2018-06

MARKOV CHAIN MONTE CARLO AND EXACT CONDITIONAL TESTS WITH THREE-WAY CONTINGENCY TABLES

Lee, Seungchan

Monterey, CA; Naval Postgraduate School

http://hdl.handle.net/10945/59705

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MARKOV CHAIN MONTE CARLO AND EXACT CONDITIONAL TESTS WITH THREE-WAY CONTINGENCY TABLES

by

Seungchan Lee

June 2018

Thesis Advisor: Ruriko Yoshida
Second Reader: Michael P. Atkinson

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** June 2018	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE MARKOV CHAIN MONTE CARLO AND EXACT CONDITIONAL TESTS WITH THREE-WAY CONTINGENCY TABLES	**5. FUNDING NUMBERS**
6. AUTHOR(S) Seungchan Lee	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

We propose an algorithm modifying a popular exact conditional test involving the goodness-of-fit of contingency tables. This study focuses on improving the efficiency of Markov chain Monte Carlo (MCMC) when sampling three-way contingency tables--defined as log-linear models with three discrete random categorical variables consisting of finite levels--under the no-three-way interaction model. Standard to MCMC, we approximate the null distribution by sampling tables from the conditional distribution. However, our proposal involves expanding the conditional state space to include tables with cell count values of -1. We apply the proposed methodology, described in full detail, to randomly generated sparse and non-sparse data sets. Our results show that traditional asymptotic methods on sparse contingency tables yield inaccurate results. We also prove mathematically that a Markov chain with our proposed method is connected (i.e., ergodic) on the conditional state space for 3x3xK, with K >= 3. The output from applying the proposed methodology provides conclusive evidence that the distribution of the test statistics for sparse data sets does not resemble the asymptotic distribution.

14. SUBJECT TERMS Categorical Data Analysis, Markov-chain Monte Carlo, Chi-Square Test for Independence, Kolmogorov-Smirnov Test, three-way contingency tables, no-three-way interaction model, log-linear models, exact conditional test, goodness-of-fit test, sparse data, asymptotic distribution.	**15. NUMBER OF PAGES** 99
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

THIS PAGE INTENTIONALLY LEFT BLANK

MARKOV CHAIN MONTE CARLO AND EXACT CONDITIONAL TESTS WITH THREE-WAY CONTINGENCY TABLES

Seungchan Lee
Captain, United States Marine Corps
BS, U.S. Naval Academy, 2011

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 2018

Approved by: Ruriko Yoshida
Advisor

Michael P. Atkinson
Second Reader

Patricia A. Jacobs
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

We propose an algorithm modifying a popular exact conditional test involving the goodness-of-fit of contingency tables. This study focuses on improving the efficiency of Markov chain Monte Carlo (MCMC) when sampling three-way contingency tables--defined as log-linear models with three discrete random categorical variables consisting of finite levels--under the no-three-way interaction model. Standard to MCMC, we approximate the null distribution by sampling tables from the conditional distribution. However, our proposal involves expanding the conditional state space to include tables with cell count values of -1. We apply the proposed methodology, described in full detail, to randomly generated sparse and non-sparse data sets. Our results show that traditional asymptotic methods on sparse contingency tables yield inaccurate results. We also prove mathematically that a Markov chain with our proposed method is connected (i.e., ergodic) on the conditional state space for 3x3xK, with K >= 3. The output from applying the proposed methodology provides conclusive evidence that the distribution of the test statistics for sparse data sets does not resemble the asymptotic distribution.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

CDA	Categorical Data Analysis.
DTMC	Discrete Time Markov Chain
IPF	Iterative Proportional Fitting
K–S Test	Kolmogorov–Smirnov Test
MCMC	Markov chain Monte Carlo
MLE	Maximum Likelihood Estimator
SIS	Sequential Importance Sampling

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

We propose an algorithm modifying a popular exact conditional test involving the goodness-of-fit of three-way contingency tables, defined as log-linear models with three discrete random categorical variables consisting of finite levels [1]. A goodness-of-fit hypothesis test checks for significantly strong associations (i.e., correlations) between categorical variables [1]. With three-dimensional tables, we call a table where all three variables share no correlation, a no-three-way interaction model. We develop an alternative method in our research to conduct a goodness-of-fit test in order to determine whether we select a no-three-way interaction model or a more complicated model for use in Categorical Data Analysis (CDA).

We focus particularly on improving the efficiency of Markov chain Monte Carlo (MCMC) when sampling three-way contingency tables under the no-three-way interaction model. Standard to MCMC, we approximate the null distribution by sampling tables from the conditional distribution [2]. However, our proposal involves expanding the conditional state space to include tables with cell counts of -1. We apply the proposed methodology to randomly generated sparse and non-sparse data sets.

Two approaches currently account for the traditional ways of conducting CDA: asymptotic and exact. An asymptotic distribution, or limiting distribution, represents the convergence of a sequence of distributions [3]. When calculating exact statistical results prove difficult, approximating the results based on known properties or behaviors of certain statistics in large samples offers the best alternative.

However, asymptotic distributions typically require certain stipulations in order to ensure accuracy. In other words, we usually use the asymptotic distribution of the test statistics as the null distribution for goodness-of-fit tests if all of the expected cell counts exceed five [1]. Therefore, goodness-of-fit tests conducted with asymptotic methods on sparse contingency tables, or tables with mostly zeroes or values less than five, might produce inaccurate or biased results.

Fisher's exact test provides an alternative for smaller tables, such as $2 \times 2 \times 2$, that do not satisfy the conditions needed to use the asymptotic distribution [1]. Fisher's exact test

involves enumerating all possible tables in the conditional state space with the fixed sufficient statistic from the observed table. However, enumerating every table in the conditional state space may require infeasible amount of computation for large contingency tables, such as $3 \times 3 \times 3$ and beyond.

We can neither use the asymptotic distribution nor Fisher's exact test for large sparse contingency tables. Instead, we can approximate the null distribution by sampling tables from the conditional tables, in a similar fashion to the MCMC developed by Diaconis and Sturmfels [2]. Our limitations in conducting CDA with sparse data serve as the motivation behind our proposed modification to this popular MCMC sampling method. In order to efficiently and properly sample three-way contingency tables, under the no-three-way interaction model, we look to fill in the gap where no effective test currently exists.

Our proposed MCMC sampling method properly functions in sampling three-way tables, under a no-three-way interaction model, for both sparse and non-sparse tables. We also prove mathematically that the proposed method can sample a table from anywhere in the conditional state space without any bias in the case of $3 \times 3 \times K$ tables for $K \geq 3$. Our simulation results show strong evidence that for larger tables, such as $4 \times 4 \times 4$ and greater, the proposed method samples contingency tables from anywhere in the conditional state space without bias. We can deduce that our proposed MCMC method simulates a Markov chain on the connected transition graph so that it will not produce a sampling bias by observing the unimodal distribution of test statistics for sparse data; a not-well-mixed chain generally outputs a multimodal distribution of test statistics.

The results of both sparse and non-sparse data simulations allow us to conclude that the no-three-way interaction model fits well with the three-way tables sampled from the conditional distribution. For non-sparse tables, the results of our simulations show that the χ^2 distribution accurately summarizes the null distribution of the test statistics, validating our algorithm with established norms. Conversely, the results for sparse tables show that the null distribution is not well summarized by the χ^2 distribution. Thus, we can reasonably conclude that for sparse tables, the χ^2 distribution does not accurately approximate the null distribution of test statistics.

We believe our algorithm contributes to the development of accurate algorithms for CDA of sparse data and exact conditional tests. Since contingency tables will likely remain as a widely used means of analysis for CDA, we anticipate this algorithm to further the development of sampling and testing methods for MCMC. Our proposed algorithm can sample any sparse or non-sparse $I \times J \times K$ contingency table, under the no-three-way interaction model, in any field of research for conducting goodness-of-fit tests. For sparse tables, our method estimates the null distribution of test statistics more accurately than traditional methods that involve asymptotic distributions, since our results provide strong statistical evidence that traditional asymptotic methods on sparse contingency tables yield inaccurate results.

List of References

[1] A. Agresti, *Categorical Data Analysis*, 2nd ed. (Wiley Series in Probability and Statistics). Hoboken, NJ, USA: Wiley-Interscience, 2002.

[2] P. Diaconis and B. Sturmfels, “Algebraic algorithms for sampling from conditional distributions,” *Ann. Statist.*, no. 1, pp. 363–397, 2002.

[3] T. Epps, *Probability and Statistical Theory for Applied Researchers* (World Scientific Books). 27 Warren Street, Suite 401-402, Hackensack, NJ, USA: World Scientific Publishing Co. Pte. Ltd., January 2013, no. 8831. Available: https://ideas.repec.org/b/wsi/wsbook/8831.html

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

I would like to first thank my wife, Viviana Lee, for providing me with the support I needed while I navigated through this challenging and exhausting endeavor. The fact that I completed my thesis speaks more about you than it does about me. I want to also thank my feisty yet precious and adorable daughter, Ava Rey, who inadvertently encouraged me to efficiently employ all available time to my thesis when not absorbed by ensuring her survival. Thank you for reminding me of the many important lessons pertaining to appreciating the simple things in life. . . like uninterrupted sleep. I hope that you also develop a sense of humor by the time (or perhaps, if) you ever get around to reading this.

Thank you, Rudy, for your supervision, guidance, *patience*, education, and diligence. You gave me a chance to learn something in great detail, sometimes at the expense of your sanity. Your patience and composure to my rudimentary questions speaks volumes for your passion and devotion to your work. Thank you for investing in my academic development.

Thank you, Professor Atkinson, for your meticulous reviews and many important suggestions to the structural makeup of this work. I remain grateful for your quick turnarounds in returning my drafts with incredibly useful feedback. You shaped the writing to allow for a more coherent structure for the reader to follow. I learned a lot from your professionalism, diligence, and attention to detail, which I will work to emulate in my profession.

THIS PAGE INTENTIONALLY LEFT BLANK

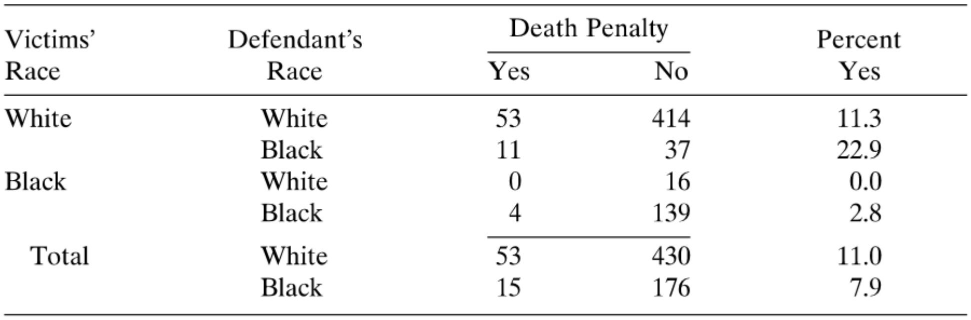
CHAPTER 1:
Introduction

This chapter provides the background and objective of our research. We will explain the technical terms and definitions used below in greater depth in Chapter 2.1. This chapter introduces the problem as well as the general concepts about the topics covered in this study.

1.1 Background

In Categorical Data Analysis (CDA), researchers typically employ contingency tables in order to identify potential associations or interrelations between multiple categorical variables. As seen in Figure 1.1, contingency tables display counts of outcomes by each categorical *level*, or unique groups within each category. This research delves into three-dimensional (or *three-way*) contingency tables, selects the appropriate model to properly analyze those tables, and proposes a modification to a popular way of collecting samples to analyze.

The following example displays a 2×2×2 contingency table with three categorical variables: victims' race, defendants' race, and death penalty [1].



Victims'	Defendant's	Death Penalty		Percent
Race	Race	Yes	No	Yes
White	White	53	414	11.3
	Black	11	37	22.9
Black	White	0	16	0.0
	Black	4	139	2.8
Total	White	53	430	11.0
	Black	15	176	7.9

Figure 1.1. Death Penalty Verdict by Defendant's Race and Victims' Race. Source: [1].

Here, the data shows the cross-classification of observations by the levels of the three aforementioned categorical variables. Through this example, we observe the relationship of race as it relates to the individuals convicted of homicide receiving the death penalty [1].

The organization of data in this structure provides researchers the format needed to answer whether or not an association exists between receiving the death penalty, defendants' race, and victims' race.

The first step in studying potential associations between variables involves establishing the appropriate model to use for analysis. Since data drives data science and statistics, we want to ensure that we select the model that best fits with the given data. We call this process of selecting the proper model the model selection.

One of the first steps in model selection includes computing the *Maximum Likelihood Estimator (MLE)*, which represents the contingency table with parameter values that will maximize the probability of getting the observed table under the given model. In order to compute the MLE, we have to calculate a *sufficient statistic* from the observed table. We define sufficient statistic as the statistic that facilitates the calculation of the MLE under a certain model. Note that many contingency tables may have the same sufficient statistic. We call this collection, or *set*, of all possible contingency tables satisfying the given sufficient statistic as the *conditional state space*. Identifying the best model will help explain all of the other tables in the conditional state space.

For example, Table 1.1 displays a simple 2×2 contingency table.

Table 1.1. Example, 2x2 Contingency Table with Fixed Variables. Table consists of 2 categorical variables: Cat1, Cat2. "Cat" shortened from Category, with 2 Levels per Cat. Marginal sums on the last row and column.

	Cat2		
Cat1	Cat2-Level1	Cat2-Level2	Total by Cat1
Cat1-Level1	1	0	1
Cat1-Level2	0	1	1
Total by Cat2	1	1	2

If we use marginal sums as the fixed sufficient statistic, we find Table 1.2 as the only other table in the conditional state space that shares the same column and row sum values.

Table 1.2. Another Example, 2x2 Contingency Table with Fixed Variables. Table consists of 2 categorical variables: Cat1, Cat2. "Cat" shortened from Category, with 2 Levels per Cat. Marginal sums on the last row and column.

	Cat2		
Cat1	Cat2-Level1	Cat2-Level2	Total by Cat1
Cat1-Level1	0	1	1
Cat1-Level2	1	0	1
Total by Cat2	1	1	2

Another step during model selection includes a test called a goodness-of-fit test. A form of hypothesis test, the goodness-of-fit test checks for significantly strong associations (i.e., correlations) between categorical variables. After computing the MLE of the *null* model, we calculate a test statistic, which we elaborate on in Chapter 2.1, to measure the closeness between each sampled table from the conditional state space and the MLE. The distribution of this test statistic forms the null distribution, which we use to observe the characteristics of the tables in the conditional state space. We also take note of what value would deem a test statistic as rare if we assume that the null model best fits the data.

A common evaluation in selecting a model for *three-way tables* includes a hypothesis test to determine whether a correlation exists between all three variables at once. In this particular case, we call a table where all three variables share no correlation a *no-three-way interaction model*. In Figure 1.1, we can use the goodness-of-fit test in order to determine whether we can use this model under which no correlation exists between race and death penalty sentencing. For the purposes of our research, we developed an alternative method to conduct a goodness-of-fit test in order to determine whether we select a no-three-way interaction model or a more complicated model.

Two approaches currently account for the traditional ways of conducting CDA: asymptotic and exact. An asymptotic distribution, or *limiting distribution*, represents the convergence of a sequence of distributions [2]. When calculating exact statistical results prove difficult, approximating the results based on known properties or behaviors of certain statistics in large samples offer the best alternative. The *asymptotic distribution theory* attempts to find this convergence through a series of distributions [2].

However, asymptotic distributions typically require certain stipulations in order to ensure accuracy. In other words, there is guidance that we should only use the asymptotic distribution of the test statistics as the null distribution for goodness-of-fit tests if all of the expected cell counts exceed five [1]. Therefore, goodness-of-fit tests conducted with asymptotic methods on *sparse contingency tables*, or tables with mostly zeroes or values less than five, might produce inaccurate or biased results.

Fisher's exact test provides an alternative for smaller tables, such as $2 \times 2 \times 2$, that do not satisfy the conditions needed to use the asymptotic distribution [1]. Fisher's exact test involves enumerating all possible tables in the conditional state space with the fixed sufficient statistic from the observed table. However, enumerating every table in the conditional state space may require infeasible amount of computation for large contingency tables, such as $3 \times 3 \times 3$ and beyond.

Since we can neither use the asymptotic distribution nor Fisher's exact test for large *sparse contingency tables*, we instead approximate the null distribution by sampling tables from the *conditional distribution*. Diaconis and Sturmfels [3] developed one of the most popular exact conditional tests through Markov chain Monte Carlo (MCMC), which samples through the conditional state space via a relatively simple procedure, which we will explain in more detail in Chapters 2.1 and 3. In this research, we propose a modification to this popular MCMC sampling method in order to efficiently and properly sample three-way contingency tables, under the no-three-way interaction model, to fill in the gap where no effective test currently exists.

1.2 Research Objectives

We look to address the following objectives in this thesis:

- Conduct MCMC sampling for three-way contingency tables under a no-three-way interaction model using the idea from Bunea and Besag [4].
- Prove mathematically that the proposed method can sample a table from anywhere in the conditional state space without any bias in the case of $3 \times 3 \times K$ tables for $K \geq 3$.
- Provide evidence through simulations that the proposed method can sample a table from anywhere in the conditional state space without any bias for any $I \times J \times K$ tables for $I,\ J,\ K \geq 3$.

- Employ our proposed algorithm on sparse three-way contingency tables, under the no-three-way interaction model, to show that the asymptotic distribution does not apply to the distribution of test statistics for sparse tables.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Definitions, Motivation, and Literature Review

The research in this thesis builds on several related studies and findings. This chapter outlines the definitions of frequently used terminology and terms. It also includes the motivation behind the research as well as the literature review of specific works this thesis builds on.

2.1 Definitions

This section defines key words, concepts, and algorithms as applied to the research of this thesis. This selection also explains in greater detail some of the concepts addressed in Chapter 1. Therefore, please note that some definitions refer to figures and tables from Chapter 1.

2.1.1 Basic Notations

We define *levels* as a finite subset of categories, *cell* as a specific event with discrete (or countable) frequency counts, and *cell count* as the number of occurrences observed for that particular event. Each cell count may reflect the outcome of a multinomial probability distribution [5]. In Figure 1.1, each *type* of race and death penalty sentencing represents the levels while *values* inside the table represent the cell count.

We also frequently use variables $\mathbb{N}$ and $\mathbb{Z}$ during definitions, equations, and algorithms. Let $\mathbb{N} = \{1, 2, \ldots\}$ (i.e., $\mathbb{N}$) represent the set of all natural numbers and $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$ (i.e., $\mathbb{Z}$) represent the set of all integers. These definitions apply throughout this thesis.

2.1.2 Contingency Tables

We define *categorical variables* as a type of data with measurement scales separable into groups or a set of categories [1]. Contingency tables display the multivariate frequency distribution, or counts of outcomes, of those categorical variables in a matrix or table format, as seen in earlier examples. Analysts use contingency tables to study potential relationships or correlations between the set of categories.

