI(\hat{\gamma}))^{-1} \right]_{11}$. In summary,

the Wald test statistics do not give any valuable testing results for observations on sample space boundary.

The score test statistics can easily deal with observations on sample space boundary since it only requires maximum likelihood estimators calculated under the null hypothesis. The likelihood ratio test statistics can also be a good alternative for testing such observations if we don't reparameterize the model. Even under the reparameterization the likelihood ratio test statistics perform very well. The main reason is that the likelihood ratio test statistics only consider the maximum values of likelihood function. In other words, even though no finite maximum likelihood estimates of δ_1 exist for $y_1 = 0$ or $y_1 = 1$ under the reparameterization, the "approximate" maximum likelihood estimates of δ_1 still satisfy

$$\frac{1}{1 + \exp\left(-(\widehat{\beta}_1 + \sum_{m=1}^M \widehat{\beta}_{m+1} x_{m1} + \widehat{\delta}_1)\right)} \approx 0$$

if $y_1 = 0$, or

$$\frac{1}{1 + \exp\left(-(\widehat{\beta}_1 + \sum_{m=1}^M \widehat{\beta}_{m+1} x_{m1} + \widehat{\delta}_1)\right)} \approx 1$$

if $y_1 = 1$. This implies that the maximum values of likelihood function, i.e., the denominators in likelihood ratio test statistics, calculated under the reparameterization is very close to that calculated from the original likelihood function. It turns out that the likelihood ratio test statistics in our real applications are almost invariant to the reparameterization for observations on sample space boundary.

The same argument can be readily applied to the one-sided tests. Given a pair of one-sided tests:

$$\begin{aligned} (1) & H_{0,1} : \delta_1 = 0 \quad \text{vs.} \quad H_{1,1} : \delta_1 > 0 \\ (2) & H_{0,1'} : \delta_1 = 0 \quad \text{vs.} \quad H_{1,1'} : \delta_1 < 0, \end{aligned}$$

it can be easily seen that three one-sided test statistics for one test of (1) and (2) will have the same numerical values as their two-sided counterparts. Consequently, the Wald test statistic will face the same problem for observations on sample space boundary, whereas score and likelihood ratio test statistics will handle such observations well.

Appendix B

SAS Programs

B.1 $H_0^i : s_i = 0$ vs. $H_1^i : s_i > 0$ in a Binary Study

B.1.1 Wald Test Statistic

```
proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 s1=0.05 s2=0.5;
mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1 +
beta2*dose2*(1+dose1)**s2)));
model y~binomial(n,mu);
bounds s1>=0;
bounds beta1>0;
bounds beta2>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
proc iml;
use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];
s1=para[4,1];s2=para[5,1];
use complete;
read all into mix;
```

```

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;
                I33=I11;I34=I11;I35=I11;
                I44=I11;I45=I11;
                I55=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2];y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2));
p=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2)));
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp/((1+exp)**2);
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*((1+dose2)**s1);
s_beta2_m[i,1]=der_p*der_e*dose2*((1+dose1)**s2);
s_s1_m[i,1]=der_p*der_e*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I15[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*

```

```

dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
dose2*((1+dose1)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I44[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta1*dose1*((1+dose2)**s1)
*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

I55[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)
*log(1+dose1)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

end;

s=J(5,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);
s[4,1]=sum(s_s1_m);
s[5,1]=sum(s_s2_m);

I_11=sum(I11); I_12=sum(I12); I_13=sum(I13);
I_14=sum(I14); I_15=sum(I15);
I_21=I_12; I_22=sum(I22); I_23=sum(I23);

```

```

I_24=sum(I24);I_25=sum(I25);
I_31=I_13;    I_32=I_23;    I_33=sum(I33);
I_34=sum(I34);I_35=sum(I35);
I_41=I_14;    I_42=I_24;    I_43=I_34;
I_44=sum(I44);I_45=sum(I45);
I_51=I_15;    I_52=I_25;    I_53=I_35;
I_54=I_45;    I_55=sum(I55);

I_n=(I_11||I_12||I_13||I_14||I_15)//
      (I_21||I_22||I_23||I_24||I_25)//
      (I_31||I_32||I_33||I_34||I_35)//
      (I_41||I_42||I_43||I_44||I_45)//
      (I_51||I_52||I_53||I_54||I_55);

I_ninv=inv(I_n);

chi_W_s1=s1*s1/I_ninv[4,4];
if ((chi_w_s1-0)<1e-5) then Chi_P_s1=1;
else Chi_P_s1=(1-probchi(Chi_W_s1,1))/2;
print chi_W_s1 chi_p_s1;
quit;

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.001 beta2=0.005 s1=0.5 s2=0.05;
mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**S1
+ beta2*dose2*(1+dose1)**S2)));
model y~binomial(n,mu);
bounds s2>=0;
bounds beta1>0;
bounds beta2>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];

```

```

s1=para[4,1];s2=para[5,1];
use complete;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;
                I33=I11;I34=I11;I35=I11;
                I44=I11;I45=I11;
                I55=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2];y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2));
p=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2)));
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp/((1+exp)**2);
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*((1+dose2)**s1);
s_beta2_m[i,1]=der_p*der_e*dose2*((1+dose1)**s2);
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);

```

```

I15[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*dose2*((1+dose1)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I44[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta1*dose1*((1+dose2)**s1)
*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

I55[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)
*log(1+dose1)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

end;

s=J(5,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);

```

```

s[4,1]=sum(s_s1_m);
s[5,1]=sum(s_s2_m);

I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);
I_14=sum(I14);I_15=sum(I15);
I_21=I_12;    I_22=sum(I22);I_23=sum(I23);
I_24=sum(I24);I_25=sum(I25);
I_31=I_13;    I_32=I_23;    I_33=sum(I33);
I_34=sum(I34);I_35=sum(I35);
I_41=I_14;    I_42=I_24;    I_43=I_34;
I_44=sum(I44);I_45=sum(I45);
I_51=I_15;    I_52=I_25;    I_53=I_35;
I_54=I_45;    I_55=sum(I55);

```

```

I_n=(I_11||I_12||I_13||I_14||I_15)//
      (I_21||I_22||I_23||I_24||I_25)//
      (I_31||I_32||I_33||I_34||I_35)//
      (I_41||I_42||I_43||I_44||I_45)//
      (I_51||I_52||I_53||I_54||I_55);

```

```

I_ninv=inv(I_n);

```

```

chi_W_S2=s2*s2/I_ninv[5,5];
if (chi_W_s2-0)<1e-5 then Chi_P_s2=1;
else Chi_P_s2=(1-probchi(Chi_w_s2,1))/2;

```

```

print chi_W_s2 chi_p_s2;quit;

```

B.1.2 Score Test Statistic

```

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.001 beta2=0.005 s2=0.5;
mu=1/(1+exp(-(beta0+beta1*dose1 + beta2*dose2*(1+dose1)**S2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;

```

```

ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
proc iml;
use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];
s2=para[4,1];s1=0;
use complete;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;
                I33=I11;I34=I11;I35=I11;
                I44=I11;I45=I11;
                I55=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2];y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+beta2*dose2*(1+dose1)**s2));
p=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1+beta2*dose2*(1+dose1)**s2)));
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp/((1+exp)**2);
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*((1+dose2)**s1);
s_beta2_m[i,1]=der_p*der_e*dose2*((1+dose1)**s2);
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

```

```

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
I15[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
dose2*((1+dose1)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I44[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)*
log(1+dose2)*beta1*dose1*((1+dose2)**s1)
*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)*
log(1+dose2)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

I55[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)*
log(1+dose1)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

end;

```

```

s=J(5,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);
s[4,1]=sum(s_s1_m);
s[5,1]=sum(s_s2_m);

I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);I_15=sum(I15);
I_21=I_12;    I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);I_25=sum(I25);
I_31=I_13;    I_32=I_23;    I_33=sum(I33);I_34=sum(I34);I_35=sum(I35);
I_41=I_14;    I_42=I_24;    I_43=I_34;    I_44=sum(I44);I_45=sum(I45);
I_51=I_15;    I_52=I_25;    I_53=I_35;    I_54=I_45;    I_55=sum(I55);

I_n=(I_11||I_12||I_13||I_14||I_15)//
      (I_21||I_22||I_23||I_24||I_25)//
      (I_31||I_32||I_33||I_34||I_35)//
      (I_41||I_42||I_43||I_44||I_45)//
      (I_51||I_52||I_53||I_54||I_55);

I_ninv=inv(I_n);

chi=s'*I_ninv*s;

I22=I_n[4,4];I11=(I_n[1:3,1:3]||I_n[1:3,5])/(I_n[5,1:3]||I_n[5,5]);
I21=I_n[1:3,4]/I_n[5,4];I12=I21';
s_s1=s[4,1];
l=(I22-I12*inv(I11)*I21)/n_mix;
U=(s_s1)/(l*sqrt(n_mix));

if U<0 then chi_p=1;
else if ((chi-0)<1e-5) then chi_p=1;
else Chi_P=0.5*(1-probchi(Chi,1));
print chi chi_p U;quit;

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.01 beta2=0.005 s1=0.1;

```

```

mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**S1 + beta2*dose2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];
s1=para[4,1];s2=0;
use complete;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;
                    I33=I11;I34=I11;I35=I11;
                        I44=I11;I45=I11;
                            I55=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2];y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2));
p=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1+
beta2*dose2*(1+dose1)**s2)));
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp/((1+exp)**2);
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

```

```

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*((1+dose2)**s1);
s_beta2_m[i,1]=der_p*der_e*dose2*((1+dose1)**s2);
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
I15[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*dose2*((1+dose1)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);

I44[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta1*dose1*((1+dose2)**s1)
*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

```

```

I55[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)
*log(1+dose1)*beta2*dose2*((1+dose1)**s2)
*log(1+dose1);

end;

s=J(5,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);
s[4,1]=sum(s_s1_m);
s[5,1]=sum(s_s2_m);

I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);I_15=sum(I15);
I_21=I_12;    I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);I_25=sum(I25);
I_31=I_13;    I_32=I_23;    I_33=sum(I33);I_34=sum(I34);I_35=sum(I35);
I_41=I_14;    I_42=I_24;    I_43=I_34;    I_44=sum(I44);I_45=sum(I45);
I_51=I_15;    I_52=I_25;    I_53=I_35;    I_54=I_45;    I_55=sum(I55);

I_n=(I_11||I_12||I_13||I_14||I_15)//
      (I_21||I_22||I_23||I_24||I_25)//
      (I_31||I_32||I_33||I_34||I_35)//
      (I_41||I_42||I_43||I_44||I_45)//
      (I_51||I_52||I_53||I_54||I_55);

I_ninv=inv(I_n);

chi=s'*I_ninv*s;

I22=I_n[5,5];I11=(I_n[1:4,1:4]);
I21=I_n[1:4,5];I12=I21';
s_s2=s[5,1];
l=(I22-I12*inv(I11)*I21)/n_mix;
U=(s_s2)/(l*sqrt(n_mix));

```

```

if U<0 then chi_p=1;
else if ((chi-0)<1e-5) then chi_p=1;
else Chi_P=0.5*(1-probchi(Chi,1));
print chi chi_p U;quit;