2.1.3 Poisson Distribution

This discrete probability distribution measures the probability of the count of random and mutually independent occurrences for a particular event within a specified period of time. Simply stated, we often use this distribution in order to model counts, particularly to model the arrival of events in a given time period. μ represents the parameter of the distribution which denotes the mean number of occurrences in one time period [6]. If x represents discrete number of observed occurrences over a period of time, the Poisson distribution takes the probability mass function:

$$P(x) = \frac{e^{-\mu}\mu^{x}}{x!}.$$

We draw each cell count values for the contingency tables used in this study from the Poisson distribution.

2.1.4 Log-Linear Models

As natural tools for analyzing multinomial categorical data, we use log-linear models to analyze relationships between categorical variables [7]. These models can provide information regarding potential association or interaction patterns between categorical variables [1]. Log-linear models typically assume nominal (qualitative and unordered) discrete variables, but can also work for ordinal and matched data [7]. For example, the Poisson distribution models the simple example shown in Table 1.1 with the parameter μ_{ij} such that

$$\log(\mu_{ij}) = \lambda + \lambda_{i}^{Cat1} + \lambda_{j}^{Cat2} + \lambda_{ij}^{Cat1,Cat2},$$

where i represents the levels in Cat1, j represents the levels in Cat2, and $\lambda_{ij}^{Cat1,Cat2}$ represents the relationship between the two categorical variables.

2.1.5 Maximum Likelihood Estimator

The maximum likelihood estimate of a parameter represents a value where the probability of the initial data takes on the most likely value [1]. Simply put, MLE represents the estimate of the parameters; we use these parameters in order to estimate cell counts for contingency tables.

2.1.6 Sufficient Statistics

Sufficient statistics represent values computed from a given data set such that the set of those values contains enough information to infer the MLE. For the purposes of this research, we define the sufficient statistic as the marginal sums, or the sums of each dimension's vectors, of the observed table [1], [8]. Simply put, the sufficient statistic consists of the sums of each level within each categorical variable. In Tables 1.1 and 1.2, we see the sufficient statistic represented under the "total" counts on the last row and column.

2.1.7 No-Three-Way Interaction Model

For this study, we define three-way tables as log-linear models with three discrete random variables such that each random variable consists of finite levels. For example, Figure 1.1 represents a three-way table that includes finite levels within three discrete random variables. In order to analyze a potential correlation between variables, we define our working model.

Suppose we have three categorical variables $T \in \{1, \dots, I\}$, $Y \in \{1, \dots, J\}$, and $Z \in \{1, \dots, K\}$. Also, suppose X_{ijk} represents a cell count in the ith level of T, the jth level of Y, and the kth level of Z. Under this model, the contingency table $\mathbf{X}$ has cell counts X_{ijk}, for $i = 1, \dots, I$, $j = 1, \dots, J$ and $k = 1, \dots, K$. The Poisson distribution models this table with the parameter μ_{ijk} such that

$$\log(\mu_{ijk}) = \lambda + \lambda_i^T + \lambda_j^Y + \lambda_k^Z + \lambda_{ij}^{TY} + \lambda_{ik}^{TZ} + \lambda_{jk}^{YZ} + \lambda_{ijk}^{TYZ}. \tag{2.1}$$

Note that the above forms a log-linear model for a three-way table. Furthermore, if we have

$$\lambda_{ijk}^{TYZ} = 0,$$

then we state that the three random variables have no interaction. We call this model the *no-three-way interaction model*.

Under this model, one can show that the marginal sums can serve as the *sufficient statistics* of the model [1]. We define marginal distributions as the sum of the levels within each categorical variable; in other words, we calculate the total count in each dimension without accounting for the other variables:

$$\begin{array}{rcl} X_{\cdot jk} & := & \sum_{i=1}^{I} X_{ijk}, \\ X_{i\cdot k} & := & \sum_{j=1}^{J} X_{ijk}, \\ X_{ij\cdot} & := & \sum_{k=1}^{K} X_{ijk}. \end{array}$$

Recall that we call the *set* of all possible contingency tables satisfying the given sufficient statistics the *conditional state space*. In the case of Table 1.1, we saw Table 1.2 as the only other table in the conditional state space.

2.1.8 Iterative Proportional Fitting

Iterative Proportional Fitting (IPF) involves an iterative method for estimating the MLE for a particular model of log-linear models. Specifically, IPF estimates the MLE under the no-three-way interaction model. This procedure takes the following algorithm:

Algorithm 2.1.1 *IPF Algorithm*

- ***Input****: The observed table,* $\mathbf{x^0} = \left(\mathbf{x^0_{ijk}}\right)_{\mathbf{1 \leq i \leq I,\, 1 \leq j \leq J,\, 1 \leq k \leq K}} \in \mathbb{Z}^{\mathbf{I \times J \times K}}$ *for* $I, J, K \in \mathbb{N}$.
- ***Output****: The estimated MLE,* $\boldsymbol{m} = \left(m_{ijk}\right)_{1 \leq i \leq I,\, 1 \leq j \leq J,\, 1 \leq k \leq K}$*, under the no-three-way interaction model.*
- ***Algorithm****:*
  1. *Initialize* $m^1_{ijk} = 1$ *for* $1 \leq i \leq I,\ 1 \leq j \leq J,\ 1 \leq k \leq K$.
  2. *Compute the marginals:*

$$\begin{array}{rl} x_{ij+} = \sum_{k=1}^{K} x^0_{ijk} & \text{for} \quad 1 \leq i \leq I,\ 1 \leq j \leq J, \\ x_{i+k} = \sum_{j=1}^{J} x^0_{ijk} & \text{for} \quad 1 \leq i \leq I,\ 1 \leq k \leq K, \\ x_{+jk} = \sum_{i=1}^{I} x^0_{ijk} & \text{for} \quad 1 \leq j \leq J,\ 1 \leq k \leq K. \end{array}$$

3. *Until convergence, iterate for $l = 1, 2, \ldots$:*

$$m_{ijk}^{3\cdot l-1} = \frac{m_{ijk}^{3\cdot l-2} x_{ij+}}{\sum_{k=1}^{K} m_{ijk}^{3\cdot l-2}} \quad \textit{for} \quad 1 \le i \le I,\ 1 \le j \le J,$$

$$m_{ijk}^{3\cdot l} = \frac{m_{ijk}^{3\cdot l-1} x_{i+k}}{\sum_{j=1}^{J} m_{ijk}^{3\cdot l-1}} \quad \textit{for} \quad 1 \le i \le I,\ 1 \le k \le K,$$

$$m_{ijk}^{3\cdot l+1} = \frac{m_{ijk}^{3\cdot l} x_{+jk}}{\sum_{i=1}^{I} m_{ijk}^{3\cdot l}} \quad \textit{for} \quad 1 \le j \le J,\ 1 \le k \le K.$$

4. *Return* **m**.

For our research, we simplify the problem by forcing each dimension and its marginal sums to remain consistent and fixed throughout the procedure. Furthermore, we also force the marginal sums to not equate to zero since that may result in a scenario where the MLE does not exist. We enforce these restrictions during the table generation process, shown in Appendix A. Failure to adhere to these stipulations will negate our proposed algorithm.

2.1.9 Markov Chain

Markov chains model a wide array of fields through stochastic systems that transition from one *state*, an event or occurrence in the form of values, to another *state*. Markov chains connects a series of these randomly generated states where each state in the chain only depends on the previous state and not the sequences prior to that immediate predecessor [1]. This principle of "memorylessness" (or "Markov" property) simplifies the *conditional probability*. We also define a "connected" transition graph for a Markov chain as, if for any states u, v in the conditional state space, u can traverse to v and v can traverse to u.

Conditional probability denotes the probability of an event happening given that another event already occurred. A *stochastic process*, X_n for $n = 0, 1, \ldots$, refers to a collection of random variables where X_n represents a *state* (or event) at period n, while n (random variable) accounts for an index (such as time) [9]. In such cases, X_0 denotes the initial state and the subsequent states at period n depend upon the state preceding it.

A contingency table in the conditional state space represents a "state" in our Markov chain for this study. For example, Table 1.1 would represent the initial state, X_0, while Table 1.2

would represent the subsequent state, X_1, since both states belong to the same conditional state space. Figure 2.1 illustrates a simple transition graph.

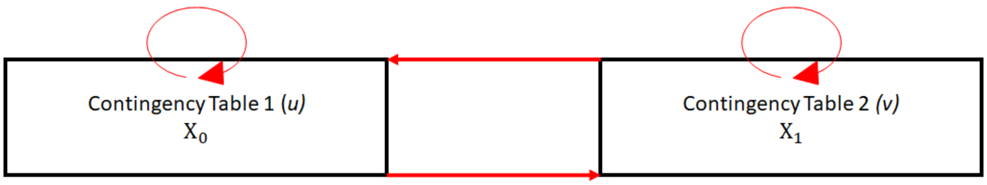


Figure 2.1. Simple Transition Graph. States u, v represent tables in the conditional state space while the red arrows represent a one-step transition, or *basic move*. Red arrows that circle back to themselves represent moves that connects the initial state to states outside of the conditional state space. The arrow circles back to itself because the state eventually connects back to the conditional state space.

2.1.10 Basic Move

Let **b** represent a $I \times J \times K$ table such that

i

	k	k'
j	1	-1
j'	-1	1

i'

	k	k'
j	-1	1
j'	1	-1

,

where $1 \leq i, i' \leq I$, $1 \leq j, j' \leq J$, $1 \leq k, k' \leq K$, $i \neq i'$, $j \neq j'$, and $k \neq k'$, with all other cells at zero. We call **b** a *basic move*, or a *one-step transition* from one connected table to another [3].

In this study, a set of all possible basic moves links each state in the Markov chain for all states within the conditional state space. For example, if we apply the ith level (e.g., two-dimensional version) of the above **b** to Table 1.2, it connects that table directly to Table 1.1. The red arrows in Figure 2.1 represent a single basic move in the Markov chain. In the same figure, the red arrow that circles back to itself indicates that applying the basic move takes the contingency table outside of the conditional state space. For example, if we apply the *i*th level of above **b** to Table 1.1, the resulting table would result in negative cell count values, as seen Table 2.1.

Table 2.1. Another Example, 2x2 Contingency Table with Fixed Variables. Table consists of 2 categorical variables: Cat1, Cat2. "Cat" shortened from Category, with 2 Levels per Cat. Table belongs outside of the conditional state space due to negative cell count values. Note that the marginal sums still equal those of Tables 1.1 and 1.2.

	Cat2		
Cat1	Cat2-Level1	Cat2-Level2	Total by Cat1
Cat1-Level1	2	−1	1
Cat1-Level2	−1	2	1
Total by Cat2	1	1	2

Since it would not make sense for a categorical variable to take a negative value (i.e., a negative *yes* or *no* count for death penalty in Figure 1.1), the table would fall outside of the conditional state space.

We must also distinguish basic moves from *moves*. *Moves* also represent a one-step transition; however, a *move* includes more than four elements in a single layer. For example, the following table represents a *move*, as opposed to a basic move, because it includes more than four elements in a single layer:

$$\begin{array}{|c|c|c|c|} \hline +1 & -1 & 0 & 0 \\ \hline 0 & +1 & -1 & 0 \\ \hline -1 & 0 & +1 & 0 \\ \hline \end{array} .$$

The marginal sums for a *move* must still equate to zero, as shown above.

2.1.11 Discrete Time Markov Chain

Discrete Time Markov Chain (DTMC) *transitions* from a state i to a state j over discrete periods n, although it may transition to the same state for multiple periods in a row. In a DTMC, p_{ij} represents the one-period transition probability that a system moves to state

j during the next period, given current state i. The transition probability must satisfy the following properties:

- $0 \leq p_{ij} \leq 1, \forall i, j$. The transition probability cannot take on negative values or values greater than 1.
- $\sum_j p_{ij} = 1, \forall i$. This condition states that the system must transition to some state next period.

In this research, each state of the DTMC represents each connected contingency table (through basic moves) that satisfies the sufficient statistics. The transition graph reflects the connection of states within the Markov chain. Figure 2.1 represents a simple DTMC transition graph.

2.1.12 Markov Basis

In a previous study, Diaconis and Sturmfels defined the notion of a Markov basis for a log-linear model [3]. Suppose $x^0 = (x^0)_{ijk_{1\leq i\leq I, 1\leq j\leq J, 1\leq k\leq K}}$ represents an observed three-way table. Then, F_{x^0} represents the conditional state space with all tables that share the same marginal sums as candidate table, x_0:

$$F_{x^0} = \left\{ X_{ijk} \in \mathbb{R}^{I\times J\times K} : X_{ijk} \geq 0,\ X_{\cdot jk} = (x^0)_{\cdot jk},\ X_{i\cdot k} = (x^0)_{i\cdot k},\ X_{ij\cdot} = (x^0)_{ij\cdot} \right\}.$$

We define Markov basis, M, as a collection of all moves such that for fixed positive integers I, J, K, for any tables $X, X' \in F_{x^0}$, we find a sequence

$$X' = \sum_{s=1}^{S} (X + M_s),$$

with

$$\sum_{s=1}^{s'} (X + M_s) \in F_{x^0},$$

for all $1 \leq s' \leq S$, any $M_i \in M$, $S \geq 1$, and for any possible observed table, x^0 [5]. Stated simply, a Markov basis represents a set of moves that allows for all tables in the conditional state space to connect via a Markov chain while remaining within that conditional state space. For a more comprehensive explanation of the Markov basis, please refer directly to the cited sources for reference.

2.1.13 Irreducibility and Aperiodicity

Ideally, Markov chains possess both fundamental characteristics of irreducibility and aperiodicity. Irreducibility means that any state in the state space consists of a positive probability of visiting all other states in the chain [10]. Aperiodicity means that the transition of the chain should not include an inescapable cycle in their transitions [10]. In our Markov chain of contingency tables, a state remains in the same state *only if* the cell counts exceed acceptable parameters (i.e., cell count takes -2 as a value) once we apply the basic move. We look to prove the irreducibility of our Markov chain, in the extended conditional state space that includes -1 cell count values for $3 \times 3 \times K$ tables, in Chapter 4.

2.1.14 Hypergeometric Distribution

The hypergeometric distribution serves as our conditional distribution because the process involves sampling *without replacement* [6]. We categorize the hypergeometric distribution as a conditional distribution since each outcome requires the development of a new distribution. For example, suppose we have N items, of which k number of defective items exist. If we remove one item at a time sequentially, the outcome of the previous draw will influence the next draw so long as that first item remains out of the original group of items. Thus, the hypergeometric distribution involves *dependent* trials [6].

We illustrate the example in a 2×2 contingency table format, where N represents the total number of items, k represents the number of defective items, x represents the number of defective items drawn, and n represents the number of non-defective items drawn:

	Drawn	Not Drawn	Total
Defective	x	$k-x$	k
Non-defective	$n-x$	$N+x-n-k$	$N-k$
Total	n	$N-n$	N

.

If x represents the number of defective items drawn, the hypergeometric distribution takes the probability mass function which simplifies to [6]:

$$P(X = x) = \frac{\binom{k}{x}\binom{N-k}{n-x}}{\binom{N}{n}}.$$

2.1.15 Monte Carlo Simulation

Monte Carlo simulations model stochastic systems and calculate probabilities for a variety of outcomes [11]. Monte Carlo simulations perform random sampling within a large number of experiments in order to provide empirical data to evaluate statistical characteristics [12]. This technique uses inputs to conduct upwards of tens of thousands of simulations in order to evaluate complex models. Mathematicians employ this method in order to find approximate solutions to numerical problems difficult to solve by other means [9].

2.1.16 Metropolis-Hastings Algorithm

The Metropolis-Hastings algorithm samples a sequence of random observations from a *proposed* probability distribution [10]. The Metropolis-Hastings algorithm generally supports sampling from high-dimensional distributions. Furthermore, this algorithm forces the Markov chain toward points with higher probabilities in the conditional state space. That process allows us to sample points according to the proposed probability distribution. The output determines whether the initial table or the proposed table becomes the next state.

Algorithm 2.1.2 *Metropolis Algorithm on the Set of Tables*

- ***Input**: The proposed state, $\mathbf{X}^*$, and the initial state, $\mathbf{X}$, along with a general log-linear model, F.*
- ***Output**: Next state, $\mathbf{X}'$*
- ***Algorithm**:*
  1. *Compute the ratio of probability, which represents the ratio of the probability of the proposal over the probability of the current state. We use this ratio as a metric to accept or reject the proposal. We calculate the ratio through the equation,*

$$r = \frac{p(X^*|m)}{p(X|m)},$$

where, m, represents the sufficient statistics under, F. Since marginal sums serve as the sufficient statistic for no-three-way interaction models, the equation translates to

$$r = \frac{\prod_{\text{all cell counts } j \text{ in } X} j!}{\prod_{\text{all cell counts } k \text{ in } X^*} k!},$$

where $\prod_{\text{all cell counts } j \text{ in } X} j!$ *represents the product of the factorial of all cell counts, j, in X and* $\prod_{\text{all cell counts } k \text{ in } X^*} k!$ *represents the product of the factorial of all cell counts, k, in* X^**, as shown in Example 2.1.3*

2. *Set:*

$$\mathbf{X}' = \begin{cases} \mathbf{X}^* & \textit{With probability } \min(r, 1) \textit{ and if } \mathbf{X}^* \geq \mathbf{0} \\ \mathbf{X} & \textit{Otherwise.} \end{cases}$$

3. *Return* $\mathbf{X}'$.

Example 2.1.3 *Suppose we have a table,*

$$X = \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ \hline \end{array}.$$

Then we calculate

$$\prod_{\text{all cell counts } j \text{ in } X} j! = 1!2!3!4!5!6!.$$

This Metropolis-Hasting algorithm ensures that MCMC samples from the hypergeometric distribution.

2.1.17 Markov Chain Monte Carlo

A commonly used, computer-driven statistical sampling method, MCMC combines the properties of both Markov chain and Monte Carlo simulation. This method involves traversing through the state space by adhering to the fixed marginal sums of the given contingency table [13]. MCMC simulations allow for the characterizing of a distribution despite limited knowledge of the distribution's mathematical properties [12]. Simply put, the MCMC simulation estimates the expectation of statistics for hard combinatorial problems [10]. In statistics, MCMC remains one of the most popular methods to sample from the conditional distribution of multi-way contingency tables. Data sets continue to grow in size and magnitude; therefore, so does the importance of an efficient sampling approach.

The following algorithm simulates a DTMC on the conditional state space:

Algorithm 2.1.4 *MCMC Basic Move Algorithm on Contingency Tables*

- ***Input****: The observed* $I \times J \times K$ *table,* X_0*, with* $I, J, K \in \mathbb{N}$*, and sample size,* N*.*
- ***Output****: Sample tables in accordance with the conditional distribution.*
- ***Algorithm****:*
  1. *Initialize the set of sampled tables,* $\mathbf{S} = \emptyset$*.*
  2. *For* $i = 1, \cdots, N$*, do the following:*
     - 2.1. *Pick distinct pairs of dimensional indices,* $i, i' \in \{1, 2, \ldots, I\}$*,* $j, j' \in \{1, 2, \ldots, J\}$ *and* $k, k' \in \{1, 2, \ldots, K\}$*.*
     - 2.2. *Let basic move,* b*, represent a* $I \times J \times K$ *table such that*

		k	k'
i	j	1	−1
	j'	−1	1

		k	k'
i'	j	−1	1
	j'	1	−1

     with all other cells at zero.
     - 2.3. *Sample a proposal table,* $X_i = b + X_{i-1}$*.*
     - 2.4. *Accept the proposal according to the Metropolis Hasting Algorithm, described in Algorithm 2.1.2.*
     - 2.5. *Verify that the marginal sums of the proposal table matches the marginal sums of the initial table and that no cell count has a negative value. If verification fails, return to the initial state.*
     - 2.6. *Upon verification, add* X_i *to* S*.*
  3. *Return* $\mathbf{S}$*.*

This algorithm describes the basic MCMC sampling process employed in this study. The simple example outlined under the definition for basic moves illustrates this process. However, this approach does not guarantee a connected Markov chain; in other words, this process does not guarantee aperiodicity and irreducibility. For a more comprehensive definition of this sampling method, please refer directly to the cited references from this subsection.

2.1.18 Asymptotic Distribution

When calculating exact statistical results proves difficult, approximate results based on known properties or behaviors of certain statistics in large samples offer the best alternative. An asymptotic distribution, or *limiting distribution*, represents the hypothetical distribution, or *convergence*, of a sequence of distributions [2]. The *asymptotic distribution theory* attempts to find this convergence through a series of distributions [2].

2.1.19 Chi-Squared (χ^2) and Goodness-of-Fit Tests

In CDA, the χ^2 test examines two statistical phenomena. The χ^2 test for independence involves checking for the *independence* of variables while the goodness-of-fit test involves calculating the *likelihood* that the observed distribution happened by chance. Typical uses of the goodness-of-fit tests involve evaluating how well a proposed probability model, or *theoretical model*, fits to an observed data set.

A commonly used form of hypothesis testing, goodness-of-fit tests look to see how well a proposed model fits to a given data set by comparing the observed table with the MLE [1]. In the case of three-way tables, the test measures how well the observed distribution of data (i.e., reality) fits to a theoretical model that distributes the data under the assumption that all three variables have an *interaction*. We reject the null hypothesis if the observed data does not fit the model because the likelihood of an interaction between variables increases. Figure 2.2 shows the work flow diagram of a goodness-of-fit test used in this research. Please note that the arrows denote required tasks prior to the start of the subsequent step.

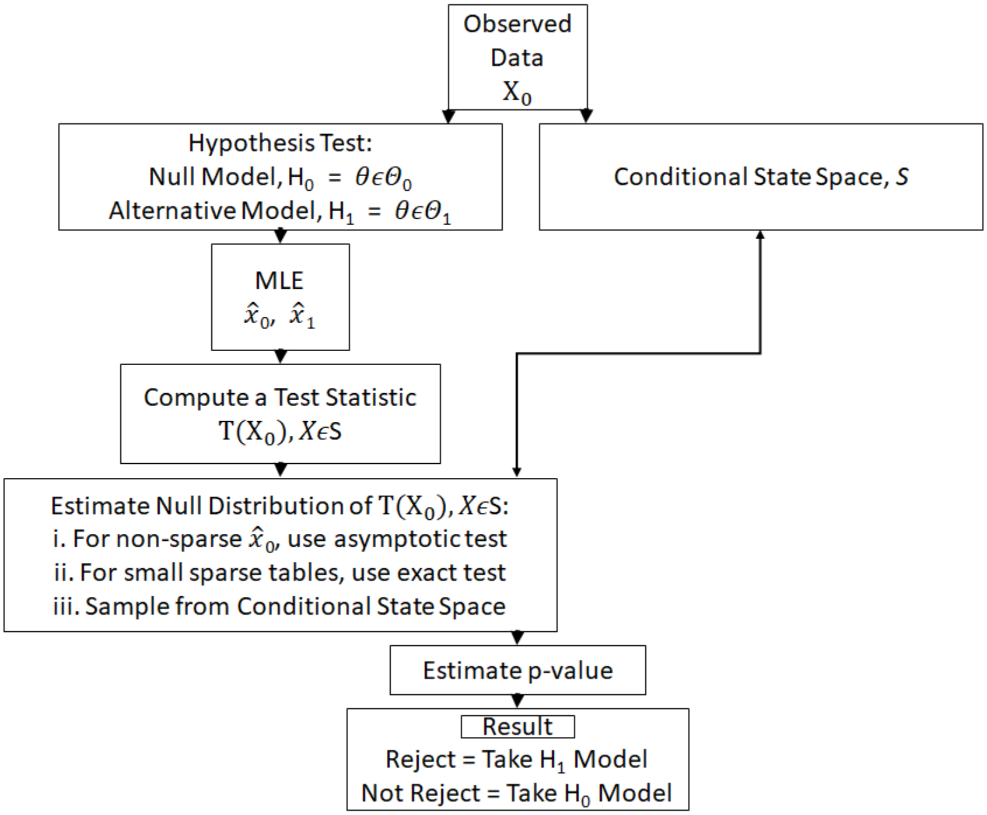

Observed Data X_0
Hypothesis Test: Null Model, $H_0 = \theta\epsilon\Theta_0$ Alternative Model, $H_1 = \theta\epsilon\Theta_1$
Conditional State Space, S
MLE $\hat{x}_0, \hat{x}_1$
Compute a Test Statistic $T(X_0), X\epsilon S$
Estimate Null Distribution of $T(X_0), X\epsilon S$:
i. For non-sparse $\hat{x}_0$, use asymptotic test
ii. For small sparse tables, use exact test
iii. Sample from Conditional State Space
Estimate p-value
Result
Reject = Take H_1 Model
Not Reject = Take H_0 Model

Figure 2.2. Step-by-Step Goodness-of-Fit Test Workflow

The null hypothesis of a χ^2 test states the independence of variables. Mathematically, independence means that the product of the normalized marginal sums equals the probability of any cell. For this research, we use the test to analyze the relationship between three variables and consider the following hypotheses:

$$\begin{aligned} H_0 &: \quad \lambda_{ijk}^{TYZ} = 0 \\ H_1 &: \quad \lambda_{ijk}^{TYZ} \neq 0. \end{aligned}$$

Like all hypothesis testing, we begin by assuming the null hypothesis is true. In this case,

$$H_0 : \lambda_{ijk}^{TYZ} = 0.$$

We conduct hypothesis testing using the following procedure. First, we calculate the *test* *statistics* of the observed table x_0. The test statistics serves as a metric to measure a distance from a given table to the *MLE* under the proposed model to the observed table. In this case, a test statistic measures how far a given table varies from the MLE under the no-three-way interaction model. In our research, we use *Pearson's χ^2 test statistics*. The χ^2 test statistic measures the deviation from the observed values and the calculated expected values in order to determine the conclusion of the test. The calculation itself involves summing the squared differences between the observed data set and the MLE,

$$\chi^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \frac{(x_{ijk}^0 - m_{ijk})^2}{m_{ijk}},$$

where m_{ijk} represents the MLE. Under the hypothesis test, the null distribution mirrors the test statistics between a table $\mathbf{X}$ generated under the *conditional distribution* given the sufficient statistics and the MLE under the no-three-way interaction model. In this situation, hypergeometric distribution represents the conditional distribution, given the sufficient statistics computed from the observed table x_0.

χ^2 tests apply the asymptotic distribution theory. We must note that we can only use the asymptotic distribution of the test statistics as the null distribution for goodness-of-fit tests if all of the expected cell counts exceed five [1]. Asymptotic distribution theory shows that the asymptotic distribution of Pearson's χ^2 test statistics for tables, $\mathbf{X}$, generated under the conditional distribution is the χ^2 distribution with degree of freedom $(I-1)(J-1)(K-1)$. However, we cannot use the asymptotic distribution with sparse tables. Instead, we want to conduct the *exact conditional test*.

2.1.20 Fisher's Exact Test

Fisher's exact test, one of the most famous exact conditional tests, exists as an alternative for smaller tables, such as $2 \times 2 \times 2$ or $3 \times 3 \times 3$, that do not satisfy the condition that each of the expected cell counts exceed five [1]. Fisher's exact test involves enumerating all possible tables in the conditional state space with the fixed sufficient test statistics (i.e., marginal sums) from the observed table, X_0. We then compute the null distribution by calculating all test statistics (such as the χ^2 test statistic or the log-likelihood ratio test statistic) for each table in the conditional state space. However, enumerating every table in the conditional state space may not be feasible for large contingency tables. Therefore, we *approximate* the null distribution by sampling tables from the conditional distribution.

2.1.21 P-Value

During hypothesis tests, we calculate the p-value in order to determine the results of our tests. The value will dictate whether we have sufficient evidence to reject the null hypothesis, H_0, for the alternate hypothesis, H_1. We employ the following exact conditional test algorithm in order to calculate the p-value:

Algorithm 2.1.5 *Exact Conditional Test*

- ***Input***: *The observed table,* $\mathbf{x^0} \in \mathbb{Z}^{\mathbf{I\times J\times K}}$ *for* $I, J, K \in \mathbb{N}$. *Sample size, n.*
- ***Output***: *The estimated p-value.*
- ***Algorithm***:
    1. *Compute the estimated MLE via IPF, as shown in Algorithm 2.1.1.*
    2. *Compute the sufficient statistics from* $\mathbf{x^0}$ *for the MLE under the null model.*
    3. *Compute the test statistic,* $\chi^2(\mathbf{x^0})$.
    4. *Sample tables,* $\mathbf{x^1}, \ldots, \mathbf{x^n}$, *from the conditional state space using the sufficient statistics.*
    5. *Estimate p-value for the hypotheses by computing*

       $$\frac{\sum_{i=1}^{n} \mathbb{I}_{\chi^2(\mathbf{x^i}) \geq \chi^2(\mathbf{x^0})}}{n},$$

       where $\mathbb{I}$ *represents the indicator function for all of the test statistic values greater than the* χ^2 *value of the MLE.*

The significance level, α, determines the threshold of whether or not to reject the null hypothesis. For example, if we set α at .05, we can only reject the null hypothesis in favor of the alternative hypothesis if the p-value calculates below .05. If the p-value does drop below .05, it indicates evidence against the null hypothesis. In our research, we set the significance level, α, at .05.

2.1.22 Sparse Tables

In CDA, sparse tables describe tables populated with many cells with small counts (the expected cell count fails to exceed five) and/or zeroes. For example, Tables 1.1, 1.2, and 2.1 represent sparse tables. As we consider higher-dimensional contingency tables, the likelihood of encountering sparseness will increase. Since log-linear models correspond to marginal sums, biased or inaccurate MLE may result when sampling from sparse contingency tables [14]. In this case, χ^2 distributions may do a poor job of approximating the sampling distribution of test statistics [14].

As stated earlier, traditional asymptotic methods require that each cell count exceed five in all or most cells of the contingency table. If this condition does not hold, we label the table as sparse since the χ^2 approximations of goodness-of-fit statistics may inaccurately evaluate the fit of the proposed model to the data set [1]. Thus, identifying a proper method in analyzing these types of tables serves as one of the primary motivations for this research.

2.2 Motivation

In this research, we examine whether or not our proposed MCMC sampling method can efficiently and properly sample three-way contingency tables, under the no-three-way interaction model, from a hypergeometric distribution. The analysis of contingency tables applies to a multitude of fields involving data science. We look to fill a need for accurate model fitting and sampling in these fields due to an increasing demand by both industry and government.

Traditional methods may yield inaccurate or biased results for large sparse contingency tables. The likelihood of encountering sparseness in contingency tables increases as the dimensions of the table increase. Therefore, a growing need exists to find an efficient and appropriate statistical method to conduct analysis given the prevalence of and sparseness in “big data.” In the field of CDA, data sets will almost certainly have some cell counts of zero. However, many researchers still opt to use asymptotic distribution analysis for sparse contingency tables despite concerns of inaccurate results because of a lack of alternative procedures.

We study MCMC primarily because another popular sampling method, Sequential Importance Sampling (SIS), cannot draw samples from the conditional distribution–namely, the hypergeometric distribution. In order to run MCMC properly, the transition graph of the chain must connect–unlike SIS that samples independently from the conditional state space [15]. If the transition graph consists of unconnected chains, we cannot sample tables from those parts of the graph. This leads to a biased conclusion. Therefore, this thesis shows that our

Many researchers have previously studied MCMC approaches developed by Diaconis and Sturmfels using Markov bases [3]. However, we cannot compute Markov bases for tables larger than $4 \times 4 \times 4$ under this model because of computational limitations [16]. Computing a Markov basis for the no-three-way interaction model proves difficult because, generally, the number of elements (moves) in a Markov basis can become arbitrarily large (i.e., no upper bound exists) [17]. Additionally, SIS may draw rejected samples, or samples outside of the conditional state space, and require computationally inefficient integer programming.

This thesis focuses on three-way contingency tables under the no-three-way interaction model, a special case of log-linear models. Since no methods currently exist to compute a Markov basis for tables larger than $4 \times 4 \times 4$, we apply the idea proposed by Bunea and Besag (1996) and Chen et al. (2005) in order to connect a Markov chain in the conditional state space. Their proposed method involves allowing -1 as a cell count in order to connect Markov chains in the conditional state space [4], [5]. We elaborate more on their work in the next section.

Combined with the idea of Bunea and Besag, we will prove that a MCMC with a set of basic moves on $3 \times 3 \times K$, for $K \geq 3$, connects within the conditional state space. We will also provide evidence through simulations that our method works for $I \times J \times K$ tables, for $I,\ J,\ K \geq 3$, under the no-three-way interaction model. This work produces a new statistical method for sampling and analyzing categorical data. Furthermore, this research may aid efforts in identifying appropriate statistical analysis procedures for big data. We anticipate this algorithm will aid in efforts to analyze data pertaining to data and information sciences.

2.3 Literature Review

Several works laid the foundation for this research. Although this research draws from many established statistical practices and influences (e.g., Categorical Data Analysis by Agresti), this research combines the findings from three primary works in related fields for the proposed modification to MCMC sampling. Therefore, this section focuses on those works to set the proper context of this research. For a more comprehensive understanding of these studies, please refer directly to the cited sources for reference.

2.3.1 Diaconis and Sturmfels

The findings of Diaconis and Sturmfels contributed the most to this research given that they developed MCMC sampling for the discrete exponential family with Markov basis. Applying the MCMC method involves approximating the null distribution of the test statistic for goodness-of-fit test by sampling contingency tables from the conditional distribution. In 1998, Diaconis and Sturmfels defined the notion of a Markov basis. For example, if we consider the independence model on two random variables, the Markov chain on all contingency tables in the conditional state space connects through a set of basic moves. They called this guarantee—to connect all states of a Markov chain in the conditional state space—a Markov basis. With any given sufficient statistics under the model, they initiated a MCMC approach based on a Gröbner basis computation for testing statistical fitting [3]. Their research also involved constructing Markov chain algorithms for sampling contingency tables from discrete exponential families [3]. Since its inception, studies pertaining to the structure of Markov bases remain a popular source for academic interest in computational algebraic statistics. Despite computational advances, one may fail to calculate a Markov basis for some statistical models because the minimal Markov basis for a model may include an exponential number of moves.

2.3.2 Bunea and Besag

Bunea and Besag's work initiated the idea of allowing -1 cell count values for contingency tables while sampling via MCMC. However, their particular study only focused on contingency tables with dimensions $2 \times J \times K$, where $I, J \in \mathbb{N}$, under the no-three-way interaction model. Their paper reviews MCMC exact tests for assessing the goodness-of-fit of probability models to observed data sets [4].

In our study, we apply this concept in order to *expand* the conditional state space to include tables with cell count values of -1. For example, applying this principle would allow us to include Table 2.1 in the expanded conditional state space because it shares the sufficient statistics as Tables 1.1 and 1.2. Although we would normally not include tables with negative values, we make an exception for those tables with -1 values that still share the same sufficient statistic as the other tables in the conditional state space.

2.3.3 Chen et al.

The work by Chen et al. introduces multiple concepts studied by this current research. Chen et al. presented algebraic methods for studying the connectivity of Markov moves with the assumption of all positive margins [5]. They also developed Markov sampling methods for exact conditional inference of statistical models, where computing a Markov basis may pose challenges [5]. Their study focused on positive marginal sums greater than zero and found that sets of Markov moves that connect tables with positive margins may work better than calculating a full Markov basis [5]. Chen et al. also investigated the condition of allowing -1 cell count values from a theoretical perspective. They found some necessary conditions to identify connecting Markov chains when allowing negative table entries [5].

CHAPTER 3:
Methodology and Data

This chapter introduces the methodology, simulation procedures, and simulation data sets. The methodology section will also include the proposed modification to the already-popular MCMC sampling method developed by Diaconis and Sturmfels. The simulation section introduces hypothesis testing as it relates to our simulation.

3.1 Methodology

This section discusses the proposed modification to a sampling method developed by Diaconis and Sturmfels. Already popular in exact conditional testing, Diaconis and Sturmfels proved that contingency tables within the same conditional state space can connect through a set of MCMC basic moves called a Markov basis [3]. The foundational principle of this procedure involves approximating the null distribution by sampling from the conditional distribution [3]. Specifically, the approach involves approximating the null distribution of the test statistic, calculated from each table sampled from the hypergeometric distribution.

We must note that we cannot always compute a Markov basis for every problem because computing the minimal Markov basis for larger models may require an infeasible amount of computational time. Therefore, we propose a MCMC sampler without computing a Markov basis by allowing any cell count to possess -1 as a value in order to connect Markov chains in the conditional state space, effectively expanding the conditional state space to temporarily include those tables. However, we do not accept a sampled table with -1 as a cell count value; we only allow a table to have -1 as a means of connecting to another table in the conditional state space.

The following algorithm describes our proposed method. The only difference between this algorithm and the MCMC Algorithm 2.1.4 involves allowing cell counts to take -1 (Step 2.2.5).

Algorithm 3.1.1 *Proposed MCMC Basic Move Algorithm on Three-way Contingency Tables, under the No-Three-Way Interaction Model*

- ***Input**: The observed $I \times J \times K$ table, X_0, with $I, J, K \in \mathbb{N}$, and sample size, N. No-three-way Interaction Model, F.*
- ***Output**: Sampled tables in accordance with the hypergeometric distribution.*
- ***Algorithm**:*
  1. *Initialize the set of sampled tables, $\mathbf{S} = \emptyset$.*
  2. *For $i = 1, \cdots, N$, do the following:*
     - 2.1. *Pick distinct pairs of dimensional indices, i, $i' \in \{1, 2, \ldots, I\}$, j, $j' \in \{1, 2, \ldots, J\}$ and k, $k' \in \{1, 2, \ldots, K\}$.*
     - 2.2. *Let basic move, b, represent a $I \times J \times K$ table such that*

i

	k	k'
j	1	-1
j'	-1	1

i'

	k	k'
j	-1	1
j'	1	-1

with all other cells at zero.