```

B.1.3 Likelihood Ratio Test

```

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 s1=0.05 s2=0.5;
mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**s1
+ beta2*dose2*(1+dose1)**s2)));
model y~binomial(n,mu);
bounds s1>=0;
bounds beta1>0;
bounds beta2>0;
ods output FitStatistics=L_full;
run;

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 s2=0.5;
mu=1/(1+exp(-(beta0+beta1*dose1 +
beta2*dose2*(1+dose1)**S2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
ods output FitStatistics=L_reduced;
run;

proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];
if ((chi_L-0)<1e-5) then chi_s1_P=1;
else chi_s1_P=0.5*(1-probchi(chi_L,1));
print Chi_L chi_s1_p;quit;

proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.001 beta2=0.005 s1=0.5 s2=0.05;

```

```

mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**S1
+ beta2*dose2*(1+dose1)**S2)));
model y~binomial(n,mu);
bounds s2>=0;
bounds beta1>0;
bounds beta2>0;
ods output FitStatistics=L_full;
run;
proc nlmixed data=complete corr cov;
parms beta0=-3 beta1=0.001 beta2=0.005 s1=0.5;
mu=1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**S1 + beta2*dose2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
ods output FitStatistics=L_reduced;
run;
proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];
if chi_L<0 then chi_L=0;
if ((chi_L-0)<1e-5) then chi_s2_p=1;
else chi_s2_P=0.5*(1-probchi(chi_L,1));
print Chi_L chi_s2_p;quit;

```

B.2 $H_{0,i} : \delta_i = 0$ vs. $H_{1,i} : \delta_i > 0$ in Tramadol and Acetaminophen Data Set

Wald Test

```

Data complete;
set complete;
I_3=0;
run;

```

```

%Macro L00CV;
%Do i=20 %TO 91;
Data Complete_L00;
set complete;
if ID=&i then do; I_3=1;end;
run;

Data pred_One;
set complete;
if ID=&i;
run;

proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 s1=-0.05 s2=0.5 delta=0.5;
mu=(1-I_3)*1/(1+exp(-(beta0+beta1*dose1*
(1+dose2)**S1 + beta2*dose2*(1+dose1)**S2)))
+I_3*1/(1+exp(-(beta0+beta1*dose1+ beta2*dose2+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
ods output ParameterEstimates=para_Est;
title "&i";

run;

Data add;
set complete_L00;
if I_3=1;
run;
Data mix;
set complete_L00;
if I_3=0;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];
s1=para[4,1]; s2=para[5,1]; delta=para[6,1];

```

```

use add;
read all into add ;
use mix;
read all into mix;

dose1=add[1,1];dose2=add[1,2]; y=add[1,3];m=add[1,4];
exp_a=exp(-(beta0+beta1*dose1+beta2*dose2+delta));
if exp_a<1E-15 then exp_a=1E-15;
p=1/(1+exp_a);
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp_a/(1+exp_a)**2;
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_a=der_p*der_e*1;
s_beta1_a=der_p*der_e*dose1;
s_beta2_a=der_p*der_e*dose2;
s_s1_a=0;
s_s2_a=0;
s_delta_a=der_p*der_e*1;

I11_a=der_2e*der_2p;
I12_a=der_2e*der_2p*dose1;
I13_a=der_2e*der_2p*dose2;
I14_a=0;I15_a=0;I16_a=der_2e*der_2p*1;

I22_a=der_2e*dose1*der_2p*dose1;
I23_a=der_2e*dose1*der_2p*dose2;
I24_a=0;I25_a=0;I26_a=der_2e*dose1*der_2p*1;

I33_a=der_2e*dose2*der_2p*dose2;
I34_a=0;I35_a=0;I36_a=der_2e*dose2*der_2p*1;

I44_a=0;I45_a=0;I46_a=0;

I55_a=0;I56_a=0;

I66_a=der_2e*der_2p*1;

```

```

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;
                I33=I11;I34=I11;I35=I11;I36=I11;
                I44=I11;I45=I11;I46=I11;
                I55=I11;I56=I11;
                I66=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+
          beta2*dose2*(1+dose1)**s2));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1)**s2;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2))*((1+dose2)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1))*((1+dose1)**s2);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*

```

```

((1+dose2)**s1)*log(1+dose2);
I15[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);
I16[i,1]=0;

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I26[i,1]=0;

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*dose2*((1+dose1)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I36[i,1]=0;

I44[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I46[i,1]=0;

I55[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I56[i,1]=0;

I66[i,1]=0;
end;

s=J(6,1,0);

```

```

s[1,1]=sum(s_beta0_m)+s_beta0_a;
s[2,1]=sum(s_beta1_m)+s_beta1_a;
s[3,1]=sum(s_beta2_m)+s_beta2_a;
s[4,1]=sum(s_s1_m)+s_s1_a;
s[5,1]=sum(s_s2_m)+s_s2_a;
s[6,1]=s_delta_a;

I_11=sum(I11)+I11_a;I_12=sum(I12)+I12_a;I_13=sum(I13)+I13_a;
I_14=sum(I14)+I14_a;I_15=sum(I15)+I15_a;I_16=sum(I16)+I16_a;
I_21=I_12;          I_22=sum(I22)+I22_a;I_23=sum(I23)+I23_a;
I_24=sum(I24)+I24_a;I_25=sum(I25)+I25_a;I_26=sum(I26)+I26_a;
I_31=I_13;          I_32=I_23;          I_33=sum(I33)+I33_a;
I_34=sum(I34)+I34_a;I_35=sum(I35)+I35_a;I_36=sum(I36)+I36_a;
I_41=I_14;          I_42=I_24;          I_43=I_34;
I_44=sum(I44)+I44_a;I_45=sum(I45)+I45_a;I_46=sum(I46)+I46_a;
I_51=I_15;          I_52=I_25;          I_53=I_35;
I_54=I_45;          I_55=sum(I55)+I55_a;I_56=sum(I56)+I56_a;
I_61=I_16;          I_62=I_26;          I_63=I_36;
I_64=I_46;          I_65=I_56;          I_66=sum(I66)+I66_a;

I_n=(I_11||I_12||I_13||I_14||I_15||I_16)//
      (I_21||I_22||I_23||I_24||I_25||I_26)//
      (I_31||I_32||I_33||I_34||I_35||I_36)//
      (I_41||I_42||I_43||I_44||I_45||I_46)//
      (I_51||I_52||I_53||I_54||I_55||I_56)//
      (I_61||I_62||I_63||I_64||I_65||I_66);

/*if s[8,1]=. then do;
x=&i; print x exp_a;
end;*/

I_ninv=inv(I_n);
chi_W=delta**2/I_ninv[6,6];
if (chi_w-0)<1e-5 then chi_p=1;
else Chi_P=0.5*(1-probchi(Chi_W,1));

use pred_one;
read all into one;

p_value=one[,1:4]||chi_W||chi_p||one[,5]||s_delta_a;
create p_value from p_value
[colname={'dose1','dose2','y','n','chi_w','pvalue_w','ID','s_delta'}];

```

```

append from p_value;

%IF &i=20 %Then %DO;
Data pvalue_wald;
set P_value;run;
%END;
%ELSE %DO;
Data pvalue_wald;
set pvalue_wald p_value;
run;%END;

%END;
%mend LOOCV;

%loocv;

proc print data=pvalue_wald;run;

```

Score Test

```

Data complete;
set complete;
I_3=0;
run;

%Macro LOOCV;
%Do i=20 %TO 91;
Data Complete_L00;
set complete;
if ID=&i then do; I_3=1;end;
run;

Data pred_One;
set complete;
if ID=&i;
run;

proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 s1=-0.05 s2=0.5;
mu=(1-I_3)*1/(1+exp(-(beta0+beta1*dose1*

```

```

(1+dose2)**S1 + beta2*dose2*(1+dose1)**S2)))
+I_3*1/(1+exp(-(beta0+beta1*dose1+ beta2*dose2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
ods output ParameterEstimates=para_Est;
title "&i";
run;

Data add;
set complete_L00;
if I_3=1;
run;
Data mix;
set complete_L00;
if I_3=0;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1];
s1=para[4,1]; s2=para[5,1]; delta=0;
use add;
read all into add ;
use mix;
read all into mix;

dose1=add[1,1];dose2=add[1,2]; y=add[1,3];m=add[1,4];
exp_a=exp(-(beta0+beta1*dose1+beta2*dose2+delta));
if exp_a<1E-15 then exp_a=1E-15;
p=1/(1+exp_a);
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp_a/(1+exp_a)**2;
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_a=der_p*der_e*1;

```

```

s_beta1_a=der_p*der_e*dose1;
s_beta2_a=der_p*der_e*dose2;
s_s1_a=0;
s_s2_a=0;
s_delta_a=der_p*der_e*1;

I11_a=der_2e*der_2p;
I12_a=der_2e*der_2p*dose1;
I13_a=der_2e*der_2p*dose2;
I14_a=0;I15_a=0;I16_a=der_2e*der_2p*1;

I22_a=der_2e*dose1*der_2p*dose1;
I23_a=der_2e*dose1*der_2p*dose2;
I24_a=0;I25_a=0;I26_a=der_2e*dose1*der_2p*1;

I33_a=der_2e*dose2*der_2p*dose2;
I34_a=0;I35_a=0;I36_a=der_2e*dose2*der_2p*1;

I44_a=0;I45_a=0;I46_a=0;

I55_a=0;I56_a=0;

I66_a=der_2e*der_2p*1;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;
      I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;
            I33=I11;I34=I11;I35=I11;I36=I11;
                  I44=I11;I45=I11;I46=I11;
                          I55=I11;I56=I11;
                                  I66=I11;

```

```

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; y=mix[i,3];m=mix[i,4];
exp=exp(-(beta0+beta1*dose1*(1+dose2)**s1+
          beta2*dose2*(1+dose1)**s2));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1)**s2;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2))*((1+dose2)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1))*((1+dose1)**s2);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2);
I14[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2)**s1)*log(1+dose2);
I15[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1)**s2)*log(1+dose1);
I16[i,1]=0;

I22[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose1*((1+dose2)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*dose2*((1+dose1)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2)**s1)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I26[i,1]=0;

I33[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*dose2*((1+dose1)**s2);