  - 2.3. *Sample a proposal table, $X_i = b + X_{i-1}$.*
  - 2.4. *Accept proposal according to the Metropolis-Hasting Algorithm, as defined Algorithm 2.1.2.*
  - 2.5. *Verify that the marginal sums of the proposal table match the marginal sums of the initial table and that no cell count has a value less than -1. If verification fails, return to the initial state.*
  - 2.6. *Upon verification, add X_i to S.*

  3. *Return $\mathbf{S}$.*

We illustrate our modification through the following example. Suppose we have a $i \times j \times k$

contingency table, where $i = 2$, $j = 3$, and $k = 3$,

$i = 1$

1	0	0
0	1	0
0	0	1

$i = 2$

0	1	0
0	0	1
1	0	0

,

where the two-way marginal sums equate to one. From this table, we want to get to the following table in the same conditional state space

$i = 1$

0	1	0
0	0	1
1	0	0

$i = 2$

1	0	0
0	1	0
0	0	1

,

which by definition shares the same marginal sums. In order to connect these two tables via basic moves, we must use a table in the *extended conditional state space*. Specifically, we must allow any cell count from this table to possess a value of -1. Thus, if we apply the following basic move to the original table,

$i = 1$

-1	$+1$	0
$+1$	-1	0
0	0	0

$i = 2$

$+1$	-1	0
-1	$+1$	0
0	0	0

,

we get the following table from the extended conditional state space, which shares the same marginal sums as the previous table but with -1 as a cell counts:

$$
\begin{array}{cc}
i = 1 & \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & 0 & 0 \\ \hline 0 & 0 & 1 \\ \hline \end{array} \\
i = 2 & \begin{array}{|c|c|c|}\hline 1 & 0 & 0 \\ \hline -1 & 1 & 1 \\ \hline -1 & 0 & 0 \\ \hline \end{array} .
\end{array}
$$

From here, if we apply one more basic move,

$$
\begin{array}{cc}
i = 1 & \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline -1 & 0 & +1 \\ \hline +1 & 0 & -1 \\ \hline \end{array} \\
i = 2 & \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline +1 & 0 & -1 \\ \hline -1 & 0 & +1 \\ \hline \end{array} ,
\end{array}
$$

we end up with the next state in the Markov chain within the traditional conditional state space:

$$
\begin{array}{cc}
i = 1 & \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \\ \hline 1 & 0 & 0 \\ \hline \end{array} \\
i = 2 & \begin{array}{|c|c|c|}\hline 1 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \\ \hline \end{array} .
\end{array}
$$

Although the combination of these two basic moves connects these tables from the same conditional state space, we cannot call this set of moves a Markov basis because the process included a table outside of the traditional conditional state space.

So to summarize the proposed modifications from this research to Diaconis and Sturmfels' MCMC:

- We prove connectivity for $3 \times 3 \times K$ contingency tables, for $K \geq 3$, through MCMC with a set of basic moves, but without computing the Markov basis, by allowing any cell counts to take values of -1. As defined earlier, a Markov basis represents a set of one-step transition moves, which allow for all tables in the conditional state space to connect via a Markov chain.
- We use marginal sums as the sufficient statistics, as defined by the no-three-way interaction model. By simplifying the likelihood function when computing the MLE, we see that the marginal sums provide sufficient information to infer the MLE.
- We calculate the χ^2 test statistic from each sampled table in order to conduct goodness-of-fit testing, where the null hypothesis states a no-three-way interaction.
- We conduct a Kolmogorov–Smirnov Test (K–S Test) to show that the null distribution of the test statistics for sparse contingency tables does not come from the χ^2 distribution. For non-sparse tables, the K–S Test shows that the null distribution of the test statistics matches the χ^2 distribution.
- We determine that we cannot use asymptotic distributions (i.e., χ^2 distribution) for sparse contingency tables. Therefore, we sample contingency tables from the expanded conditional state space in order to estimate the null distribution of the test statistics.

We apply this sampling method to three-way contingency tables under the no-three-way interaction model. For this type of model, no methods currently exist in computing a Markov basis for tables larger than $4 \times 4 \times 4$. To summarize, we apply the ideas of Bunea and Besag (2000) and Chen et al. (2005) in order to connect a Markov chain the in the extended conditional state space. These ideas involve temporarily allowing -1 as a valid cell count value for contingency tables in order to connect tables via a Markov chain.

3.2 Simulation

We provide strong evidence through simulations that our method works for any $I \times J \times K$ tables, for I, J, $K \geq 3$, under the no-three-way interaction model. We break up the simulations into two parts: non-sparse and sparse contingency tables. In order to apply our proposed algorithm on sparse data, we must first prove that it functions properly on non-sparse data. We must ensure that our proposed methodology aligns with already-established procedures for conventional non-sparse data since mathematicians proved and validated such methods.

3.2.1 Procedures

Both parts of the simulations adhere to the following procedures. Elaboration on certain steps not defined in detail in Chapter 2.1 will follow in subsequent sections. This report includes the R codes used for the simulations in Appendix A.

1. Sample a three-dimensional table, where sampling for each cell comes from a Poisson distribution. Each table must not have any marginal sums that equal zero for the algorithm to function properly.
2. Estimate the MLE for the sampled table through the IPF approach as described in Algorithm 2.1.1.
3. Calculate the χ^2 statistic, using the observed table and MLE as arguments.
4. Sample tables from the conditional state space via MCMC as described in Algorithm 3.1.1, using the marginal sums as the sufficient statistic. During the MCMC sampling process, tables can have cell count values of -1 but it will not count as a sampled table. Tables with -1 will exist in the expanded conditional state space in order to allow tables within the traditional conditional state space to connect.
5. Calculate the χ^2 statistics using each sampled table and the MLE as the argument.
6. Conduct a goodness-of-fit test on the null hypothesis that states a no-three-way interaction. This test concurrently involves model selection between the null and alternative models. The test itself depends on the calculation of the p-value from the test statistics collected throughout the simulation. We calculate the p-value in accordance with Algorithm 2.1.5.
7. Conduct the K–S Test to check whether the null distribution of the test statistics closely resembles an asymptotic distribution.
8. Record the p-value of both goodness-of-fit and K–S Tests for the simulation.

3.2.2 Parameters

Simulations will run for both sparse and non-sparse contingency tables with dimensions of $3 \times 3 \times K$, for $K \geq 3$, and a $4 \times 4 \times 4$ and $5 \times 5 \times 5$ table. Although we sample the cell counts of the observed table from the Poisson distribution, we sample each table from the hypergeometric distribution. This applies to both sparse and non-sparse simulations.

3.2.3 MCMC and Metropolis-Hastings Algorithm

The MCMC process, as described in Algorithm 2.1.4, plays the most crucial role in the sampling process. During the simulation, the MCMC function takes the observed contingency table as the input argument along with the desired sample size, N. The algorithm continues until the set of sample tables from the conditional state space equate to N. If, and only if, the sampled table's marginal sums equal the marginal sums of the initial table once we apply the basic move, and the table does not have a -1 as a cell count value, we add that sample table to the set.

The Metropolis-Hastings algorithm, Algorithm 2.1.2, ensures that the MCMC samples from the hypergeometric distribution. In our employment of this algorithm, we do not calculate a Markov basis for the MCMC due to the general infeasibility of computing them for the no-three-way interaction model. We instead *expand* the conditional state space to include tables with negative values. We skip calculating a Markov basis because, stated simply, a Markov basis represents a set of moves that allows for all tables in the conditional state space to connect via a Markov chain without leaving the conditional state space. A set of moves only becomes a Markov basis if, and only if, applying the set of moves to any state x in the conditional state space connects that table with any other state x' in the conditional state space via a single chain while staying within the conditional state space. Therefore, by leaving the conditional state space, we do not calculate a Markov basis.

We introduced the ratio of probability in Algorithm 2.1.2 as the following general equation

$$r = \frac{\prod_{\text{all cell counts j in } X} j!}{\prod_{\text{all cell counts k in } X^*} k!},$$

where X represents the current state and X^* represents the proposed state in the conditional state space. In the simulation, we take the log of this function in order to avoid potential numerical errors from computing large numbers. In the simulation Metropolis-Hastings function (re: compute.ratio function in R code, Appendix A), we calculate the ratio, r, through the following equation [3],

$$r = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K} \log\left(\mathbf{x_{ijk}^{n+1}}!\right) - \log\left(\mathbf{x_{ijk}^{n}}!\right),$$

where $\mathbf{x}^{\mathbf{n}}$ represents the current table, or state, in the MCMC and $\mathbf{x}^{\mathbf{n+1}}$ represents the proposed state. We then establish a random parameter, u, from the uniform distribution, $\mathbb{U} \in [0, 1]$. If $\min(1, e^r) \leq u$, we accept the move and continue the iteration with the accepted move as the initial state; otherwise, we do not accept the move and the table remains in the same state.

3.2.4 Burn-In

Burn-in describes the process of throwing away a set number of initial samples at the beginning of each trial. We conduct this process in order to minimize the influence of the initial table. We want to minimize any potential for bias at the start of each trial.

3.2.5 Goodness-of-Fit Test

This simulation incorporates the goodness-of-fit test, defined in Chapter 2.1, for model selection. Using the approximated MLE from the step prior, we calculate the χ^2 test statistic for each table. Suppose we have the null hypothesis, H_0, versus the alternative hypothesis, H_1. We apply MCMC for the goodness-of-fit test via the following algorithm.

Algorithm 3.2.1 *MCMC for Goodness-of-Fit*

- ***Input**: An observed three-way table,* $\mathbf{x}^{\mathbf{0}}$*, and the sample size, N. Number of burn-in, B. Model for* H_0*:* F_0*.*
- ***Output**: A list of test statistics computed from sampled tables from the hypergeometric distribution.*
- ***Algorithm**:*
  1. *With initial table* $\mathbf{x}^{\mathbf{0}}$*, sample B many tables (where* $\mathbf{x}^{\mathbf{0}*}$ *represent the last, or Bth, sampled table) under* F_0*, using Algorithm 2.1.4.*
  2. *Initiate* $L = \emptyset$*.*
  3. *Compute the MLE,* $\mu^{\mathbf{0}}$*, under* H_0*, via IPF (Algorithm 2.1.1).*
  4. *For* $i = 1, \cdots, N$*, do the following:*
     - *4.1. Sample a table,* $\mathbf{x}^{\mathbf{i}*}$*, under* F_0*, with* $\mathbf{x}^{\mathbf{(i-1)}*}$ *using Algorithm 2.1.4.*

```
        4.2. Compute the Pearson's $\chi^2$ test statistic:
                $$s = \sum_{j,k} \frac{(x^{i*}_{jkl} - \mu^0_{jkl})^2}{\mu^0_{jkl}}.$$
        4.3. Add s to L.
    5. Return L.
```

Ultimately, we conduct the goodness-of-fit test in order to determine whether we select a no-three-way interaction model or a more complicated model. The distribution of this test statistic forms the null distribution, which we use to observe the characteristics of the tables in the conditional state space. We also take note of what value would deem a test statistic as rare if we assume that the null model best fits the data.

As defined earlier, goodness-of-fit tests help determine whether the null model or alternative model fits better with an observed data set. If the observed data set does not significantly differ from the expectation under the null model, then we select the null model. If significantly different, then we will select an alternative model. In order to measure the difference in models, we utilize the χ^2 test statistic to measure the difference between two tables. The alternative model suggests an interaction (e.g., a correlation) between variables.

3.2.6 K–S Test

The K–S Test serves as a means of testing the equality of distributions:

$$H_0 : \text{Two distributions are the same.}$$
$$H_1 : \text{Two distributions are not the same.}$$

For this research, we employ the K–S Test to check whether the null distribution of the test statistic resembles an asymptotic distribution:

$$H_0 : \text{Null distribution of test statistics resembles the } \chi^2 \text{ distribution.}$$
$$H_1 : \text{Null distribution of test statistics does not resemble the } \chi^2 \text{ distribution.}$$

3.3 Data

For both non-sparse and sparse simulations, we generated random tables from a Poisson distribution with a parameter λ. As previously defined, this means that we sampled each cell count from a Poisson distribution. The random table generator function also ensures that no marginal sums equate to zero. Once we generated a complete table, we used this table as an observed table given the fixed sufficient statistic. Sample tables from the conditional state space come from a hypergeometric distribution via MCMC. We included the code that specifies the method in which we generate these random tables in Appendix A.

3.3.1 Non-Sparse Data

For non-sparse simulation data, we set the Poisson distribution variable with $\lambda = 20$. As stated earlier, we want to first test our algorithm on non-sparse data in order to ensure that it functions properly in accordance with the asymptotic distribution. If our simulations results align with established procedures, it would validate our algorithm as an appropriate approach for sparse data as well. We generated five tables of varying dimensions for this simulation.

1. 3x3x3
2. 3x3x4
3. 3x3x5
4. 3x3x6
5. 3x3x7
6. 4x4x4
7. 5x5x5

We expect two outcomes from the simulation results involving non-sparse tables. First, we expect to fail in rejecting the null hypothesis from the goodness-of-fit test results 95% of the time; we expect to fail in rejecting the null model that states a no-three-way interaction 95% of the time. Second, we expect to fail in rejecting the null hypothesis from the K–S Test results; we expect to see that the distribution of the test statistic comes from the asymptotic distribution.

3.3.2 Sparse Data

For sparse data, we set $\mu = \lambda = 6$ in Equation 2.1 so that it ensures that the observed cell counts do not meet the criteria needed to use the asymptotic distribution. Unlike non-sparse data, we do not know what to expect from the test results. We do not expect the K–S Test results to show that the distribution of the test statistic comes from the asymptotic distribution. We apply our algorithm to the following tables:

1. 3x3x3
2. 3x3x4
3. 3x3x5
4. 3x3x6
5. 3x3x7
6. 4x4x4
7. 5x5x5

We display the results for these tables in Chapter 4.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Mathematical Proof, Results, and Analysis

This chapter provides the mathematical proof of the connectivity for $3\times3\times K$ tables. It also examines the results of the simulations. We include the analysis for each type of simulation within the respective section.

4.1 Proof of Connectivity

In this section, we prove mathematically that the set of basic moves with the proposed scheme of allowing each cell count to assume values of -1 guarantees a connected MCMC for any marginal sums, under the no-three-way interaction model, in the case of $3 \times 3 \times K$ tables, for $K \geq 3$. The significance of the no-three-way interaction model involves how the model constrains the computation of the marginal sums. For example, suppose **b** represents a move in a $3 \times 3 \times 3$ table format. Then the no-three-way interaction model constrains the cell counts in **b** as follows:

$$\begin{aligned}
\textstyle\sum_{i=1}^{3} b_{ijk} &= 0,\\
\textstyle\sum_{j=1}^{3} b_{ijk} &= 0,\\
\textstyle\sum_{k=1}^{3} b_{ijk} &= 0.
\end{aligned}$$

Theorem 4.1.1 *Consider $3 \times 3 \times K$ contingency tables, for $K \geq 3$, under the no-three-way interaction model. A MCMC on the conditional state space, the set of all $3 \times 3 \times K$ contingency tables with fixed 2-dimensional marginal sums, connects with the set of basic moves if each cell count can assume values of -1. We label this connected MCMC as* ergodic.

Proof We use the results from Aoki and Takemura (2003) to begin our proof. In their study, Aoki and Takemura computed the minimal Markov basis for $3 \times 3 \times K$ contingency tables, for $K \geq 3$, under the no-three-way interaction model where seven combinatorial types categorized all possible moves in the Markov basis [18]. If we can show that we can connect the positive component to the negative component of each element of the minimal Markov basis through basic moves by allowing each cell to assume a -1 cell count value, we prove our theorem.

Let b represent a $I \times J \times K$ table such that,

		k	k'
i	j	1	−1
	j'	−1	1

		k	k'
i'	j	−1	1
	j'	1	−1

where $1 \leq i, i' \leq I$, $1 \leq j, j' \leq J$, $1 \leq k, k' \leq K$, $i \neq i'$, $j \neq j'$, and $k \neq k'$, and with all other cell counts at zero. We call $\pm b$ a *basic move*. We denote this table, b, as $(i, i'; j, j'; k, k')$.

Degree indicates the sum of positive or negative components of the move; the sum of the positive components must equal the sum of the negative components. For example,

		k	k'
i	j	1	0
	j'	0	1

		k	k'
i'	j	0	1
	j'	1	0

represents the positive component of $\mathbf{b}$, or $\mathbf{b}_+$. While,

		k	k'
i	j	0	1
	j'	1	0

		k	k'
i'	j	1	0
	j'	0	1

represents the negative component of **b**, or $\mathbf{b}_-$. Thus,

$$\mathbf{b} = \mathbf{b}_+ - \mathbf{b}_-$$

and *degree* equals the sum of the positive or negative component. In this case, we have a move with a degree of 4. We must also note that all matrices of varying degrees represent a sub-matrix of a move. For example, a 5 × 5 × 5 table may consist of the above move with a degree of 4 with all other cell counts equal zero.

We begin the proof with a move with a degree of 4; we require no further proof since all moves with a degree 4 in the Markov basis are already basic moves (as defined in Chapter 2.1).

For moves with a degree of 6, the following three categories characterize the moves after we apply permutations:

	+1	−1	0	0	−1	+1	0	0	0	0	0	0
(*i*)	0	+1	−1	0	0	−1	+1	0	0	0	0	0
	−1	0	+1	0	+1	0	−1	0	0	0	0	0
	+1	−1	0	0	0	+1	−1	0	−1	0	+1	0
(*ii*)	−1	+1	0	0	0	−1	+1	0	+1	0	−1	0
	0	0	0	0	0	0	0	0	0	0	0	0
	+1	−1	0	0	0	0	0	0	−1	+1	0	0
(*iii*)	−1	+1	0	0	+1	−1	0	0	0	0	0	0
	0	0	0	0	−1	+1	0	0	+1	−1	0	0

.

For type (i), we have

$$\begin{array}{rc}
 & \begin{array}{|c|c|c|c|}\hline +1 & -1 & 0 & 0\\\hline 0 & +1 & -1 & 0\\\hline -1 & 0 & +1 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline -1 & +1 & 0 & 0\\\hline 0 & -1 & +1 & 0\\\hline +1 & 0 & -1 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline\end{array} \\
= & \begin{array}{|c|c|c|c|}\hline +1 & 0 & 0 & 0\\\hline 0 & +1 & 0 & 0\\\hline 0 & 0 & +1 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline 0 & +1 & 0 & 0\\\hline 0 & 0 & +1 & 0\\\hline +1 & 0 & 0 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline\end{array} \\
- & \begin{array}{|c|c|c|c|}\hline 0 & 1 & 0 & 0\\\hline 0 & 0 & 1 & 0\\\hline 1 & 0 & 0 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline 1 & 0 & 0 & 0\\\hline 0 & 1 & 0 & 0\\\hline 0 & 0 & 1 & 0\\\hline\end{array}\;\begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline 0 & 0 & 0 & 0\\\hline\end{array} \\
=: & \mathbf{b}^{\mathbf{1}}_{+} - \mathbf{b}^{\mathbf{1}}_{-}.
\end{array}$$

If we allow $X_{ijk} \geq -1$ for $1 \leq i \leq I,\ 1 \leq j \leq J,\ 1 \leq k \leq K$, we can show that

$$\mathbf{b}^{\mathbf{1}}_{+} = \mathbf{b}^{\mathbf{1}}_{-} + (\mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}) + (\mathbf{1}, \mathbf{2}; \mathbf{2}, \mathbf{3}; \mathbf{1}, \mathbf{3}).$$

Similarly, for type (ii), we set the positive part of the move as $\mathbf{b}^{\mathbf{2}}_{+}$ and the negative part of the move as $\mathbf{b}^{\mathbf{2}}_{-}$. We can then write

$$\mathbf{b}^{\mathbf{2}}_{+} = \mathbf{b}^{\mathbf{2}}_{-} + (\mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}) + (\mathbf{2}, \mathbf{3}; \mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{3}).$$

Similarly, for type (iii), we set the positive part of the move as $\mathbf{b}^{\mathbf{3}}_{+}$ and the negative part of the move as $\mathbf{b}^{\mathbf{3}}_{-}$. We can now write

$$\mathbf{b}^{\mathbf{3}}_{+} = \mathbf{b}^{\mathbf{3}}_{-} + (\mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}; \mathbf{1}, \mathbf{2}) + (\mathbf{2}, \mathbf{3}; \mathbf{1}, \mathbf{3}; \mathbf{1}, \mathbf{2}).$$

For the move with a degree of 7 in the Markov basis, Aoki and Takemura showed that only one combinatorial type of move remained after applying these permutations [18]:

$$\begin{array}{|c|c|c|}\hline 0 & 0 & 0\\\hline 0 & +1 & -1\\\hline 0 & -1 & +1\\\hline\end{array}\;\begin{array}{|c|c|c|}\hline -1 & 0 & +1\\\hline +1 & -1 & 0\\\hline 0 & +1 & -1\\\hline\end{array}\;\begin{array}{|c|c|c|}\hline +1 & 0 & -1\\\hline -1 & 0 & +1\\\hline 0 & 0 & 0\\\hline\end{array}\,.$$

Similar to the moves with a degree of 6, we let $\mathbf{b}_{+}^{\mathbf{4}}$ represent the positive component of the move and let $\mathbf{b}_{-}^{\mathbf{4}}$ represent the negative component of the move. Then, we have

$$\mathbf{b}_{+}^{\mathbf{4}} = \mathbf{b}_{-}^{\mathbf{4}} - (\mathbf{1},\mathbf{2};\mathbf{2},\mathbf{3};\mathbf{2},\mathbf{3}) + (\mathbf{2},\mathbf{3};\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{3}).$$

For the move with a degree of 8 in the Markov basis, Aoki and Takemura identified two combinatorial types of the move after applying the permutation:

$$(i)\quad \begin{array}{|c|c|c|c|}\hline +1 & -1 & 0 & 0 \\ \hline -1 & +1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|}\hline -1 & 0 & +1 & 0 \\ \hline +1 & 0 & 0 & -1 \\ \hline 0 & 0 & -1 & +1 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|}\hline 0 & +1 & -1 & 0 \\ \hline 0 & -1 & 0 & +1 \\ \hline 0 & 0 & +1 & -1 \\ \hline \end{array}$$

$$(ii)\quad \begin{array}{|c|c|c|c|}\hline 0 & +1 & 0 & -1 \\ \hline 0 & 0 & -1 & +1 \\ \hline 0 & -1 & +1 & 0 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|}\hline 0 & -1 & 0 & +1 \\ \hline +1 & 0 & 0 & -1 \\ \hline -1 & +1 & 0 & 0 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0 \\ \hline -1 & 0 & +1 & 0 \\ \hline +1 & 0 & -1 & 0 \\ \hline \end{array}\ .$$

We let $\mathbf{b}_{+}^{\mathbf{5}}$ represent the positive component of the move (i) and let $\mathbf{b}_{-}^{\mathbf{5}}$ represent the negative component of the move (i). We also let $\mathbf{b}_{+}^{\mathbf{6}}$ represent the positive component of the move (ii) and let $\mathbf{b}_{-}^{\mathbf{6}}$ represent the negative component of the move (ii). Then, we have

$$\begin{aligned} \mathbf{b}_{+}^{\mathbf{5}} &= \mathbf{b}_{-}^{\mathbf{5}} + (\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{2}) - (\mathbf{2},\mathbf{3};\mathbf{1},\mathbf{2};\mathbf{2},\mathbf{3}) + (\mathbf{2},\mathbf{3};\mathbf{2},\mathbf{3};\mathbf{3},\mathbf{4}), \\ \mathbf{b}_{+}^{\mathbf{6}} &= \mathbf{b}_{-}^{\mathbf{6}} + (\mathbf{2},\mathbf{3};\mathbf{2},\mathbf{3};\mathbf{1},\mathbf{3}) - (\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{3};\mathbf{2},\mathbf{4}) + (\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{3};\mathbf{2},\mathbf{4}). \end{aligned}$$

For the move with a degree of 10, Aoki and Takemura showed that only one combinatorial type of move remained after applying the permutations:

$$\begin{array}{|c|c|c|c|c|}\hline +1 & -1 & 0 & 0 & 0 \\ \hline -1 & +1 & 0 & -1 & +1 \\ \hline 0 & 0 & 0 & +1 & -1 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|c|}\hline -1 & 0 & +1 & 0 & 0 \\ \hline +1 & 0 & 0 & 0 & -1 \\ \hline 0 & 0 & -1 & 0 & +1 \\ \hline \end{array}\ \begin{array}{|c|c|c|c|c|}\hline 0 & +1 & -1 & 0 & 0 \\ \hline 0 & -1 & 0 & +1 & 0 \\ \hline 0 & 0 & +1 & -1 & 0 \\ \hline \end{array}\ .$$

Similar to the moves with a degree of 8, we let $\mathbf{b}_{+}^{\mathbf{7}}$ represent the positive component of the move and let $\mathbf{b}_{-}^{\mathbf{7}}$ represent the negative component of the move. We now have

$$\mathbf{b}_{+}^{\mathbf{7}} = \mathbf{b}_{-}^{\mathbf{7}} + (\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{2};\mathbf{1},\mathbf{2}) - (\mathbf{2},\mathbf{3};\mathbf{1},\mathbf{2};\mathbf{2},\mathbf{3}) + (\mathbf{2},\mathbf{3};\mathbf{2},\mathbf{3};\mathbf{3},\mathbf{4}) - (\mathbf{1},\mathbf{2};\mathbf{2},\mathbf{3};\mathbf{4},\mathbf{5}).$$

Since we can write all seven types of moves in the Markov basis as a linear combination of basic moves if we allow cell count values of -1, we conclude the proof.

4.2 Results

We examine the results of the simulations in this section. Each trial of the simulation follows the procedures as outlined in Section 3.2.1, with the elaboration of the tests in subsequent sections in Chapter 3. Please note that we include every resulting histogram, displaying the distribution of χ^2 test statistics, in Appendix B.

As outlined in Section 3.2.1, both non-sparse and sparse simulations conducted two tests: the goodness-of-fit test and the K–S Test. We employ the goodness-of-fit test for model fitting, where we analyze the relationship between the three variables. Recall that we considered the hypotheses:

$$H_0: \ \lambda_{ijk}^{TYZ} = 0.$$
$$H_1: \ \lambda_{ijk}^{TYZ} \neq 0.$$

We begin by setting the significance level, α, at .05 and assuming that the null hypothesis is true. Thus, we only reject the null hypothesis, which states that there is no interaction between the three variables in favor of the alternative hypothesis, which states that there is an interaction between them, if the p-value calculates below .05.

For K–S Test, we test the following hypotheses:

H_0 : Null distribution of test statistics resembles the χ^2 distribution.

H_1 : Null distribution of test statistics does not resemble the χ^2 distribution.

Similar to the goodness-of-fit test, the p-value dropping below .05 indicates a strong evidence against the null hypothesis.

4.2.1 Non-Sparse Data

For each unique dimension, the simulation involved ten trials of $n = 10,000$ samples. Each trial begins with a generation of a contingency table. We set the burn-in value, B, at $2,500$. The following table outlines the average p-value from each simulation. Since the tables meet the requirements of the asymptotic distribution, we expect the results to follow.

Dimensions	Goodness-of-Fit Avg. P-value	K–S Test Avg. P-value
$3 \times 3 \times 3$	0.3259	0.4571
$3 \times 3 \times 4$	0.6169	0.4007
$3 \times 3 \times 5$	0.3959	0.2093
$3 \times 3 \times 6$	0.6146	0.2631
$3 \times 3 \times 7$	0.6523	0.2818
$4 \times 4 \times 4$	0.4970	0.2363
$5 \times 5 \times 5$	0.3664	0.2480

.

Analysis

As expected, we fail to reject the null hypothesis for both tests. The results of the goodness-of-fit test shows that the no-three-way interaction model appropriately fits with the observed data. The K–S Test shows that the null distribution of the test statistic matches the χ^2 distribution, validating our algorithm with established norms. In other words, the distribution of the test statistics converges to the χ^2 distribution. Therefore, we can conclude that our proposed MCMC sampling method properly functions in sampling three-way tables under a no-three-way interaction model.

4.2.2 Sparse Data

For dimensions $3 \times 3 \times K$ for $K \geq 3$, the simulation involved ten trials of $n = 100,000$ samples. For larger dimensions, we decreased to a single trial but increased the samples to $n = 10,000,000$. We increased the sample size in order to avoid having the simulation get stuck at a particular state. We set the burn-in value, B, at 25% of the sample size. The following table outlines the average p-value from each simulation. Since the tables do not meet the requirements of the asymptotic distribution, we anticipate that the distribution of the test statistics do not converge to the χ^2 distribution.

Dimensions	Goodness-of-Fit Avg. P-value	K–S Test Avg. P-value
$3 \times 3 \times 3$	0.2650	0.0000
$3 \times 3 \times 4$	0.3449	0.0000
$3 \times 3 \times 5$	0.1295	0.0000
$3 \times 3 \times 6$	0.0957	0.0000
$3 \times 3 \times 7$	0.2361	0.0000
$4 \times 4 \times 4$	0.7742	0.0000
$5 \times 5 \times 5$	0.9119	0.0000

.

Analysis

Since all of the p-values exceeds the significance threshold at $\alpha = .05$, we conclude that, on average, the no-three-way interaction model appropriately fits with the observed data. Although too difficult to prove mathematically, the simulation also shows that for larger tables, such as $4 \times 4 \times 4$ shown in Figure B.13 and B.14, the proposed method appears to sample contingency tables from the conditional state space without bias since the distribution of test statistics looks unimodal. If not well mixed, then the chain tends to get stuck in some states and it causes a multimodal distribution. A smooth unimodal distribution shows strong evidence that we sampled everywhere in the state space.

The K–S Test showed that the χ^2 distribution does not accurately summarize the null distribution of the test statistic for sparse contingency tables, as seen from highly significant p-values. Based on these results, we can reasonably conclude that using the asymptotic distribution for sparse tables may result incorrect conclusions during goodness-of-fit testing. We can thus conclude that our proposed MCMC sampling method properly functions in sampling sparse three-way tables under a no-three-way interaction model for sparse data.

CHAPTER 5: Conclusion

Our proposed MCMC sampling method properly functions in sampling three-way tables, under a no-three-way interaction model, for both sparse and non-sparse tables. In Chapter 4, we proved mathematically that the proposed method can sample a table from anywhere in the conditional state space without any bias in the case of $3 \times 3 \times K$ tables for $K \geq 3$. We believe our algorithm contributes to the development of accurate algorithms for CDA of sparse data and exact conditional tests.

Our simulation results show strong evidence that for larger tables, such as $4 \times 4 \times 4$ and greater, the proposed method samples contingency tables from anywhere in the conditional state space without bias. However, we found it too difficult to mathematically prove that this algorithm samples without bias. We can deduce that our proposed MCMC method simulates a Markov chain on the connected transition graph so that it will not produce a sampling bias by observing the unimodal distribution of test statistics for sparse data; a not-well-mixed chain generally outputs a multimodal distribution of test statistics.

The results of both simulations allow us to conclude that the no-three-way interaction model fits well with the three-way tables sampled from the conditional distribution. For non-sparse tables, the K–S Test shows that the χ^2 distribution accurately summarizes the null distribution of the test statistics, validating our algorithm with established norms. Furthermore, the K–S Test showed that the χ^2 distribution does not accurately summarize the null distribution of the test statistics for sparse contingency tables. Thus, we can reasonably conclude that for sparse tables, the χ^2 distribution does not accurately approximate the null distribution of the test statistics.

Since contingency tables will likely remain as a widely used means of analysis for CDA, we anticipate this algorithm to further the development of sampling and testing methods for MCMC. Our proposed algorithm can sample any sparse or non-sparse $I \times J \times K$ contingency table, under the no-three-way interaction model, in any field of research for conducting goodness-of-fit tests. For sparse tables, our method estimates the null distribution of test statistics more accurately than traditional methods that involve asymptotic distributions.

5.1 Follow-On Work

For follow-on work, we recommend configuring the algorithm to allow the function to sample tables where some marginal sums equate to zero. Additionally, we recommend applying this algorithm to real-world data in fields of data and information science. Finally, integrating this modification with SIS may result in an even more efficient sampling algorithm for multidimensional contingency tables.