```

```

I34[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1)**s2)
*beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I36[i,1]=0;

I44[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*
beta1*dose1*((1+dose2)**s1)*log(1+dose2);
I45[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2)**s1)
*log(1+dose2)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I46[i,1]=0;

I55[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1)**s2)
*log(1+dose1)*
beta2*dose2*((1+dose1)**s2)*log(1+dose1);
I56[i,1]=0;

I66[i,1]=0;
end;

s=J(6,1,0);
s[1,1]=sum(s_beta0_m)+s_beta0_a;
s[2,1]=sum(s_beta1_m)+s_beta1_a;
s[3,1]=sum(s_beta2_m)+s_beta2_a;
s[4,1]=sum(s_s1_m)+s_s1_a;
s[5,1]=sum(s_s2_m)+s_s2_a;
s[6,1]=s_delta_a;

I_11=sum(I11)+I11_a; I_12=sum(I12)+I12_a; I_13=sum(I13)+I13_a;
I_14=sum(I14)+I14_a; I_15=sum(I15)+I15_a; I_16=sum(I16)+I16_a;
I_21=I_12; I_22=sum(I22)+I22_a; I_23=sum(I23)+I23_a;
I_24=sum(I24)+I24_a; I_25=sum(I25)+I25_a; I_26=sum(I26)+I26_a;
I_31=I_13; I_32=I_23; I_33=sum(I33)+I33_a;
I_34=sum(I34)+I34_a; I_35=sum(I35)+I35_a; I_36=sum(I36)+I36_a;
I_41=I_14; I_42=I_24; I_43=I_34;
I_44=sum(I44)+I44_a; I_45=sum(I45)+I45_a; I_46=sum(I46)+I46_a;
I_51=I_15; I_52=I_25; I_53=I_35;
I_54=I_45; I_55=sum(I55)+I55_a; I_56=sum(I56)+I56_a;
I_61=I_16; I_62=I_26; I_63=I_36;

```

```

I_64=I_46;          I_65=I_56;          I_66=sum(I66)+I66_a;

I_n=(I_11||I_12||I_13||I_14||I_15||I_16)//
      (I_21||I_22||I_23||I_24||I_25||I_26)//
      (I_31||I_32||I_33||I_34||I_35||I_36)//
      (I_41||I_42||I_43||I_44||I_45||I_46)//
      (I_51||I_52||I_53||I_54||I_55||I_56)//
      (I_61||I_62||I_63||I_64||I_65||I_66);

/*if s[8,1]=. then do;
x=&i; print x exp_a;
end;*/

I_ninv=inv(I_n);
chi=I_ninv[6,6]*s_delta_a**2;
if s_delta_a<0 then Chi_p=1;
else if (Chi-0<1e-5) then Chi_P=1;
else Chi_P=(1-probchi(Chi,1))/2;

use pred_one;
read all into one;

p_value=one[,1:4]||chi||chi_p||one[,5]||s_delta_a;
create p_value from p_value
[colname={'dose1', 'dose2', 'y', 'n', 'chi_s',
'pvalue_s', 'ID', 's_delta'}];
append from p_value;

%IF &i=20 %Then %DO;
Data pvalue_score;
set P_value;run;
%END;
%ELSE %DO;
Data pvalue_score;
set pvalue_score p_value;
run;%END;

%END;
%mend LOOCV;

```

```
%loocv;
```

```
proc print data=pvalue_score;run;
```

Likelihood Ratio Test

```
Data complete;  
set complete;  
I_3=0;  
run;
```

```
%Macro LOOCV;  
%Do i=20 %TO 91;  
Data Complete_L00;  
set complete;  
if ID=&i then do; I_3=1;end;  
run;
```

```
Data pred_One;  
set complete;  
if ID=&i;  
run;
```

```
proc nlmixed data=complete_L00 corr cov;  
parms beta0=-3 beta1=0.05 beta2=0.005  
s1=-0.05 s2=0.5 delta=0.5;  
mu=(1-I_3)*1/(1+exp(-(beta0+beta1*dose1*(1+dose2)**S1  
+ beta2*dose2*(1+dose1)**S2)))  
+I_3*1/(1+exp(-(beta0+beta1*dose1+ beta2*dose2+delta)));  
model y~binomial(n,mu);  
bounds delta>=0;  
bounds beta1>0;  
bounds beta2>0;  
title "ID=&i";  
ods output FitStatistics=L_full;  
run;
```

```
proc nlmixed data=complete_L00 corr cov;  
parms beta0=-3 beta1=0.05 beta2=0.005 s1=-0.05 s2=0.5;  
mu=(1-I_3)*1/(1+exp(-(beta0+beta1*dose1*
```

```

(1+dose2)**S1 + beta2*dose2*(1+dose1)**S2)))
+I_3*1/(1+exp(-(beta0+beta1*dose1+ beta2*dose2)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
title "ID=&i";
ods output FitStatistics=L_reduced;
run;

```

```

Data add;
set complete_L00;
if I_3=1;
run;
Data mix;
set complete_L00;
if I_3=0;
run;

```

```

proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

```

```

chi_L=l_r[1,1]-l_full[1,1];
if (chi_L-0)<1e-5 then chi_L_P=1;
else chi_L_P=(1-probchi(chi_L,1))/2;

```

```

use pred_one;
read all into one;

```

```

use pred_one;
read all into one;

```

```

p_value=one[,1:4]||chi_L||chi_L_p||one[,5];
create p_value from p_value
[colname={'dose1','dose2','y','n',
'chi_l','pvalue_lik','ID'}];
append from p_value;

```

```

%IF &i=20 %Then %DO;
Data pvalue_Lik;
set P_value;run;
%END;
%ELSE %DO;
Data pvalue_Lik;
set pvalue_lik p_value;
run;%END;

%END;
%mend LOOCV;

%loocv;

proc print data=pvalue_lik;run;

```

B.3 $H_0 : s_i = 0$ vs. $H_1 : s_i > 0$ in a Tertiary Mixture Study

B.3.1 Wald Test Statistic

```

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=1 s1=0.5 s2=-0.05 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+M+PBO)**S1
+ beta2*P*(1+M+PBO)**S2
+beta3*PBO*(1+M+P)**S3
))) ;
bounds s1>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
model y~binomial(n,mu);
ods output ParameterEstimates=para_Est;
run;

proc iml;
use para_Est;

```

```

read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1];s2=para[6,1]; s3=para[7,1];
use total;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;
I15=I11;I16=I11;I17=I11;
I22=I11;I23=I11;I24=I11;
I25=I11;I26=I11;I27=I11;
I33=I11;I34=I11;
I35=I11;I36=I11;I37=I11;
I44=I11;I45=I11;I46=I11;I47=I11;
I55=I11;I56=I11;I57=I11;
I66=I11;I67=I11;
I77=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

```

```

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*
(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*
(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*
(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

```

```

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*
((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*
((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*

```

```
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
```

```
end;
```

```
s=J(7,1,0);  
s[1,1]=sum(s_beta0_m);  
s[2,1]=sum(s_beta1_m);  
s[3,1]=sum(s_beta2_m);  
s[4,1]=sum(s_beta3_m);  
s[5,1]=sum(s_s1_m);  
s[6,1]=sum(s_s2_m);  
s[7,1]=sum(s_s3_m);
```

```
I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);  
I_15=sum(I15);I_16=sum(I16);I_17=sum(I17);  
I_21=I_12; I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);  
I_25=sum(I25);I_26=sum(I26);I_27=sum(I27);  
I_31=I_13; I_32=I_23; I_33=sum(I33);I_34=sum(I34);  
I_35=sum(I35);I_36=sum(I36);I_37=sum(I37);  
I_41=I_14; I_42=I_24; I_43=I_34; I_44=sum(I44);  
I_45=sum(I45);I_46=sum(I46);I_47=sum(I47);  
I_51=I_15; I_52=I_25; I_53=I_35; I_54=I_45;  
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);  
I_61=I_16; I_62=I_26; I_63=I_36; I_64=I_46;  
I_65=I_56; I_66=sum(I66);I_67=sum(I67);  
I_71=I_17; I_72=I_27; I_73=I_37; I_74=I_47;  
I_75=I_57; I_76=I_67; I_77=sum(I77);
```

```
I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//  
(I_21||I_22||I_23||I_24||I_25||I_26||I_27)//  
(I_31||I_32||I_33||I_34||I_35||I_36||I_37)//  
(I_41||I_42||I_43||I_44||I_45||I_46||I_47)//  
(I_51||I_52||I_53||I_54||I_55||I_56||I_57)//  
(I_61||I_62||I_63||I_64||I_65||I_66||I_67)//  
(I_71||I_72||I_73||I_74||I_75||I_76||I_77);
```

```
I_ninv=inv(I_n);  
chi=s1**2/I_ninv[5,5];
```

```
I22=I_n[5,5];I11=(I_n[1:4,1:4]||I_n[1:4,6:7])//
```

```

(I_n[6:7,1:4]||I_n[6:7,6:7]);
I21=I_n[1:4,5]/I_n[6:7,5];I12=I21';
s_s1=s[5,1];
Chi_alt=(s1**2)*(I22-I12*inv(I11)*I21);
if chi-0<1e-5 then chi_p=1;
else Chi_P=(1-probchi(Chi,1))/2;

print chi chi_p;quit;

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=1 s1=-0.5 s2=0.2 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P*(1+M+PB0)**s2+beta3*PB0*(1+M+P)**s3))) ;
model y~binomial(n,mu);
bounds s2>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output ParameterEstimates=para_Est;
run;

proc iml;
use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1];s2=para[6,1]; s3=para[7,1];
use total;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;

```

```

I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;
I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;
I44=I11;I45=I11;I46=I11;I47=I11;
I55=I11;I56=I11;I57=I11;
I66=I11;I67=I11;
I77=I11;

```

```

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

```

```

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

```

```

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

```

```

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*
((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*
((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*

```

```

((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I55[i,1]=der_2e*der_2p*beta1*dose1*

```

```

((1+dose2+dose3)**s1)*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

end;

s=J(7,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);
s[4,1]=sum(s_beta3_m);
s[5,1]=sum(s_s1_m);
s[6,1]=sum(s_s2_m);
s[7,1]=sum(s_s3_m);

I_11=sum(I11); I_12=sum(I12); I_13=sum(I13); I_14=sum(I14);
I_15=sum(I15); I_16=sum(I16); I_17=sum(I17);
I_21=I_12; I_22=sum(I22); I_23=sum(I23); I_24=sum(I24);
I_25=sum(I25); I_26=sum(I26); I_27=sum(I27);
I_31=I_13; I_32=I_23; I_33=sum(I33); I_34=sum(I34);
I_35=sum(I35); I_36=sum(I36); I_37=sum(I37);
I_41=I_14; I_42=I_24; I_43=I_34; I_44=sum(I44);
I_45=sum(I45); I_46=sum(I46); I_47=sum(I47);

```

```

I_51=I_15;    I_52=I_25;    I_53=I_35;    I_54=I_45;
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);
I_61=I_16;    I_62=I_26;    I_63=I_36;    I_64=I_46;
I_65=I_56;    I_66=sum(I66);I_67=sum(I67);
I_71=I_17;    I_72=I_27;    I_73=I_37;    I_74=I_47;
I_75=I_57;    I_76=I_67;    I_77=sum(I77);

```

```

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77);

```

```

I_ninv=inv(I_n);
chi=s2**2/I_ninv[6,6];

```

```

I22=I_n[6,6];I11=(I_n[1:5,1:5]||I_n[1:5,7])//
(I_n[7,1:5]||I_n[7,7]);
I21=I_n[1:5,6]//I_n[7,6];I12=I21';
s_s2=s[6,1];

```

```

Chi_alt2=(s2**2)*(I22-I12*inv(I11)*I21);
if chi-0<1e-5 then chi_p=1;
else Chi_P=(1-probchi(Chi,1))/2;

```

```

print chi chi_p;quit;