APPENDIX A: Simulation R Code

```
library(cat) ## library for MLE IPF

## function to check that two arguments share the same marginal
   sums
marginals_check <- function(mat1,mat2) { ## two matrices to
   compare as arguments
  confirmation <- ((rowSums(mat1)==rowSums(mat2)) && (colSums(
   mat1)==colSums(mat2)))
  return (confirmation)
}

## function to check that two arguments share the same marginal
   sums in 3-dimensions
marginals_check3d <- function(mat1,mat2) { ## two matrices to
   compare as arguments
  x1 <- apply(mat1,c(2,3),sum)
  y1 <- apply(mat1,c(1,3),sum)
  z1 <- apply(mat1,c(1,2),sum)
  x2 <- apply(mat2,c(2,3),sum)
  y2 <- apply(mat2,c(1,3),sum)
  z2 <- apply(mat2,c(1,2),sum)
  confirmation <- ((x1==x2) && (y1==y2) && (z1==z2))
  return (confirmation)
}

## function to generate 3-dimensional tables, with lambda from
   the Poisson distribution for sampling
## this function ensure that generated tables do not have any
   marginal sums that equal zero
gen.tables <- function(d,lambda=6,nonzero=FALSE) {   ## d:
    l <- prod(d)
    u <- rpois(1,lambda)
    x <- array(u,d)
    sum1 <- apply(x,c(1,2),sum)
    sum2 <- apply(x,c(1,3),sum)
    sum3 <- apply(x,c(2,3),sum)
    if (nonzero==TRUE) {
        while (sum(sum1==0)+sum(sum2==0)+sum(sum3==0) > 0) {
            u <- rpois(1,lambda)
            x <- array(u,d)
            sum1 <- apply(x,c(1,2),sum)
            sum2 <- apply(x,c(1,3),sum)
            sum3 <- apply(x,c(2,3),sum)
        }
    }
    return(x)
}

## function to ensure all cell counts are positive
pos_check <- function(x) { ## contingecy table as argument
  all(x %in% 0:max(x))
}

## function that serves as a component in allowing negative cell
    counts
slack_check <- function(x,k) {
  ## x: contingency table
  ## k: maximum number of cells allowed to leave the conditional
    state space
  (all(x %in% -1:max(x)) && (sum(x==-1) <= k))
}

## function that returns the proper index necessary for
    randomization of matrix permutations
## code involves a simple modulus calculation so that
    permutations remain within the matrix index regardless of
```

```
   matrix size
get.index <- function(x,k) {
  ## x: random index
  ## k: variable representing the number of samples
  if (x%%k == 0) k
  else x%%k
}

## function to compute z matrix (basic move)
moves <- function(r,c) {
  ## r: number of rows
  ## c: number of columns
  k <- min(c(r,c)) # constraint variable to limit z matrix
  L <- sample(2:k,1)
  ## note: L represents the randomization of potential z matrix (
   basic move) sizes
  R <- sample(1:r,L)
  C <- sample(1:c,L)
  zz <- diag(L)

  for (i in 1:(L-1)) zz[i, i+1] <- -1
  zz[L,1] <- -1

  z <- zz
  II <- sample(1:L, 1)
  for (i in 1:L) z[i,] <- zz[get.index(II+i,L),]
  ## alternative approach (permute columns):
  for(i in 1:L) z[,i] <- zz[,get.index(II+i, L)]
  z.mat <- array(rep(0,r*c),c(r,c))
  for (i in 1:L)
    for (j in 1:L)
      z.mat[R[i], C[j]] <- z[i,j]
  return (z.mat)
}
```

```
## function that supports Metropolis−Hasting algorithm;
   calculates ratio test statistic
compute.ratio <− function(current,proposal) {
    ## current: initial table
    ## proposal: proposed table
    r <− 0
    m <− dim(current)[1]
    n <− dim(current)[2]
    l <− dim(current)[3]
    for (i in 1:m) {
        for (j in 1:n) {
            for (k in 1:l) {
                ##if(current[i, j, k] != proposal[i, j, k])
                    r <− r + (lfactorial(current[i,j,k]) −
   lfactorial(proposal[i,j,k]))
            }
        }
    }
    return(min(r,0))
}

## function that conducts the proposed approach; mcmc sampling
proposed.alg <− function (x_current,N) {
  ## disclaimer: this function slices matrices by height
  ## x_current: 3−dimensional 0−1 table
  ## N: number of desired sample matrices
  samples <− list() # initiate list to collect acceptable
   matrices (empty set, S)
  d <− dim(x_current) # set variable as dimensions of argument
   matrix (list of integers)

  ## while loop to collect acceptable matrices after conducting
   mcmc basic moves
  i <− 1 # initiate counter variable
  samples[[i]] <− x_current
```

```
i <- 2
mat <- x_current # initiate iteration variable to take permutations of tables without modifying original
success <- FALSE # initiate success-condition boolean variable
while (!success) {
    t <- sample(c(1:d[3]),2,replace=F) # select 2 time steps (re: disclaimer)
    a <- mat[,,t[1]] # first data table
    b <- mat[,,t[2]] # second data table
    z <- moves(d[1],d[2]) # step matrix via proposed method
    x_star1 <- a + z # proposal matrix 1
    x_star2 <- b - z # proposal matrix 2
    test1 <- marginals_check(a,x_star1) # ensure marginal sums match between original and proposal matrices
    test2 <- marginals_check(b,x_star2) # ensure marginal sums match between original and proposal matrices
    ## tests 3 and 4 limit the number of -1 cell count values allowed
    test3 <- slack_check(x_star1,d[1]*d[2]) # ensure cells fall within acceptable range of slack
    test4 <- slack_check(x_star2,d[1]*d[2]) # ensure cells fall within acceptable range of slack
    ## cat(test1," ",test2," ",test3," ",test4,"\n")
    if (test1 && test2 && test3 && test4 == TRUE) { # if the proposed tables meet all criteria
        mat[,,t[1]] <- x_star1 # replace original table with new table
        mat[,,t[2]] <- x_star2 # replace original table with new table
        if (pos_check(mat) == TRUE) { # if all cells within the new 3-way matrix are binary