```

```

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=0.5 s1=-0.5 s2=0.2 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
  beta2*P*(1+M+PB0)**s2+beta3*PB0*(1+M+P)**s3))) ;
model y~binomial(n,mu);
bounds s3>=0;
bounds beta1>0;

```

```

bounds beta2>0;
bounds beta3>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;

proc iml;
use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1]; s2=para[6,1]; s3=para[7,1];
use total;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11; I13=I11; I14=I11; I15=I11; I16=I11; I17=I11;
                I22=I11; I23=I11; I24=I11; I25=I11; I26=I11; I27=I11;
                I33=I11; I34=I11; I35=I11; I36=I11; I37=I11;
                I44=I11; I45=I11; I46=I11; I47=I11;
                I55=I11; I56=I11; I57=I11;
                I66=I11; I67=I11;
                I77=I11;

do i=1 to n_mix;
dose1=mix[i,1]; dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4]; m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

```

```

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```
end;
```

```
s=J(7,1,0);  
s[1,1]=sum(s_beta0_m);  
s[2,1]=sum(s_beta1_m);  
s[3,1]=sum(s_beta2_m);  
s[4,1]=sum(s_beta3_m);  
s[5,1]=sum(s_s1_m);  
s[6,1]=sum(s_s2_m);  
s[7,1]=sum(s_s3_m);
```

```
I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);  
I_15=sum(I15);I_16=sum(I16);I_17=sum(I17);  
I_21=I_12; I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);  
I_25=sum(I25);I_26=sum(I26);I_27=sum(I27);  
I_31=I_13; I_32=I_23; I_33=sum(I33);I_34=sum(I34);  
I_35=sum(I35);I_36=sum(I36);I_37=sum(I37);  
I_41=I_14; I_42=I_24; I_43=I_34; I_44=sum(I44);  
I_45=sum(I45);I_46=sum(I46);I_47=sum(I47);  
I_51=I_15; I_52=I_25; I_53=I_35; I_54=I_45;  
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);  
I_61=I_16; I_62=I_26; I_63=I_36; I_64=I_46;  
I_65=I_56; I_66=sum(I66);I_67=sum(I67);  
I_71=I_17; I_72=I_27; I_73=I_37; I_74=I_47;  
I_75=I_57; I_76=I_67; I_77=sum(I77);
```

```
I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//  
(I_21||I_22||I_23||I_24||I_25||I_26||I_27)//  
(I_31||I_32||I_33||I_34||I_35||I_36||I_37)//  
(I_41||I_42||I_43||I_44||I_45||I_46||I_47)//  
(I_51||I_52||I_53||I_54||I_55||I_56||I_57)//  
(I_61||I_62||I_63||I_64||I_65||I_66||I_67)//  
(I_71||I_72||I_73||I_74||I_75||I_76||I_77);
```

```
I_ninv=inv(I_n);  
chi=s3**2/I_ninv[7,7];  
if (chi-0<1e-5) then chi_p=1;  
else Chi_P=(1-probchi(Chi,1))/2;
```

```

I22=I_n[7,7];I11=I_n[1:6,1:6];
I21=I_n[1:6,7];I12=I21';
s_s3=s[7,1];
Chi_alt3=(s3**2)*(I22-I12*inv(I11)*I21);

print chi chi_p;quit;

```

B.3.2 Score Test Statistic

```

proc nlmixed data=total corr cov;
parms beta1=-3 beta2=0.05 beta3=0.005
beta4=1 s2=-0.05 s3=0.2;
mu=1/(1+exp(-(beta1+beta2*M + beta3*P*(1+M+PBO)**S2
+beta4*PBO*(1+M+P)**S3
))) ;
model y~binomial(n,mu);
bounds beta2>0;
bounds beta3>0;
bounds beta4>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
proc iml;
use para_Est;
read all into para;
beta1=para[1,1];
beta2=para[2,1]; beta3=para[3,1]; beta4=para[4,1];
s1=0;s2=para[5,1]; s3=para[6,1];
use total;
read all into mix;

n_mix=nrow(mix);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_beta4_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);

```

```

s_s3_m=J(n_mix,1,0);

I11=s_beta1_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;
I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;
I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;
I44=I11;I45=I11;I46=I11;I47=I11;
I55=I11;I56=I11;I57=I11;
I66=I11;I67=I11;
I77=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4];m=mix[i,5];
exp=exp(-(beta1+beta2*dose1*(1+dose2+dose3)**s1+
          beta3*dose2*(1+dose1+dose3)**s2+
          beta4*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta1_m[i,1]=der_p*der_e;
s_beta2_m[i,1]=der_p*der_e*dose1*
(1+dose2+dose3)**s1;
s_beta3_m[i,1]=der_p*der_e*dose2*
(1+dose1+dose3)**s2;
s_beta4_m[i,1]=der_p*der_e*dose3*
(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta2*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta3*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta4*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*
((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*

```

```

((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta2*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta3*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta4*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta3*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```

I55[i,1]=der_2e*der_2p*beta2*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta2*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta2*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta3*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta3*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta4*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta4*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

end;

s=J(7,1,0);
s[1,1]=sum(s_beta1_m);
s[2,1]=sum(s_beta2_m);
s[3,1]=sum(s_beta3_m);
s[4,1]=sum(s_beta4_m);
s[5,1]=sum(s_s1_m);
s[6,1]=sum(s_s2_m);
s[7,1]=sum(s_s3_m);

I_11=sum(I11); I_12=sum(I12); I_13=sum(I13); I_14=sum(I14);
I_15=sum(I15); I_16=sum(I16); I_17=sum(I17);
I_21=I_12; I_22=sum(I22); I_23=sum(I23); I_24=sum(I24);
I_25=sum(I25); I_26=sum(I26); I_27=sum(I27);
I_31=I_13; I_32=I_23; I_33=sum(I33); I_34=sum(I34);
I_35=sum(I35); I_36=sum(I36); I_37=sum(I37);

```

```

I_41=I_14;    I_42=I_24;    I_43=I_34;    I_44=sum(I44);
I_45=sum(I45);I_46=sum(I46);I_47=sum(I47);
I_51=I_15;    I_52=I_25;    I_53=I_35;    I_54=I_45;
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);
I_61=I_16;    I_62=I_26;    I_63=I_36;    I_64=I_46;
I_65=I_56;    I_66=sum(I66);I_67=sum(I67);
I_71=I_17;    I_72=I_27;    I_73=I_37;    I_74=I_47;
I_75=I_57;    I_76=I_67;    I_77=sum(I77);

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77);
I_n=I_n/n_mix;
I11=I_n[5,5];
I22=(I_n[1:4,1:4]||I_n[1:4,6:7])/(I_n[6:7,1:4]||I_n[6:7,6:7]);
I21=I_n[1:4,5]/I_n[6:7,5];I12=I21';
s_s1=s[5,1];
A_inv=(I11-I12*inv(I22)*I21);
U=(s_s1)/((I11-I12*inv(I22)*I21)*(n_mix)**(0.5));
chi=U'*A_inv*U;
if s_s1>0 then Chi_Pscore=(1-probchi(Chi,1))/2;
else Chi_Pscore=1;
print Chi_Pscore chi;

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1 s1=-0.5 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P+beta3*PB0*(1+M+P)**S3))) ;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output ParameterEstimates=para_Est;
run;

proc iml;
use para_Est;

```

```

read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1];s2=0; s3=para[6,1];
use total;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;
                I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;
                I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;
                I44=I11;I45=I11;I46=I11;I47=I11;
                I55=I11;I56=I11;I57=I11;
                I66=I11;I67=I11;
                I77=I11;

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1+dose3)**s2;

```

```

s_beta3_m[i,1]=der_p*der_e*dose3*(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*(log(1+dose2+dose3))
*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*(log(1+dose1+dose3))
*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*(log(1+dose1+dose2))
*((1+dose1+dose2)**s3);

```

```

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)
**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)
**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)
**s3)*log(1+dose1+dose2);

```

```

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

```

```

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);

```

```

I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)*
log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

end;

s=J(7,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);

```

```

s[4,1]=sum(s_beta3_m);
s[5,1]=sum(s_s1_m);
s[6,1]=sum(s_s2_m);
s[7,1]=sum(s_s3_m);

I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);
I_15=sum(I15);I_16=sum(I16);I_17=sum(I17);
I_21=I_12;    I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);
I_25=sum(I25);I_26=sum(I26);I_27=sum(I27);
I_31=I_13;    I_32=I_23;    I_33=sum(I33);I_34=sum(I34);
I_35=sum(I35);I_36=sum(I36);I_37=sum(I37);
I_41=I_14;    I_42=I_24;    I_43=I_34;    I_44=sum(I44);
I_45=sum(I45);I_46=sum(I46);I_47=sum(I47);
I_51=I_15;    I_52=I_25;    I_53=I_35;    I_54=I_45;
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);
I_61=I_16;    I_62=I_26;    I_63=I_36;    I_64=I_46;
I_65=I_56;    I_66=sum(I66);I_67=sum(I67);
I_71=I_17;    I_72=I_27;    I_73=I_37;    I_74=I_47;
I_75=I_57;    I_76=I_67;    I_77=sum(I77);

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77);

I_ninv=inv(I_n);

chi=s'*I_ninv*s;
Chi_P=1-probchi(Chi,1);

I22=I_n[6,6];I11=(I_n[1:5,1:5]||I_n[1:5,7])
//(I_n[7,1:5]||I_n[7,7]);
I21=I_n[1:5,6]/I_n[7,6];I12=I21';
s_s2=s[6,1];

```

```

l=(I22-I12*inv(I11)*I21)/n_mix;
U=(s_s2)/(1*sqrt(n_mix));
Chi_2=(s_s2**2)*inv(I22-I12*inv(I11)*I21);
chi_U=U'*1*U;

if s_s2>0 then Chi_Pscore=(1-probchi(Chi,1))/2;
else Chi_Pscore=1;
print chi_Pscore chi;quit;

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1 s1=-0.5 s2=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P*(1+M+PB0)**s2+beta3*PB0))) ;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;

proc iml;
use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1];s2=para[6,1];s3=0;
use total;
read all into mix;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;
I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;
I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;

```

```

I44=I11;I45=I11;I46=I11;I47=I11;
I55=I11;I56=I11;I57=I11;
I66=I11;I67=I11;
I77=I11;

```

```

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3]; y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)
**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)
**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)
**s3)*log(1+dose1+dose2);

```

```

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)*
dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)*
dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)