            r <- compute.ratio(samples[[i-1]],mat)
            ## cat(r, "\n")
            if (runif(1) <= exp(r))
```

```
                  samples[[i]] <- mat # append new matrix to the
   collection set
              else {
                  samples[[i]] <- samples[[i-1]]
                  mat <- samples[[i-1]]
              }
          }
          else { # if proposal has negative values
              samples[[i]] <- samples[[i-1]] # stay at the same
   state
          }
      } else {
          samples[[i]] <- samples[[i-1]]
      }
      i <- i+1
      success <- (length(samples)==N+1) # success condition
  }
  return(samples)
}

## function to calculate MLE for three-dimensional tables
MLE.IPF <- function(x) {
    ## x: observed table
    m <- c(1,2,0,1,3,0,2,3)
    fit1 <- ipf(x, margins=m, showits=TRUE)
    return(fit1)
}

## function to calculate Pearson's Chi-square test statistics
chisqStat <- function(x0, expected) {
  ## x0: sample table
  ## expected: MLE
  chiMatrix <- (x0-expected)^2/expected
  statValue <- sum(chiMatrix)
  return(statValue)
```

```
}

m<-7; n<-7; l<-7; # matrix dimensions
## Simulation characteristics
N <- 10000000 # sample size
B <- .25*N # number of burn-in (number of initial samples to
    ignore)
S <- 1 # thinning value (increments of samples to skip)

## Collection variables
#t <- 0 # computation times for each trial
#p1 <- 0 # p-values from observed tables
#p2 <- 0 # p-values from Kolmogorov-Smirnov Test for each trial

## Simulation
observed <- gen.tables(c(m,n,l),nonzero=TRUE)
mu <- MLE.IPF(observed)
f0 <- chisqStat(observed, mu)
obs <- 0 # initialized empty set for observed Chi-sq test
    statistic from sample
start.time <- Sys.time() # begin computation time
tmp <- proposed.alg(observed, N*S+B) # first iteration of the
    simulation
obs <- sapply(tmp, chisqStat, mu) # collection of test statistics
end.time <- Sys.time() # end computation time
t <- end.time-start.time # duration

## Analysis:
## burn the first B samples and thin every S samples; adjusted
    observations
## burn-in & thinning mitigates the effects of bias
adj.obs <- obs[seq(B, length(obs), S)]
## manually calculate p-value; observed p-value
obs.pvalue <- sum(adj.obs>f0)/length(adj.obs)
adj.obs <- obs[seq(B, length(obs), S)]
```

```
## manually calculate p−value; observed p−value
obs.pvalue <− sum(adj.obs>f0)/length(adj.obs) # goodness−of−fit
    test
## histogram of observations fitted with a chi−square curve
wfile <− sprintf("sparse_hist_7x7x7.pdf")
pdf(file = wfile)
hist(adj.obs, breaks=100, probability=T, xlab="Observed␣Chi−
    Squared␣Statistics",
  main="Chi−Square␣Density␣Graph␣for␣7x7x7␣Sparse␣Contingency␣
    Table")
x <− pchisq(f0, (dim(observed)[1]−1)*(dim(observed)[2]−1)*(dim(
    observed)[3]−1), lower.tail=FALSE)
## establish the standard curve for a chi−sq distribution for
    comparison
curve(dchisq(x, (dim(observed)[1]−1)*(dim(observed)[2]−1)*(dim(
    observed)[3]−1)), col='red', add=T)
## diagnostics
test.set <− rchisq(1000, (dim(observed)[1]−1)*(dim(observed)
    [2]−1)*(dim(observed)[3]−1))
dev.off()

## Results
t # simulation duration
obs.pvalue # results of the Goodness−of−Fit Test (Null Hypothesis
    : No−Three−Way Interaction)
ks.test(test.set, adj.obs)$p.value # results of the ks.test (null
     hypothesis: Chi−squared distribution)

## Simulations for 10 trials
#for (j in 1:10) { # 10 trials of simulations
#   observed <− gen.tables(c(m,n,l),nonzero=TRUE)
#   mu <− MLE.IPF(observed)
#   f0 <− chisqStat(observed, mu)
#   obs <− 0 # initialized empty set for observed Chi−sq test
    statistic from sample
#   start.time <- Sys.time() # begin computation time
#   tmp <- proposed.alg(observed, N*S+B) # first iteration of the
    simulation
#   obs <- sapply(tmp, chisqStat, mu)
#   end.time <- Sys.time()
#   t[j] <- end.time-start.time
#   ## Trial analysis:
#   ## burn the first B samples and thin every S samples; adjusted
     observations
#   ## burnning & thinning mitigates the effects of bias
#   adj.obs <- obs[seq(B, length(obs), S)]
#   ## manually calculate p-value; observed p-value
#   obs.pvalue <- sum(adj.obs>f0)/length(adj.obs)
#   p1[j] <- obs.pvalue
#   ## histogram of observations fitted with a chi-square curve
#   wfile <- sprintf("sparse_hist_6x6x6_%d.pdf", j)
#   pdf(file = wfile)
#   hist(adj.obs, breaks=100, probability=T, xlab="Observed Chi-
    Squared Statistics",
#     main="Chi-Square Density Graph for 6x6x6 Sparse Contingency
    Table")
#   x <- pchisq(f0, (dim(observed)[1]-1)*(dim(observed)[2]-1)*(dim
    (observed)[3]-1), lower.tail=FALSE)
#   ## establish the standard curve for a chi-sq distribution for
    comparison
#   curve(dchisq(x, (dim(observed)[1]-1)*(dim(observed)[2]-1)*(dim
    (observed)[3]-1)), col=’red’, add=T)
#   ## diagnostics
#   test.set <- rchisq(1000, (dim(observed)[1]-1)*(dim(observed)
    [2]-1)*(dim(observed)[3]-1))
#   dev.off()
#   p2[j] <- ks.test(test.set, adj.obs)$p.value}
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B:
Simulation Results

Every histogram in this appendix displays the distribution of the test statistics from each simulation. The red line denotes the χ^2 distribution. Note that distribution of test statistics from non-sparse data fits well with the χ^2 distribution, indicating that the χ^2 distribution accurately estimates the null distribution of test statistics. However, the distribution of the test statistics from sparse data clearly does not fit the asymptotic distribution. Since the distribution does not converge to the asymptotic distribution, we should not use the asymptotic distribution as the null distribution of test statistics for sparse data. These graphs supplement the analysis of the results as described in Sections 4.2.1 and 4.2.2. Please refer to Appendix A for the codes used to generate the simulations.