```

```

*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);

end;

s=J(7,1,0);
s[1,1]=sum(s_beta0_m);
s[2,1]=sum(s_beta1_m);
s[3,1]=sum(s_beta2_m);
s[4,1]=sum(s_beta3_m);
s[5,1]=sum(s_s1_m);
s[6,1]=sum(s_s2_m);
s[7,1]=sum(s_s3_m);

I_11=sum(I11);I_12=sum(I12);I_13=sum(I13);I_14=sum(I14);
I_15=sum(I15);I_16=sum(I16);I_17=sum(I17);
I_21=I_12; I_22=sum(I22);I_23=sum(I23);I_24=sum(I24);
I_25=sum(I25);I_26=sum(I26);I_27=sum(I27);
I_31=I_13; I_32=I_23; I_33=sum(I33);I_34=sum(I34);
I_35=sum(I35);I_36=sum(I36);I_37=sum(I37);
I_41=I_14; I_42=I_24; I_43=I_34; I_44=sum(I44);
I_45=sum(I45);I_46=sum(I46);I_47=sum(I47);
I_51=I_15; I_52=I_25; I_53=I_35; I_54=I_45;
I_55=sum(I55);I_56=sum(I56);I_57=sum(I57);
I_61=I_16; I_62=I_26; I_63=I_36; I_64=I_46;
I_65=I_56; I_66=sum(I66);I_67=sum(I67);
I_71=I_17; I_72=I_27; I_73=I_37; I_74=I_47;
I_75=I_57; I_76=I_67; I_77=sum(I77);

```

```

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77);

I_ninv=inv(I_n);

chi=s'*I_ninv*s;
Chi_P=1-probchi(Chi,1);

I22=I_n[7,7];I11=I_n[1:6,1:6];
I21=I_n[1:6,7];I12=I21';
s_s3=s[7,1];
l=(I22-I12*inv(I11)*I21)/n_mix;
U=(s_s3)/(1*sqrt(n_mix));
Chi_3=(s_s3**2)*inv(I22-I12*inv(I11)*I21);
chi_U=U'*1*U;

if s_s3>0 then Chi_Pscore=(1-probchi(Chi,1))/2;
else Chi_Pscore=1;

print Chi_Pscore chi;quit;

```

B.3.3 Likelihood Ratio Test Statistic

```

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=0.5 s2=-0.05 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+M+PB0)**S1 +
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3
))) ;

```

```

bounds s1>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
model y~binomial(n,mu);
ods output FitStatistics=L_full;
run;
proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=1 s2=-0.05 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M + beta2*P*
(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3
))) ;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output FitStatistics=L_reduced;
run;

proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];
if (chi_L-0)<1e-5 then chi_s1_P=1;
else chi_s1_P=(1-probchi(chi_L,1))/2;
print Chi_L chi_s1_p;quit;

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=-0.5 s2=0.2 s3=4;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1+
beta2*P*(1+M+PB0)**s2+beta3*PB0*(1+M+P)**s3))) ;
model y~binomial(n,mu);
bounds s2>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;

```

```

ods output FitStatistics=L_full;
run;
proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=0.5 s1=-0.5 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P+beta3*PB0*(1+M+P)**S3))) ;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output FitStatistics=L_reduced;
run;
proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];
if (chi_L-0)<1e-5 then chi_s2_P=1;
else chi_s2_P=(1-probchi(chi_L,1))/2;
print Chi_L chi_s2_p;quit;

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=-0.5 s2=-0.2 s3=0.02;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P*(1+M+PB0)**s2+beta3*PB0*(1+M+P)**s3))) ;
model y~binomial(n,mu);
bounds s3>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output FitStatistics=L_full;
run;
proc nlmixed data=total corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=-0.5 s2=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**s1 +
beta2*P*(1+M+PB0)**s2+beta3*PB0))) ;
model y~binomial(n,mu);

```

```

bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output FitStatistics=L_reduced;
run;

proc iml;
use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];

if chi_L-0<1e-5 then chi_s3_P=1;
else chi_s3_P=(1-probchi(chi_L,1))/2;
print Chi_L chi_s3_p;quit;

```

B.4 $H_{0,i} : \delta_i = 0$ vs. $H_{1,i} : \delta_m > 0$ in a Tertiary Mixture Study

B.4.1 Wald Test Statistic

```

Data total;
set total;
I1=1;I2=0;
run;

%Macro LOOCV;
%Do i=267 %TO 290;
Data Complete_LOO;
set total;
if ID=&i then do;I1=0;I2=1;end;
run;

Data pred_One;
set total;

```

```

if ID=&i;
run;

%if (&i=286) %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5
s2=-3 s3=0.5 delta=5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1+
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
%end;

%else %if (&i=274 or &i=279 or &i=280) %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5
s2=-3 s3=0.5 delta=5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1+
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
%end;

```

```

%else %if (&i=285 or &i=290) %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8
s1=-5 s2=-3 s3=10 delta=1.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)
**S1+ beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
%end;
%else %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005
beta3=1 s1=0.5 s2=-0.05 s3=0.2 delta=5;
*parms beta0=-3 beta1=5 beta2=5 beta3=0.8
s1=-5 s2=-3 s3=0.5 mu2=0.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1+
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
run;
%end;

```

```

Data add;
set complete_L00;
if I2=1;
run;
Data mix;
set complete_L00;
if I1=1;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1]; s2=para[6,1]; s3=para[7,1]; delta=para[8,1];
use add;
read all into add ;
use mix;
read all into mix;

dose1=add[1,1];dose2=add[1,2]; dose3=add[1,3];
y=add[1,4];m=add[1,5];
exp_a=exp(-(beta0+beta1*dose1+beta2*dose2+beta3*dose3+delta));
if exp_a<1E-15 then exp_a=1E-15;
p=1/(1+exp_a);
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp_a/(1+exp_a)**2;
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_a=der_p*der_e*1;
s_beta1_a=der_p*der_e*dose1;
s_beta2_a=der_p*der_e*dose2;
s_beta3_a=der_p*der_e*dose3;
s_s1_a=0;
s_s2_a=0;
s_s3_a=0;
s_delta_a=der_p*der_e*1;

```

```

I11_a=der_2e*der_2p;
I12_a=der_2e*der_2p*dose1;
I13_a=der_2e*der_2p*dose2;
I14_a=der_2e*der_2p*dose3;
I15_a=0;I16_a=0;I17_a=0;I18_a=der_2e*der_2p*1;

I22_a=der_2e*dose1*der_2p*dose1;
I23_a=der_2e*dose1*der_2p*dose2;
I24_a=der_2e*dose1*der_2p*dose3;
I25_a=0;I26_a=0;I27_a=0;I28_a=der_2e*dose1*der_2p*1;

I33_a=der_2e*dose2*der_2p*dose2;
I34_a=der_2e*dose2*der_2p*dose3;
I35_a=0;I36_a=0;I37_a=0;I38_a=der_2e*dose2*der_2p*1;

I44_a=der_2e*dose3*der_2p*dose3;
I45_a=0;I46_a=0;I47_a=0;I48_a=der_2e*dose3*der_2p*1;

I55_a=0;I56_a=0;I57_a=0;I58_a=0;

I66_a=0;I67_a=0;I68_a=0;

I77_a=0;I78_a=0;

I88_a=der_2e*der_2p*1;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;I18=I11;
      I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;I28=I11;
      I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;I38=I11;
      I44=I11;I45=I11;I46=I11;I47=I11;I48=I11;
      I55=I11;I56=I11;I57=I11;I58=I11;

```

```

I66=I11;I67=I11;I68=I11;
I77=I11;I78=I11;
I88=I11;

```

```

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3]; y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*
(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*
(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*
(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*
((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*
((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);

```

```

I16[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I18[i,1]=0;

I22[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*beta1*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*
((1+dose2+dose3)**s1)*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I28[i,1]=0;

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I38[i,1]=0;

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)

```

```
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I48[i,1]=0;
```

```
I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I58[i,1]=0;
```

```
I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I68[i,1]=0;
```

```
I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I78[i,1]=0;
```

```
I88[i,1]=0;
end;
```

```
s=J(8,1,0);
s[1,1]=sum(s_beta0_m)+s_beta0_a;
s[2,1]=sum(s_beta1_m)+s_beta1_a;
s[3,1]=sum(s_beta2_m)+s_beta2_a;
s[4,1]=sum(s_beta3_m)+s_beta3_a;
s[5,1]=sum(s_s1_m)+s_s1_a;
s[6,1]=sum(s_s2_m)+s_s2_a;
s[7,1]=sum(s_s3_m)+s_s3_a;
s[8,1]=s_delta_a;
```

```
I_11=sum(I11)+I11_a; I_12=sum(I12)+I12_a; I_13=sum(I13)+I13_a;
I_14=sum(I14)+I14_a; I_15=sum(I15)+I15_a; I_16=sum(I16)+I16_a;
I_17=sum(I17)+I17_a; I_18=sum(I18)+I18_a;
I_21=I_12; I_22=sum(I22)+I22_a; I_23=sum(I23)+I23_a;
I_24=sum(I24)+I24_a; I_25=sum(I25)+I25_a; I_26=sum(I26)+I26_a;
I_27=sum(I27)+I27_a; I_28=sum(I28)+I28_a;
```

```

I_31=I_13;          I_32=I_23;          I_33=sum(I33)+I33_a;
I_34=sum(I34)+I34_a; I_35=sum(I35)+I35_a; I_36=sum(I36)+I36_a;
I_37=sum(I37)+I37_a; I_38=sum(I38)+I38_a;
I_41=I_14;          I_42=I_24;          I_43=I_34;
I_44=sum(I44)+I44_a; I_45=sum(I45)+I45_a; I_46=sum(I46)+I46_a;
I_47=sum(I47)+I47_a; I_48=sum(I48)+I48_a;
I_51=I_15;          I_52=I_25;          I_53=I_35;
I_54=I_45;          I_55=sum(I55)+I55_a; I_56=sum(I56)+I56_a;
I_57=sum(I57)+I57_a; I_58=sum(I58)+I58_a;
I_61=I_16;          I_62=I_26;          I_63=I_36;
I_64=I_46;          I_65=I_56;          I_66=sum(I66)+I66_a;
I_67=sum(I67)+I67_a; I_68=sum(I68)+I68_a;
I_71=I_17;          I_72=I_27;          I_73=I_37;
I_74=I_47;          I_75=I_57;          I_76=I_67;
I_77=sum(I77)+I77_a; I_78=sum(I78)+I78_a;
I_81=I_18;          I_82=I_28;          I_83=I_38;
I_84=I_48;          I_85=I_58;          I_86=I_68;
I_87=I_78;          I_88=sum(I88)+I88_a;

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17||I_18)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27||I_28)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37||I_38)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47||I_48)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57||I_58)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67||I_68)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77||I_78)//
      (I_81||I_82||I_83||I_84||I_85||I_86||I_87||I_88);
print I_n;

I_ninv=inv(I_n);

/*if s[8,1]=. then do;
x=&i; print x exp_a;
end;*/

Inv_delta=I_ninv[8,8];
chi_W=delta**2/inv_delta;
if Chi_w-0<1e-5 then chi_p=1;
else Chi_P=(1-probchi(Chi_W,1))/2;

use pred_one;
read all into one;

```

```

p_value=one[,1:5]||chi_w||chi_p||one[,6]||Inv_delta;
create p_value from p_value[colname={'dose1','dose2',
'dose3','y','n','chi_W','pvalue','ID','inv_88'}];
append from p_value;

%IF &i=267 %Then %DO;
Data pvalue_W;
set P_value;run;

%END;
%ELSE %DO;
Data pvalue_W;
set pvalue_W p_value;

%END;

%END;

%mend LOOCV;

%loocv;

proc sort data=pvalue_W;
by DESCENDING pvalue;
run;
proc print data=pvalue_W;run;

```

B.4.2 Score Test Statistic

```

Data total;
set total;
I1=1;I2=0;
run;

%Macro LOOCV;
%Do i=267 %TO 290;
Data Complete_L00;
set total;

```

```

if ID=&i then do;I1=0;I2=1;end;
run;

Data pred_One;
set total;
if ID=&i;
run;

%if &i=286 %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5 s2=-3 s3=0.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1
+ beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+ I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0)))
;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output ParameterEstimates=para_Est;
run;
%end;

%else %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=0.5 s2=-0.05 s3=0.2;
*parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5 s2=-3 s3=0.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1 + beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+ I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0)));
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output ParameterEstimates=para_Est;
run;
%end;

```

```

Data add;
set complete_L00;
if I2=1;
run;
Data mix;
set complete_L00;
if I1=1;
run;

proc iml;

use para_Est;
read all into para;
beta0=para[1,1];
beta1=para[2,1]; beta2=para[3,1]; beta3=para[4,1];
s1=para[5,1]; s2=para[6,1]; s3=para[7,1]; delta=0;
use add;
read all into add ;
use mix;
read all into mix;

dose1=add[1,1];dose2=add[1,2]; dose3=add[1,3];
y=add[1,4];m=add[1,5];
exp_a=exp(-(beta0+beta1*dose1+beta2*dose2+beta3*dose3+delta));
if exp_a<1E-15 then exp_a=1E-15;
p=1/(1+exp_a);
der_p=(y/m-p)/(p*(1-p)/m);
der_e=exp_a/(1+exp_a)**2;
der_2p=m/(p*(1-p));
der_2e=(der_e)**2;

s_beta0_a=der_p*der_e*1;
s_beta1_a=der_p*der_e*dose1;
s_beta2_a=der_p*der_e*dose2;
s_beta3_a=der_p*der_e*dose3;
s_s1_a=0;
s_s2_a=0;
s_s3_a=0;
s_delta_a=der_p*der_e*1;

```

```

I11_a=der_2e*der_2p;
I12_a=der_2e*der_2p*dose1;
I13_a=der_2e*der_2p*dose2;
I14_a=der_2e*der_2p*dose3;
I15_a=0;I16_a=0;I17_a=0;I18_a=der_2e*der_2p*1;

I22_a=der_2e*dose1*der_2p*dose1;
I23_a=der_2e*dose1*der_2p*dose2;
I24_a=der_2e*dose1*der_2p*dose3;
I25_a=0;I26_a=0;I27_a=0;I28_a=der_2e*dose1*der_2p*1;

I33_a=der_2e*dose2*der_2p*dose2;
I34_a=der_2e*dose2*der_2p*dose3;
I35_a=0;I36_a=0;I37_a=0;I38_a=der_2e*dose2*der_2p*1;

I44_a=der_2e*dose3*der_2p*dose3;
I45_a=0;I46_a=0;I47_a=0;I48_a=der_2e*dose3*der_2p*1;

I55_a=0;I56_a=0;I57_a=0;I58_a=0;

I66_a=0;I67_a=0;I68_a=0;

I77_a=0;I78_a=0;

I88_a=der_2e*der_2p*1;

n_mix=nrow(mix);
s_beta0_m=J(n_mix,1,0);
s_beta1_m=J(n_mix,1,0);
s_beta2_m=J(n_mix,1,0);
s_beta3_m=J(n_mix,1,0);
s_s1_m=J(n_mix,1,0);
s_s2_m=J(n_mix,1,0);
s_s3_m=J(n_mix,1,0);

I11=s_beta0_m; I12=I11;I13=I11;I14=I11;I15=I11;I16=I11;I17=I11;I18=I11;
I22=I11;I23=I11;I24=I11;I25=I11;I26=I11;I27=I11;I28=I11;
I33=I11;I34=I11;I35=I11;I36=I11;I37=I11;I38=I11;
I44=I11;I45=I11;I46=I11;I47=I11;I48=I11;
I55=I11;I56=I11;I57=I11;I58=I11;

```

```

I66=I11;I67=I11;I68=I11;
I77=I11;I78=I11;
I88=I11;

```

```

do i=1 to n_mix;
dose1=mix[i,1];dose2=mix[i,2]; dose3=mix[i,3];
y=mix[i,4];m=mix[i,5];
exp=exp(-(beta0+beta1*dose1*(1+dose2+dose3)**s1+
          beta2*dose2*(1+dose1+dose3)**s2+
          beta3*dose3*(1+dose1+dose2)**s3));

p=1/(1+exp);
der_p=(y-m*p)/(p*(1-p));der_2p=m/(p*(1-p));
der_e=exp/((1+exp)**2);der_2e=(der_e)**2;

s_beta0_m[i,1]=der_p*der_e;
s_beta1_m[i,1]=der_p*der_e*dose1*(1+dose2+dose3)**s1;
s_beta2_m[i,1]=der_p*der_e*dose2*(1+dose1+dose3)**s2;
s_beta3_m[i,1]=der_p*der_e*dose3*(1+dose1+dose2)**s3;
s_s1_m[i,1]=der_p*der_e*beta1*dose1*
(log(1+dose2+dose3))*((1+dose2+dose3)**s1);
s_s2_m[i,1]=der_p*der_e*beta2*dose2*
(log(1+dose1+dose3))*((1+dose1+dose3)**s2);
s_s3_m[i,1]=der_p*der_e*beta3*dose3*
(log(1+dose1+dose2))*((1+dose1+dose2)**s3);

I11[i,1]=der_2p*der_2e;
I12[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1);
I13[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2);
I14[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3);
I15[i,1]=der_2e*der_2p*beta1*dose1*
((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I16[i,1]=der_2e*der_2p*beta2*dose2*
((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I17[i,1]=der_2e*der_2p*beta3*dose3*
((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I18[i,1]=0;

```

```

I22[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose1*((1+dose2+dose3)**s1);
I23[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose2*((1+dose1+dose3)**s2);
I24[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*dose3*((1+dose1+dose2)**s3);
I25[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I26[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I27[i,1]=der_2e*der_2p*dose1*((1+dose2+dose3)**s1)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I28[i,1]=0;

I33[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose2*((1+dose1+dose3)**s2);
I34[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*dose3*((1+dose1+dose2)**s3);
I35[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I36[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I37[i,1]=der_2e*der_2p*dose2*((1+dose1+dose3)**s2)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I38[i,1]=0;

I44[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*dose3*((1+dose1+dose2)**s3);
I45[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I46[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I47[i,1]=der_2e*der_2p*dose3*((1+dose1+dose2)**s3)
*beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I48[i,1]=0;

I55[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta1*dose1*((1+dose2+dose3)**s1)*log(1+dose2+dose3);
I56[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)

```

```

*log(1+dose2+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I57[i,1]=der_2e*der_2p*beta1*dose1*((1+dose2+dose3)**s1)
*log(1+dose2+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I58[i,1]=0;

I66[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta2*dose2*((1+dose1+dose3)**s2)*log(1+dose1+dose3);
I67[i,1]=der_2e*der_2p*beta2*dose2*((1+dose1+dose3)**s2)
*log(1+dose1+dose3)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I68[i,1]=0;

I77[i,1]=der_2e*der_2p*beta3*dose3*((1+dose1+dose2)**s3)
*log(1+dose1+dose2)*
beta3*dose3*((1+dose1+dose2)**s3)*log(1+dose1+dose2);
I78[i,1]=0;

I88[i,1]=0;
end;

s=J(8,1,0);
s[1,1]=sum(s_beta0_m)+s_beta0_a;
s[2,1]=sum(s_beta1_m)+s_beta1_a;
s[3,1]=sum(s_beta2_m)+s_beta2_a;
s[4,1]=sum(s_beta3_m)+s_beta3_a;
s[5,1]=sum(s_s1_m)+s_s1_a;
s[6,1]=sum(s_s2_m)+s_s2_a;
s[7,1]=sum(s_s3_m)+s_s3_a;
s[8,1]=s_delta_a;

I_11=sum(I11)+I11_a; I_12=sum(I12)+I12_a; I_13=sum(I13)+I13_a;
I_14=sum(I14)+I14_a; I_15=sum(I15)+I15_a; I_16=sum(I16)+I16_a;
I_17=sum(I17)+I17_a; I_18=sum(I18)+I18_a;
I_21=I_12; I_22=sum(I22)+I22_a; I_23=sum(I23)+I23_a;
I_24=sum(I24)+I24_a; I_25=sum(I25)+I25_a; I_26=sum(I26)+I26_a;
I_27=sum(I27)+I27_a; I_28=sum(I28)+I28_a;
I_31=I_13; I_32=I_23; I_33=sum(I33)+I33_a;
I_34=sum(I34)+I34_a; I_35=sum(I35)+I35_a; I_36=sum(I36)+I36_a;

```

```

I_37=sum(I37)+I37_a;I_38=sum(I38)+I38_a;
I_41=I_14;          I_42=I_24;          I_43=I_34;
I_44=sum(I44)+I44_a;I_45=sum(I45)+I45_a;I_46=sum(I46)+I46_a;
I_47=sum(I47)+I47_a;I_48=sum(I48)+I48_a;
I_51=I_15;          I_52=I_25;          I_53=I_35;
I_54=I_45;          I_55=sum(I55)+I55_a;I_56=sum(I56)+I56_a;
I_57=sum(I57)+I57_a;I_58=sum(I58)+I58_a;
I_61=I_16;          I_62=I_26;          I_63=I_36;
I_64=I_46;          I_65=I_56;          I_66=sum(I66)+I66_a;
I_67=sum(I67)+I67_a;I_68=sum(I68)+I68_a;
I_71=I_17;          I_72=I_27;          I_73=I_37;
I_74=I_47;          I_75=I_57;          I_76=I_67;
I_77=sum(I77)+I77_a;I_78=sum(I78)+I78_a;
I_81=I_18;          I_82=I_28;          I_83=I_38;
I_84=I_48;          I_85=I_58;          I_86=I_68;
I_87=I_78;          I_88=sum(I88)+I88_a;

```

```

I_n=(I_11||I_12||I_13||I_14||I_15||I_16||I_17||I_18)//
      (I_21||I_22||I_23||I_24||I_25||I_26||I_27||I_28)//
      (I_31||I_32||I_33||I_34||I_35||I_36||I_37||I_38)//
      (I_41||I_42||I_43||I_44||I_45||I_46||I_47||I_48)//
      (I_51||I_52||I_53||I_54||I_55||I_56||I_57||I_58)//
      (I_61||I_62||I_63||I_64||I_65||I_66||I_67||I_68)//
      (I_71||I_72||I_73||I_74||I_75||I_76||I_77||I_78)//
      (I_81||I_82||I_83||I_84||I_85||I_86||I_87||I_88);

```

```

/*if s[8,1]=. then do;
x=&i; print x exp_a;
end;*/

```

```

I_ninv=inv(I_n);
chi=I_ninv[8,8]*s_delta_a**2;
if s_delta_a<0 then chi_P=1;
else if chi-0<1e-5 then chi_p=1;
else Chi_P=(1-probchi(Chi,1))/2;

```

```

use pred_one;
read all into one;

```

```

p_value=one[,1:5]||chi||chi_p||one[,6]||s_delta_a;
create p_value from p_value[colname={'dose1','dose2',
'dose3','y','n','chi_s','pvalue','ID','s_delta'}];

```

```

append from p_value;

%IF &i=267 %Then %DO;
Data pvalue_score;
set P_value;run;

%END;
%ELSE %DO;
Data pvalue_score;
set pvalue_score p_value;

%END;

%END;
%mend LOOCV;

%loocv;

proc sort data=pvalue_score;
by DESCENDING pvalue;
run;
proc print data=pvalue_score;run;

```

B.4.3 Likelihood Ratio Test

```

proc nlmixed data=total corr cov;
parms beta0=-3 beta1=15 beta2=1.5 beta3=2
s1=0.5 s2=-0.05 s3=0.2;
mu=1/(1+exp(-(beta0+beta1*M*(1+P+PBO)**S1 +
beta2*P*(1+M+PBO)**S2
+beta3*PBO*(1+M+P)**S3
))) ;
model y~binomial(n,mu);
run;

Data total;
set total;
MPPBO=M*P*PBO;
add=(M=0)+(P=0)+(PBO=0);run;

```

```

proc sort data=total;
by MPPBO M P PBO;run;
Data total;
set total;
ID=_N_;
I1=1;I2=0;
run;

%Macro L00CV;
%Do i=267 %T0 290;
Data Complete_L00;
set total;
if ID_=&i then do;I1=0;I2=1;end;
run;

Data pred_One;
set total;
if ID_=&i;
run;

%if (&i=289) %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.1 beta2=0.1 beta3=0.8
s1=2 s2=-3 s3=2 delta=30;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PBO)**S1+
beta2*P*(1+M+PBO)**S2
+beta3*PBO*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PBO+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
run;

proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.1 beta2=0.1 beta3=0.8
s1=2 s2=-3 s3=2;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PBO)**S1

```

```

+ beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+ I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0)))
;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
ods output FitStatistics=L_reduced;
run;
%end;

```

```

%Else %if (&i=283 ) %then %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5
s2=-3 s3=0.5 delta=0.02;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1+
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
run;

```

```

proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5 s2=-3 s3=0.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1 + beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+ I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0)))
;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;

```

```

ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
ods output FitStatistics=L_reduced;
run;
%end;

%else %do;
proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=0.5 s2=-0.05 s3=0.2 delta=2;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1+
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0+delta)));
model y~binomial(n,mu);
bounds delta>=0;
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
title "ID=&i";
ods output FitStatistics=L_full;
run;

proc nlmixed data=complete_L00 corr cov;
parms beta0=-3 beta1=0.05 beta2=0.005 beta3=1
s1=0.5 s2=-0.05 s3=0.2;
*parms beta0=-3 beta1=5 beta2=5 beta3=0.8 s1=-5
s2=-3 s3=0.5;
mu=I1*1/(1+exp(-(beta0+beta1*M*(1+P+PB0)**S1 +
beta2*P*(1+M+PB0)**S2
+beta3*PB0*(1+M+P)**S3)))
+ I2*1/(1+exp(-(beta0+beta1*M+beta2*P+beta3*PB0)))
;
model y~binomial(n,mu);
bounds beta1>0;
bounds beta2>0;
bounds beta3>0;
ods output CovMatParmEst=cov_Est;
ods output ParameterEstimates=para_Est;
ods output FitStatistics=L_reduced;
run;

```

```

%end;

Data add;
set complete_L00;
if I2=1;
run;
Data mix;
set complete_L00;
if I1=1;
run;

proc iml;

use L_full;
read all into l_full;
use L_reduced;
read all into l_r;

chi_L=l_r[1,1]-l_full[1,1];
if chi_L-0<1e-5 then chi_L_P=1;
else chi_L_P=(1-probchi(chi_L,1))/2;

use pred_one;
read all into one;

p_valueLik=one[,1:5]||chi_L||chi_L_p||one[,6];
create pvalueL from p_valueLik[colname={'dose1',
'dose2', 'dose3', 'y', 'n', 'chi_1', 'pvalue', 'ID'}];
append from p_valueLik;
quit;

%IF &i=267 %Then %DO;
Data pvalue_lik;
set PvalueL;run;

%END;
%ELSE %DO;
Data pvalue_Lik;
set Pvalue_Lik pvalueL;run;

%END;

```

```
%END;  
%mend LOOCV;  
  
%loocv;  
  
proc sort data=pvalue_Lik;  
by descending pvalue;run;  
  
proc print data=pvalue_Lik;run;
```

Appendix C

Real Data Sets

C.1 Amethopterin and 6-mercaptopurine Data Set

/*Please note dose1 represents x1, dose2 represents x2,
y represents wk*yk and n represents
wk in our notation in Chapter 6*/

dose1 (mg/kg)	dose2 (mg/kg)	y	n	ID
38	0	0	8	1
50	0	0	8	2
67	0	1	8	3
90	0	0	8	4
120	0	1	8	5
160	0	1	8	6
213	0	6	8	7
284	0	5	8	8
379	0	7	8	9
506	0	8	8	10
675	0	8	8	11
900	0	8	8	12
0	90	0	8	13
0	120	0	8	14
0	160	0	8	15

0	213	0	8	16
0	284	0	8	17
0	379	2	8	18
0	506	3	8	19
0	675	2	8	20
0	900	6	8	21
0	1200	8	8	22
0	1600	8	8	23
38	5.1	0	8	24
50	6.7	0	8	25
67	9	0	8	26
90	12	1	8	27
120	16	2	8	28
160	21	2	8	29
213	28	5	8	30
284	38	6	8	31
379	50	7	8	32
506	67	8	8	33
675	90	8	8	34
28	28	1	8	35
38	38	1	8	36
50	50	2	8	37
67	67	0	8	38
90	90	0	8	39
120	120	4	8	40
160	160	5	8	41
213	213	7	8	42
284	284	8	8	43
379	379	8	8	44
506	506	8	8	45
675	675	8	8	46
9	67	0	8	47
12	90	0	8	48
16	120	0	8	49
21	160	1	8	50
28	213	4	8	51
38	284	7	8	52
50	379	7	8	53
67	506	7	8	54
90	675	8	8	55
120	900	8	8	56
160	1200	8	8	57

C.2 Malathion and Parathion Data Set

/*Please note y represents wk*yk and n represents wk in our notation in Chapter 6*/

dose1 (uM)	dose2 (uM)	y	n	ID
0.0053	0	0	20	1
0.0063	0	1	20	2
0.0074	0	4	20	3
0.0087	0	6	20	4
0.0102	0	17	20	5
0.012	0	20	20	6
0	0.006	0	20	7
0	0.0071	0	20	8
0	0.0084	0	20	9
0	0.0098	13	20	10
0	0.0116	20	20	11
0	0.0136	20	20	12
0.00355	0.002	0	20	13
0.00418	0.0024	0	20	14
0.00491	0.0028	0	20	15
0.00578	0.0033	10	20	16
0.0068	0.0039	19	20	17
0.008	0.0045	20	20	18
0.0027	0.003	0	20	19
0.0031	0.0035	0	20	20
0.0037	0.0042	0	20	21
0.0043	0.0049	6	20	22
0.0051	0.0058	20	20	23
0.006	0.0068	20	20	24
0.0018	0.004	0	20	25
0.0021	0.0047	0	20	26
0.0025	0.0056	0	20	27
0.0029	0.0066	10	20	28
0.0034	0.0077	15	20	29
0.004	0.0091	20	20	30

C.3 Tramadol and Acetaminophen Data Set

/*Please note dose1 represents x1, dose2 represents dose2, y represents wk*yk, and n represents wk in our notation in Chapter 6*/

dose1 (mg/kg po)	dose2 (mg/kg po)	y	n	ID
2	0	3	15	1
3	0	4	15	2
4	0	14	45	3
6	0	20	45	4
8	0	40	60	5
10	0	15	15	6
16	0	14	15	7
0	30	2	15	8
0	40	12	43	9
0	50	1	15	10
0	60	8	30	11
0	80	23	60	12
0	100	13	30	13
0	120	13	30	14
0	160	10	30	15
0	200	13	25	16
0	240	14	25	17
0	400	12	15	18
0	800	13	15	19
3.75	0.00375	1	13	20
7.5	0.0075	8	15	21
15	0.0015	15	15	22
1.875	0.01875	0	15	23
3.75	0.0375	5	15	24
7.5	0.075	5	15	25
15	0.15	15	15	26
1.875	0.09375	0	15	27
3.75	0.1875	4	15	28
7.5	0.375	7	15	29
15	0.75	15	15	30
3.75	1.25	3	30	31
7.5	2.5	12	30	32

15	5	28	30	33
0.94	0.94	3	15	34
1.875	1.875	8	30	35
3.75	3.75	14	30	36
5	5	12	28	37
7.5	7.5	24	30	38
15	15	15	15	39
3.75	11.25	7	30	40
5	15	7	15	41
7.5	22.5	29	30	42
2.5	12.5	7	30	43
5	25	8	30	44
10	50	30	30	45
0.47	2.66	0	15	46
0.94	5.313	4	15	47
1.88	10.625	1	15	48
3.75	21.25	5	15	49
7.5	42.5	11	15	50
15	85	15	15	51
0.94	17.813	4	30	52
1.88	36.625	10	28	53
3.75	71.25	21	30	54
5	95	22	30	55
7.5	142.5	29	30	56
15	285	15	15	57
0.25	12.5	3	30	58
0.5	25	7	30	59
1	50	9	30	60
2	100	19	30	61
4	200	27	30	62
8	400	30	30	63
0.25	25	3	60	64
0.5	50	12	60	65
1	100	19	60	66
2	200	51	60	67
4	400	55	60	68
8	800	30	30	69
0.125	25	1	60	70
0.25	50	9	60	71
0.5	100	27	60	72
1	200	44	60	73
2	400	48	60	74

4	800	30	30	75
0.0625	25	2	15	76
0.125	50	4	30	77
0.25	100	18	30	78
0.5	200	12	15	79
1	400	28	30	80
2	800	15	15	81
0.03125	25	4	30	82
0.0625	50	9	30	83
0.125	100	15	30	84
0.25	200	27	30	85
0.5	400	30	30	86
0.03125	50	2	30	87
0.0625	100	14	30	88
0.125	200	22	30	89
0.25	400	27	30	90
0.5	800	30	30	91

C.4 Marathion, Parathion and Piperonyl Butoxide Data Set

/*Please note y represents wk**yk*, n represents wk, M represents x1, P represents x2, and PBO represents x3 in our notation in Chapter 6.*/

M (uM)	P (uM)	PBO (uM)	y	n	ID
0	0	2.4	0	20	1
0	0	2.8	0	20	2
0	0	3.3	0	20	3
0	0	3.9	1.5	20	4
0	0	4.6	1	20	5
0	0	5.4	3.5	20	6
0	0	6.4	10	20	7
0	0	7.5	17.5	20	8
0	0	8.9	19.5	20	9
0	0	10.4	19.5	20	10

0	0	12.3	20	20	11
0	0	14.5	20	20	12
0	0	17	20	20	13
0	0	20	20	20	14
0	0.006	0	0	20	15
0	0.0062	0	0	20	16
0	0.0068	0	0	20	17
0	0.0071	0	0	20	18
0	0.0074	0	0	20	19
0	0.008	0	0	20	20
0	0.0084	0	0	20	21
0	0.0087	0	0	20	22
0	0.0089	0	0	20	23
0	0.0089	0	0	20	24
0	0.0089	0.00025	0	20	25
0	0.0089	0.0005	0	20	26
0	0.0089	0.001	0	20	27
0	0.0089	0.002	0	20	28
0	0.0089	0.0025	0	20	29
0	0.0089	0.005	0	20	30
0	0.0089	0.01	0	20	31
0	0.0089	0.05	0	20	32
0	0.0094	0	0	20	33
0	0.0098	0	13	20	34
0	0.0103	0	0	20	35
0	0.0104	0	3	20	36
0	0.0104	0	6	20	37
0	0.0104	0.00025	3	20	38
0	0.0104	0.0005	0	20	39
0	0.0104	0.001	0	20	40
0	0.0104	0.002	0	20	41
0	0.0104	0.0025	0	20	42
0	0.0104	0.005	0	20	43
0	0.0104	0.01	0	20	44
0	0.0104	0.05	1	20	45
0	0.0112	0	7	20	46
0	0.0116	0	20	20	47
0	0.012	1.2	0	20	48
0	0.0121	0	17	20	49
0	0.0123	0	16	20	50
0	0.0123	0	20	20	51
0	0.0123	0.00025	19	20	52

0	0.0123	0.0005	20	20	53
0	0.0123	0.001	6	20	54
0	0.0123	0.002	0	20	55
0	0.0123	0.0025	0	20	56
0	0.0123	0.005	0	20	57
0	0.0123	0.01	0	20	58
0	0.0123	0.05	13	20	59
0	0.0132	0	19	20	60
0	0.0136	0	20	20	61
0	0.0143	0	20	20	62
0	0.0145	0	20	20	63
0	0.0145	0	20	20	64
0	0.0145	0.00025	20	20	65
0	0.0145	0.0005	20	20	66
0	0.0145	0.001	20	20	67
0	0.0145	0.002	18	20	68
0	0.0145	0.0025	12	20	69
0	0.0145	0.005	3	20	70
0	0.0145	0.01	0	20	71
0	0.0145	0.05	20	20	72
0	0.015	1.2	0	20	73
0	0.0156	0	20	20	74
0	0.0169	0	20	20	75
0	0.017	0	20	20	76
0	0.017	0	20	20	77
0	0.017	0.00025	20	20	78
0	0.017	0.0005	20	20	79
0	0.017	0.001	20	20	80
0	0.017	0.002	20	20	81
0	0.017	0.0025	20	20	82
0	0.017	0.005	20	20	83
0	0.017	0.01	20	20	84
0	0.017	0.05	20	20	85
0	0.0184	0	20	20	86
0	0.019	1.2	0	20	87
0	0.02	0	20	20	88
0	0.02	0	20	20	89
0	0.02	0	20	20	90
0	0.02	0.00025	20	20	91
0	0.02	0.0005	20	20	92
0	0.02	0.001	20	20	93
0	0.02	0.002	20	20	94

0	0.02	0.0025	20	20	95
0	0.02	0.005	20	20	96
0	0.02	0.01	20	20	97
0	0.02	0.05	20	20	98
0	0.0226	0.25	4	20	99
0	0.024	1.2	0	20	100
0	0.0266	0.25	2	20	101
0	0.0266	0.5	0	20	102
0	0.03	1.2	0	20	103
0	0.031	0.78	0	20	104
0	0.0313	0.25	19	20	105
0	0.0313	0.5	3	20	106
0	0.0313	1.2	0	20	107
0	0.0313	2	0	20	108
0	0.035	1.3	0	20	109
0	0.0368	0.25	20	20	110
0	0.0368	0.5	18	20	111
0	0.0369	1.2	0	20	112
0	0.0369	2	0	20	113
0	0.037	0.78	0	20	114
0	0.042	1.3	0	20	115
0	0.043	0.78	1	20	116
0	0.0434	0.25	20	20	117
0	0.0434	0.5	20	20	118
0	0.0434	1.2	15	20	119
0	0.0434	2	0	20	120
0	0.044	2.16	0	20	121
0	0.049	1.3	0	20	122
0	0.05	1.2	12	20	123
0	0.051	0.25	20	20	124
0	0.051	0.5	20	20	125
0	0.051	0.78	7	20	126
0	0.051	1.2	13	20	127
0	0.051	2	2	20	128
0	0.052	2.16	0	20	129
0	0.055	3.6	0	20	130
0	0.058	1.3	1	20	131
0	0.06	0.5	20	20	132
0	0.06	0.78	20	20	133
0	0.06	1.2	20	20	134
0	0.06	2	6	20	135
0	0.061	2.16	0	20	136

0	0.064	3.6	2	20	137
0	0.068	1.3	20	20	138
0	0.07	0.78	20	20	139
0	0.0706	1.2	20	20	140
0	0.0706	2	5	20	141
0	0.072	2.16	14	20	142
0	0.076	3.6	3	20	143
0	0.0767	2.87	12	20	144
0	0.08	1.3	20	20	145
0	0.085	2.16	20	20	146
0	0.089	3.6	20	20	147
0	0.1	2.16	20	20	148
0	0.105	3.6	20	20	149
0	0.123	3.6	20	20	150
0	0.1281	4.79	19	20	151
0	0.214	8	20	20	152
0.0018	0.004	0	0	20	153
0.0021	0.0047	0	0	20	154
0.0025	0.0056	0	0	20	155
0.0027	0.003	0	0	20	156
0.0029	0.0066	0	10	20	157
0.0031	0.0035	0	0	20	158
0.0034	0.0077	0	15	20	159
0.00355	0.002	0	0	20	160
0.0037	0.0042	0	0	20	161
0.004	0.0091	0	20	20	162
0.00418	0.0024	0	0	20	163
0.0043	0.0049	0	6	20	164
0.0046	0	0	0	20	165
0.0046	0	0	0	20	166
0.00491	0.0028	0	0	20	167
0.0051	0	0	0	20	168
0.0051	0	0	0	20	169
0.0051	0.0058	0	20	20	170
0.0053	0	0	0	20	171
0.0056	0	0	0	20	172
0.0056	0	0	0	20	173
0.00578	0.0033	0	10	20	174
0.006	0.0068	0	20	20	175
0.0063	0	0	0	20	176
0.0063	0	0	0	20	177
0.0063	0	0	1	20	178

0.0068	0.0039	0	19	20	179
0.007	0	0	0	20	180
0.007	0	0	0	20	181
0.0074	0	0	4	20	182
0.0077	0	0	0	20	183
0.0077	0	0	1	20	184
0.008	0.0045	0	20	20	185
0.0086	0	0	0	20	186
0.0086	0	0	0	20	187
0.0087	0	0	6	20	188
0.0096	0	0	0	20	189
0.0096	0	0	4	20	190
0.0102	0	0	17	20	191
0.0106	0	0	8	20	192
0.0106	0	0	11	20	193
0.0118	0	0	19	20	194
0.0118	0	0	16	20	195
0.012	0	0	20	20	196
0.0131	0	0	16	20	197
0.0131	0	0	19	20	198
0.0131	0	0.0025	0	20	199
0.0146	0	0	20	20	200
0.0146	0	0	20	20	201
0.0162	0	0	20	20	202
0.0162	0	0	20	20	203
0.0164	0	0.0025	1	20	204
0.018	0	0	20	20	205
0.018	0	0	20	20	206
0.0197	0	0.005	0	20	207
0.02	0	0	20	20	208
0.02	0	0	20	20	209
0.0205	0	0.0025	12	20	210
0.0246	0	0.005	1	20	211
0.0256	0	0.0025	19	20	212
0.0262	0	0.01	0	20	213
0.0307	0	0.005	14	20	214
0.032	0	0.0025	20	20	215
0.0328	0	0.01	0	20	216
0.0328	0	0.02	0	20	217
0.0384	0	0.005	20	20	218
0.04	0	0.0025	20	20	219
0.041	0	0.01	9	20	220

0.041	0	0.02	0	20	221
0.048	0	0.005	20	20	222
0.0512	0	0.01	20	20	223
0.0512	0	0.02	1	20	224
0.06	0	0.005	20	20	225
0.064	0	0.01	20	20	226
0.064	0	0.02	20	20	227
0.0681	0	2.72	0	20	228
0.08	0	0.01	20	20	229
0.08	0	0.02	20	20	230
0.1	0	0.02	20	20	231
0.1	0	0.15	1	20	232
0.1	0	0.2	0	20	233
0.1	0	0.3	0	20	234
0.1	0	0.6	0	20	235
0.1138	0	4.55	11	20	236
0.12	0	0.15	1	20	237
0.12	0	0.2	2.2	20	238
0.12	0	0.3	0	20	239
0.12	0	0.6	0	20	240
0.13	0	1	0	20	241
0.15	0	0.15	20	20	242
0.15	0	0.2	8	20	243
0.15	0	0.3	1	20	244
0.15	0	0.6	2	20	245
0.16	0	1	0	20	246
0.19	0	0.15	20	20	247
0.19	0	0.2	20	20	248
0.19	0	0.3	16	20	249
0.19	0	0.6	1	20	250
0.19	0	1	0	20	251
0.19	0	7.6	15	20	252
0.22	0	1	0	20	253
0.24	0	0.15	20	20	254
0.24	0	0.2	20	20	255
0.24	0	0.3	20	20	256
0.24	0	0.6	9	20	257
0.26	0	1	0	20	258
0.3	0	0.15	20	20	259
0.3	0	0.2	20	20	260
0.3	0	0.3	20	20	261
0.3	0	0.6	20	20	262

0.31	0	1	1	20	263
0.36	0	1	2	20	264
0.43	0	1	18	20	265
0.5	0	1	20	20	266
0.0553	0.0066	2.46	0	20	267
0.0271	0.0193	1.81	0	20	268
0.0065	0.0623	2.59	17	20	269
0.019	0.0305	1.9	0	20	270
0.035	0.0161	2	0	20	271
0.0444	0.0119	2.22	0	20	272
0.0158	0.0395	2.11	0	20	273
0.0117	0.05	2.34	1	20	274
0.0924	0.011	4.11	4	20	275
0.0453	0.0323	3.02	0	20	276
0.0108	0.1041	4.32	16	20	277
0.0318	0.051	3.18	6	20	278
0.0585	0.0268	3.34	3	20	279
0.0741	0.0198	3.71	3	20	280
0.0264	0.0659	3.52	12	20	281
0.0195	0.0835	3.9	14	20	282
0.1543	0.0184	6.86	15	20	283
0.0756	0.054	5.04	9	20	284
0.0181	0.1738	7.22	20	20	285
0.0531	0.0852	5.31	12	20	286
0.0978	0.0448	5.59	10	20	287
0.1238	0.0331	6.19	13	20	288
0.0441	0.1101	5.88	19	20	289
0.0326	0.1395	6.52	20	20	290