B.1 Non-sparse Data Histograms

This section displays the outputs of the simulations described in Section 4.2.1. The simulation ran for ten trials, each with a sample size of $n = 10,000$ and burn-in value of $B = 2,500$. These simulations provide strong evidence that our proposed algorithm aligns with historical findings from traditional methods. The histograms show that the null distribution of the test statistic matches the χ^2 distribution, per established norms. In other words, the distribution of the test statistics converges to the χ^2 distribution.

B.1.1 $3 \times 3 \times 3$ Non-Sparse Tables

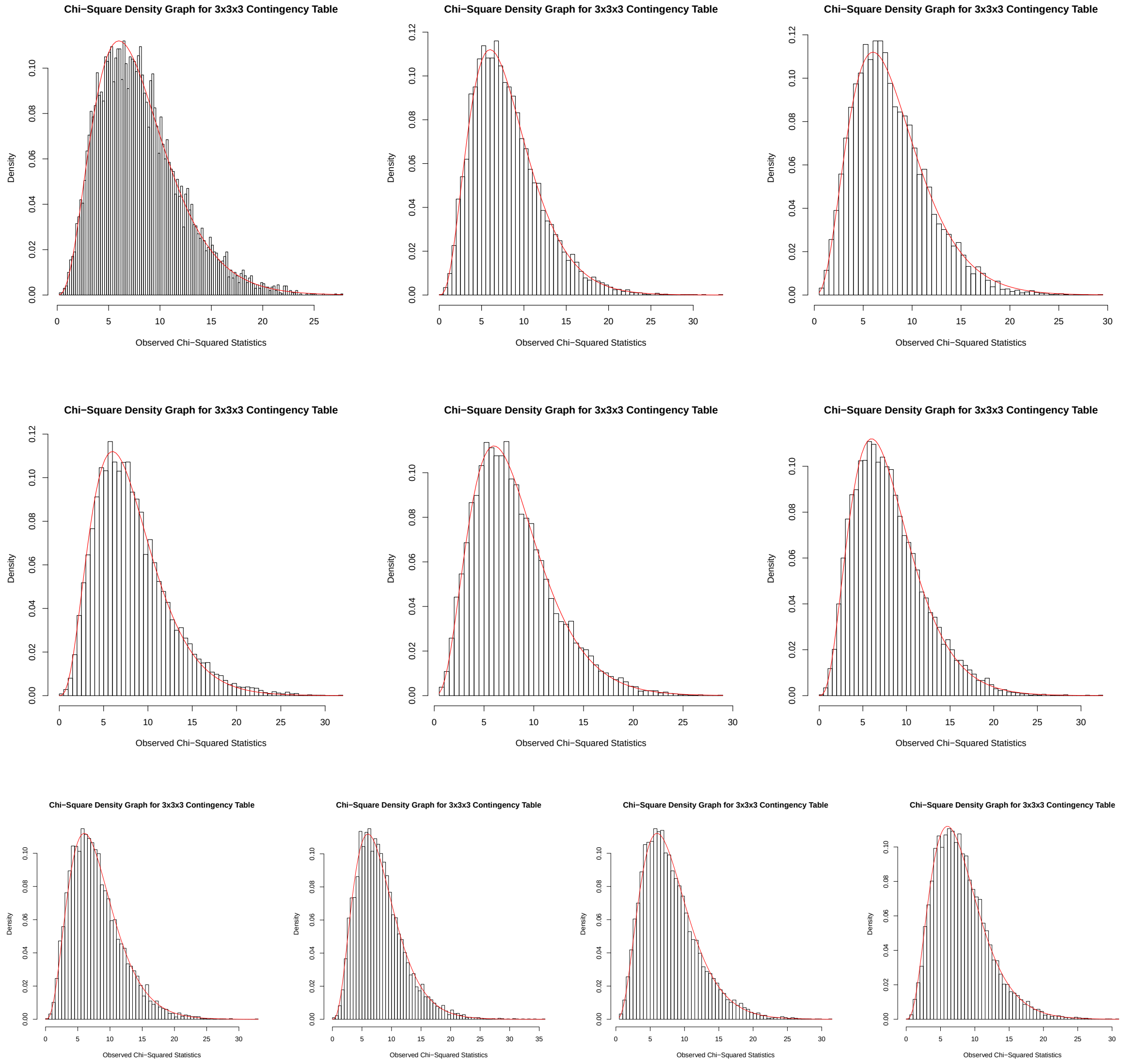


Figure B.1. Histogram of $3 \times 3 \times 3$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.2 $3 \times 3 \times 4$ Non-Sparse Tables

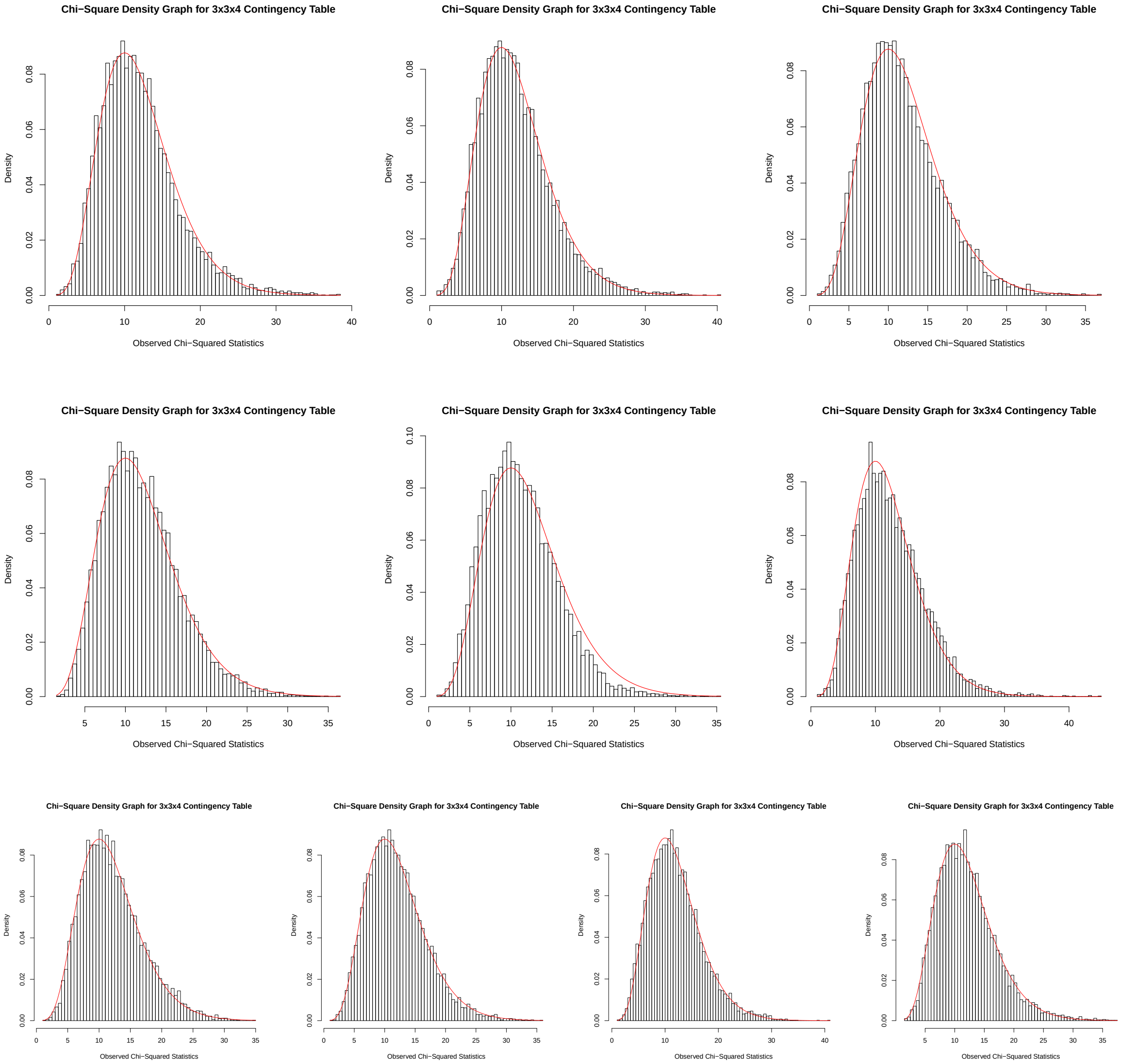


Figure B.2. Histogram of $3\times3\times4$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.3 $3 \times 3 \times 5$ Non-Sparse Tables

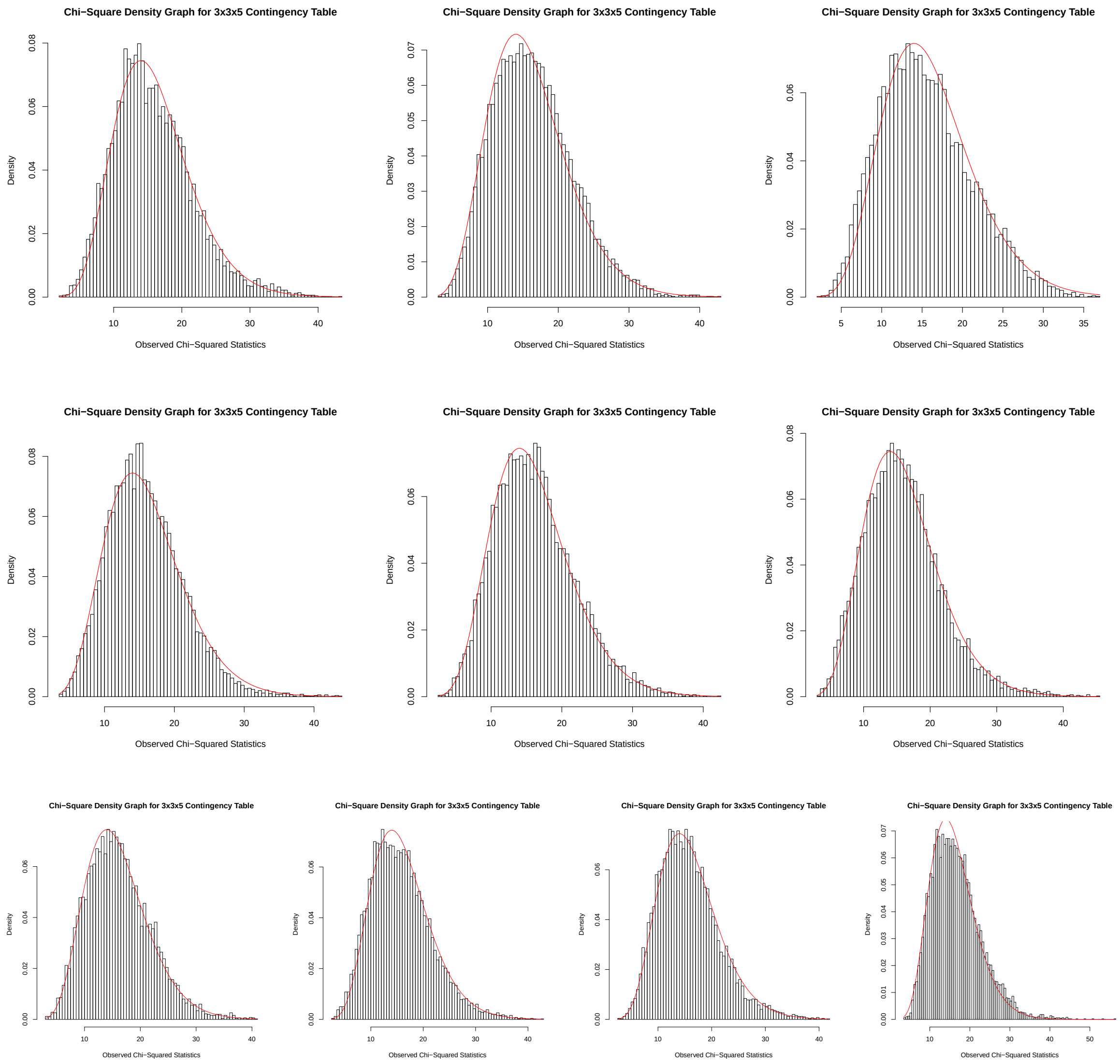


Figure B.3. Histogram of $3 \times 3 \times 5$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.4 $3 \times 3 \times 6$ Non-Sparse Tables

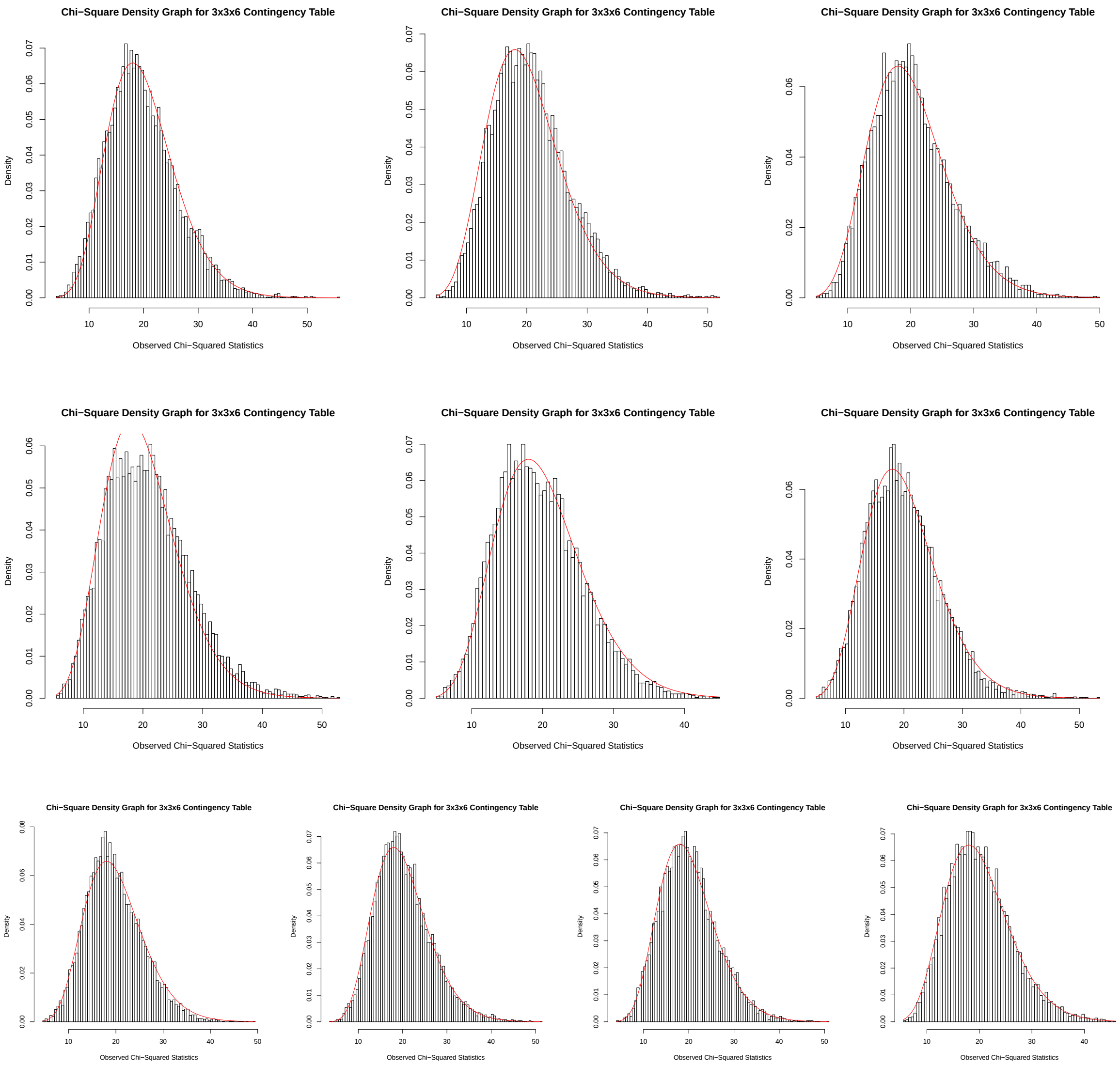
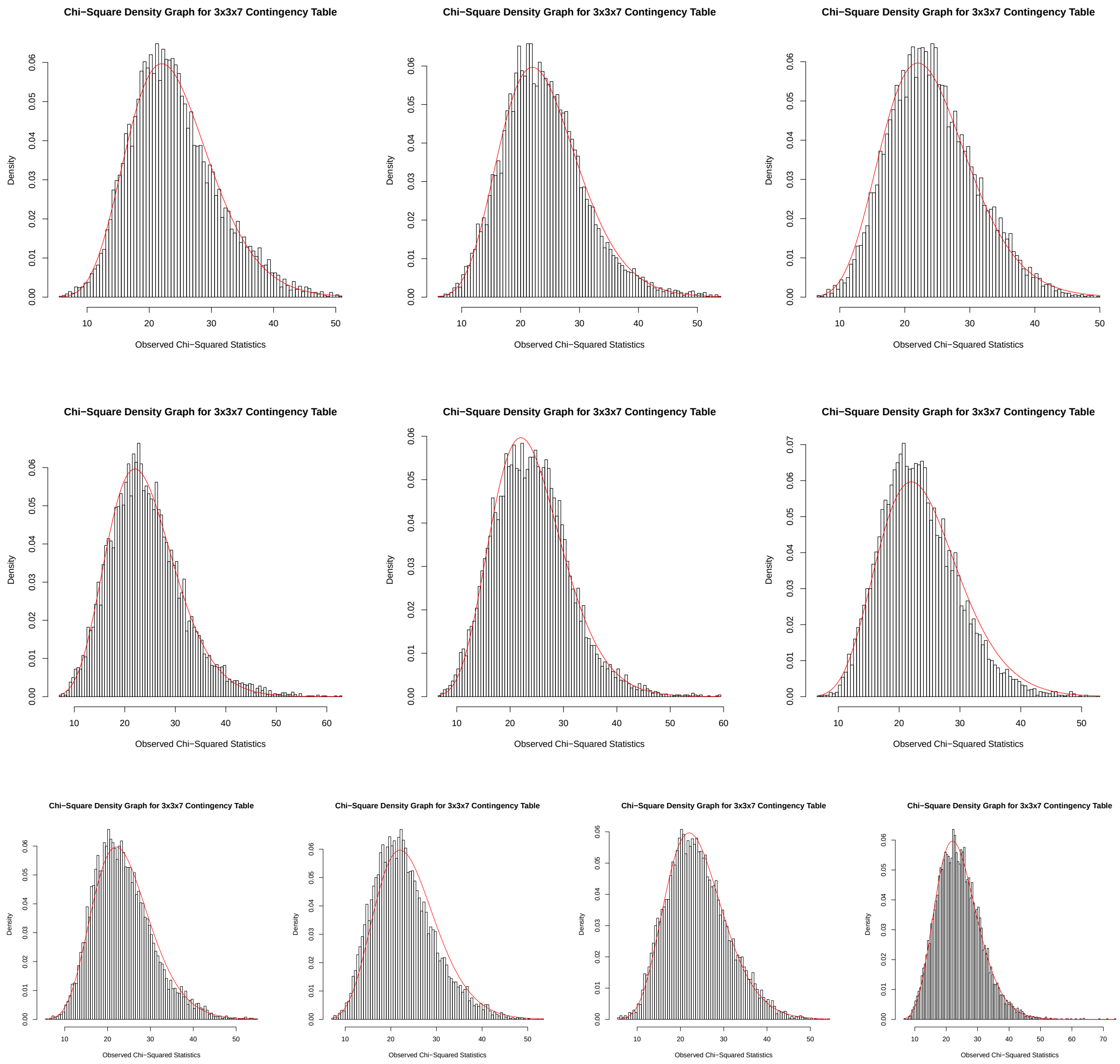


Figure B.4. Histogram of $3 \times 3 \times 6$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.5 $3 \times 3 \times 7$ Non-Sparse Tables



Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Chi−Square Density Graph for 3x3x7 Contingency Table

Figure B.5. Histogram of $3 \times 3 \times 7$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.6 $4 \times 4 \times 4$ Non-Sparse Tables

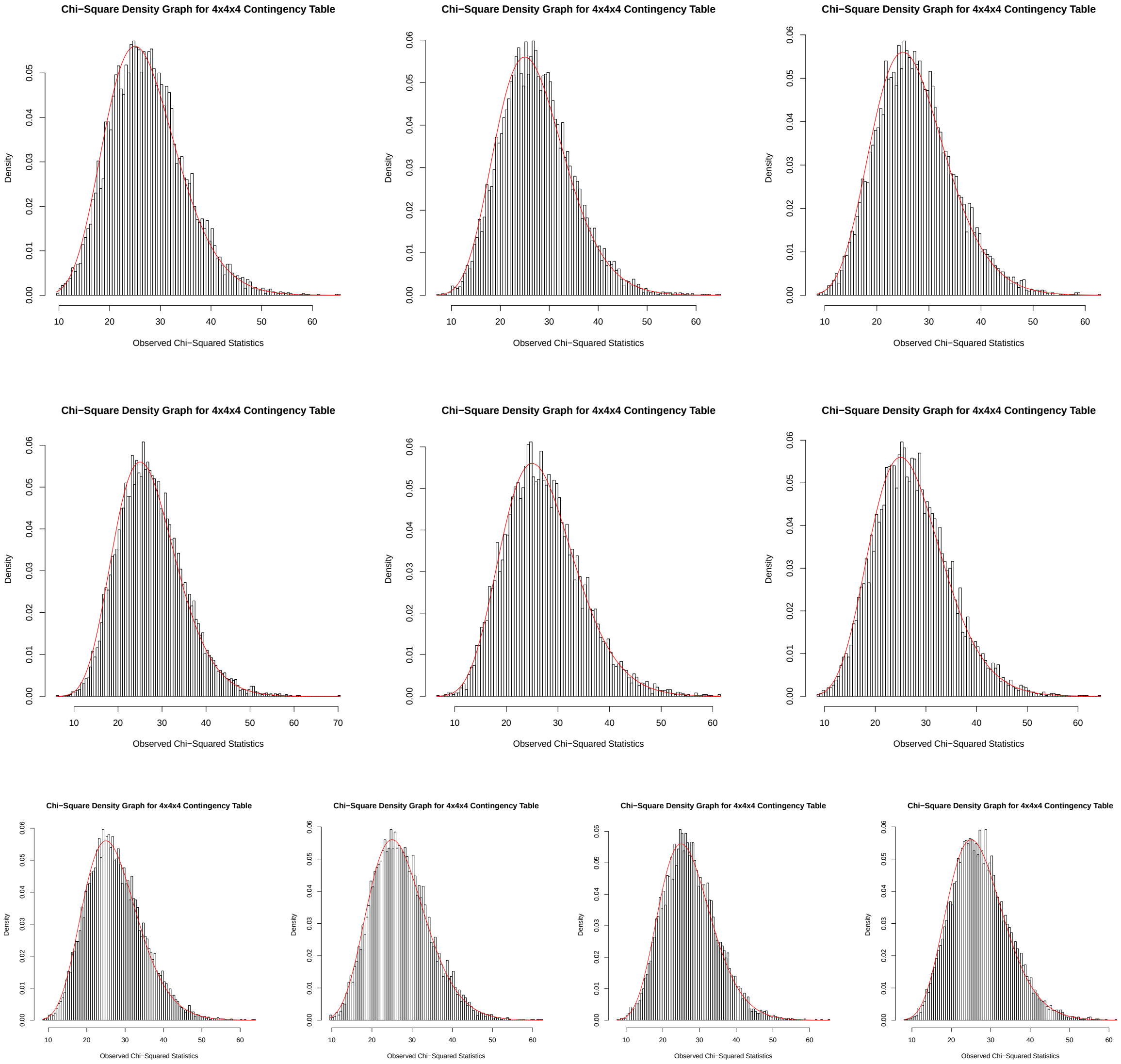


Figure B.6. Histogram of $4 \times 4 \times 4$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.1.7 $5 \times 5 \times 5$ Non-Sparse Tables

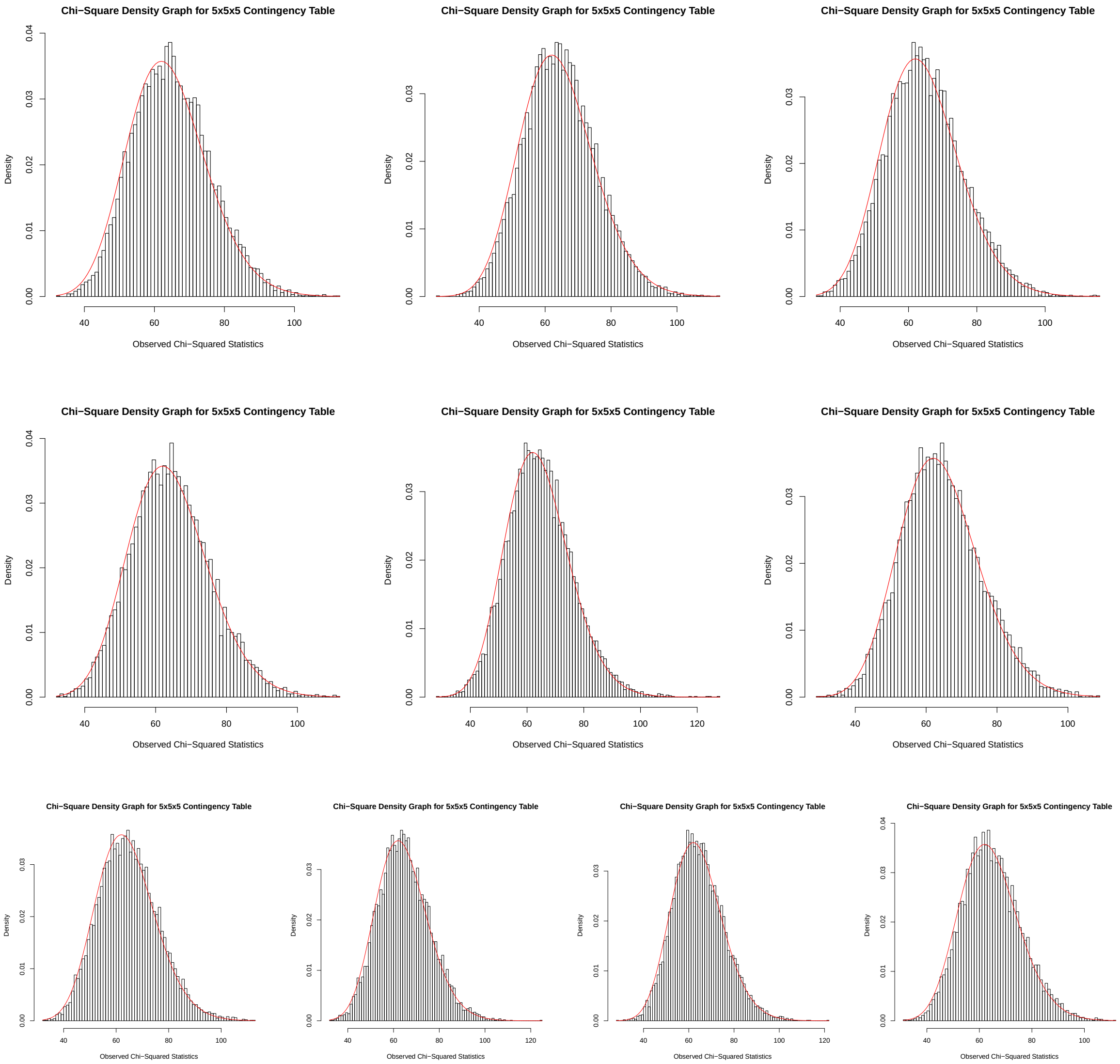
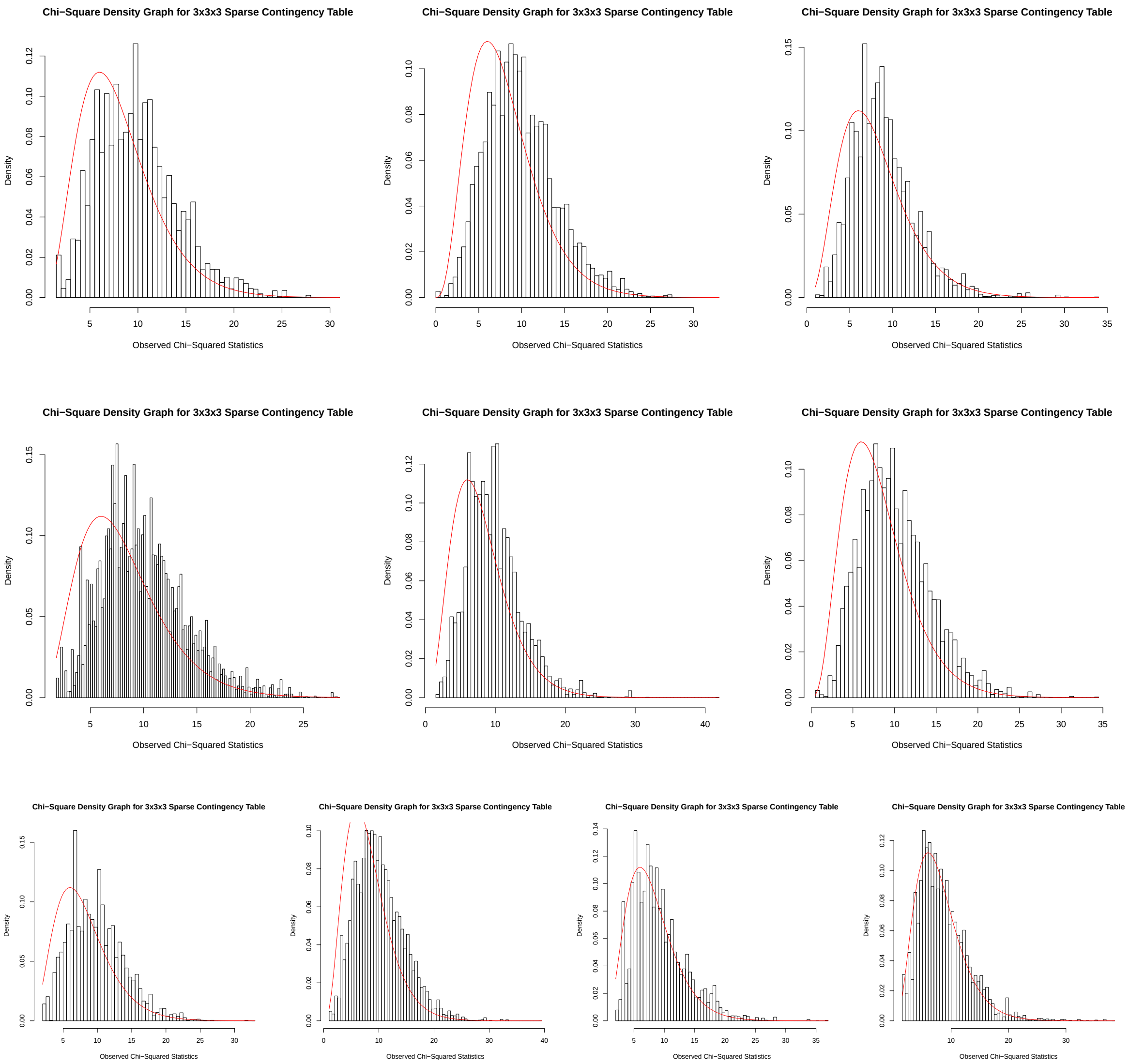


Figure B.7. Histogram of $5 \times 5 \times 5$ Tables' Distribution of Test Statistics for all trials. The distribution converges to the χ^2 distribution, as expected for non-sparse data. The red curve indicates the χ^2 distribution.

B.2 Sparse Data Histograms

This section displays the outputs of the simulations described in Section 4.2.2. For $3 \times 3 \times K$ tables, the simulation ran for ten trials, each with a sample size of $n = 1,000,000$ and burn-in value of $B = 250,000$. We increase the sample size because of the tendency for states to get stuck in the same state for sparse tables, requiring an increased number of samples in order to accurately capture the distribution. For larger tables, we increase the sample size further to $n = 10,000,000$ but only run the simulation for a single trial due to run time. Although too difficult to prove mathematically, the simulations show that for larger tables, the proposed method appears to sample contingency tables from the conditional state space without bias since the distribution of test statistics looks unimodal. If not well mixed, then the chain tends to get stuck in some states and it causes a multimodal distribution. A smooth unimodal distribution shows strong evidence that we sampled everywhere in the state space. These simulations provide strong evidence that the distribution of test statistics do not converge to the χ^2 distribution, supporting our claim that traditional approaches to analyzing sparse data may yield inaccurate or biased results.

B.2.1 $3 \times 3 \times 3$ Sparse Tables


Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

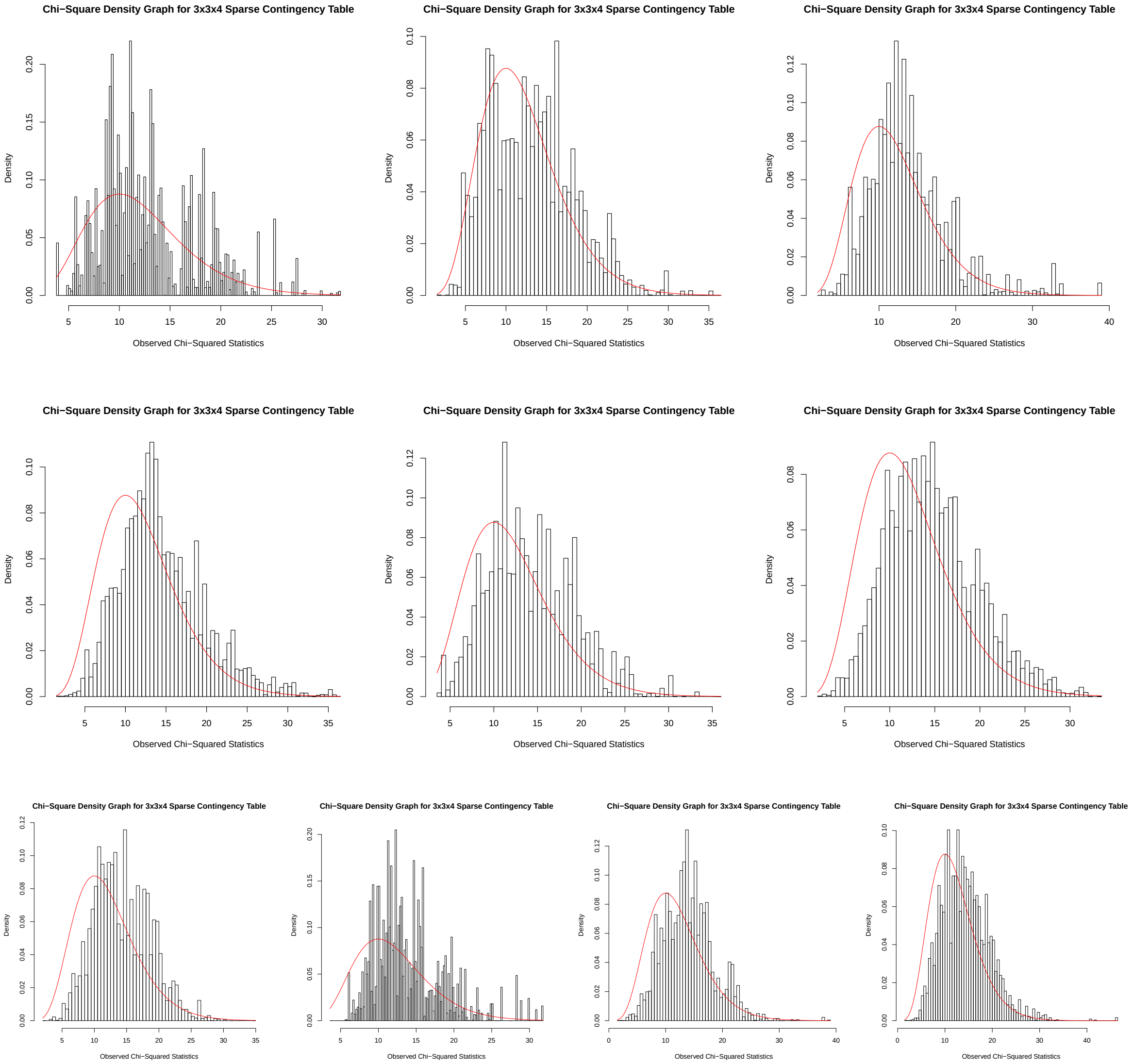
Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Chi−Square Density Graph for 3x3x3 Sparse Contingency Table

Figure B.8. Histogram of $3 \times 3 \times 3$ Tables' Distribution of Test Statistics for all trials. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.2 $3 \times 3 \times 4$ Sparse Tables


Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Chi−Square Density Graph for 3x3x4 Sparse Contingency Table

Figure B.9. Histogram of $3 \times 3 \times 4$ Tables' Distribution of Test Statistics for all trials. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.3 $3 \times 3 \times 5$ Sparse Tables

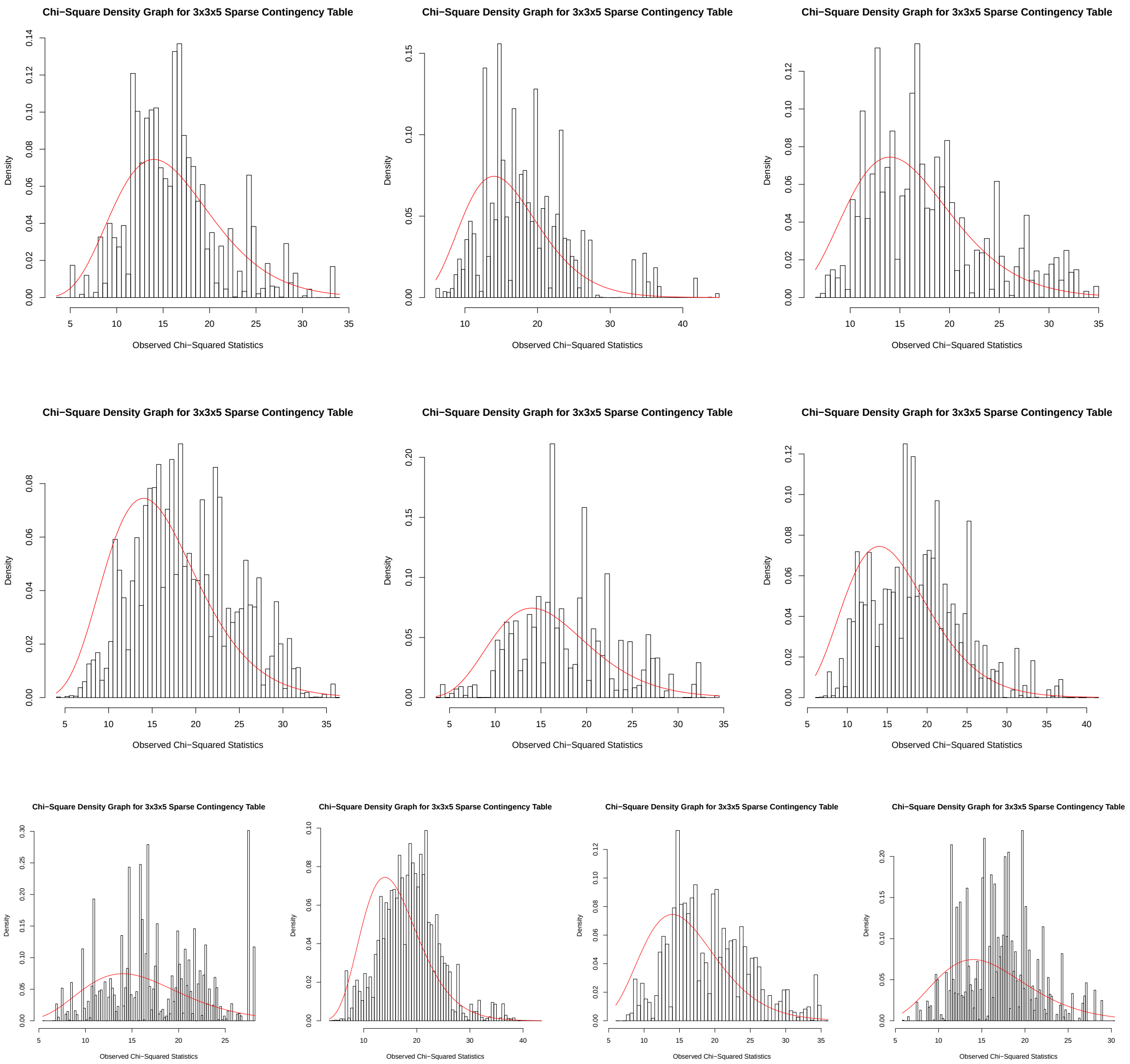


Figure B.10. Histogram of $3 \times 3 \times 5$ Tables' Distribution of Test Statistics for all trials. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.4 $3 \times 3 \times 6$ Sparse Tables

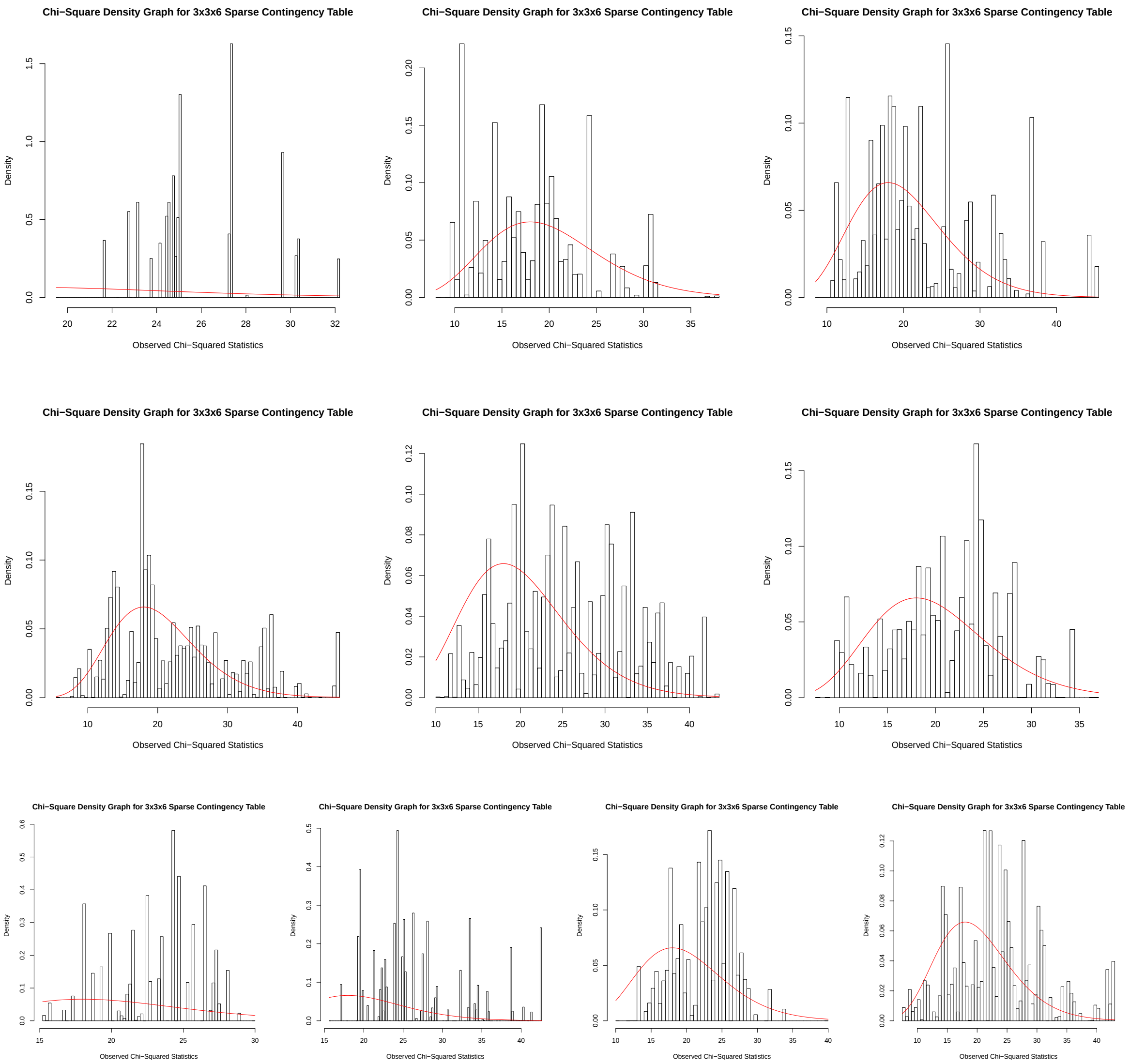
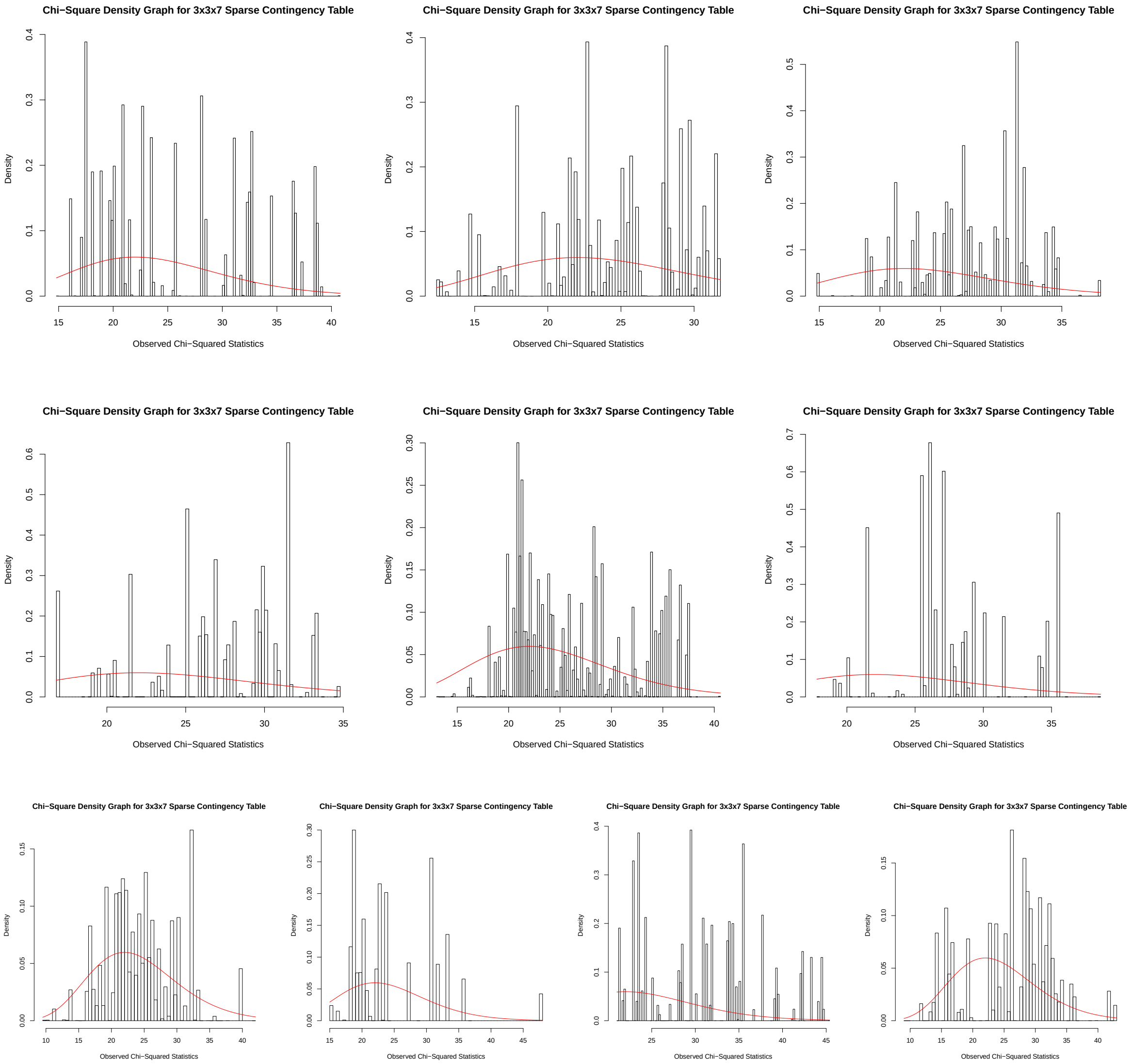


Figure B.11. Histogram of $3 \times 3 \times 6$ Tables' Distribution of Test Statistics for all trials. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.5 $3 \times 3 \times 7$ Sparse Tables


Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Chi−Square Density Graph for 3x3x7 Sparse Contingency Table

Figure B.12. Histogram of $3 \times 3 \times 7$ Tables' Distribution of Test Statistics for all trials. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.6 $4 \times 4 \times 4$ Table

Figure B.13 displays the distribution of χ^2 test statistics for a single trial.

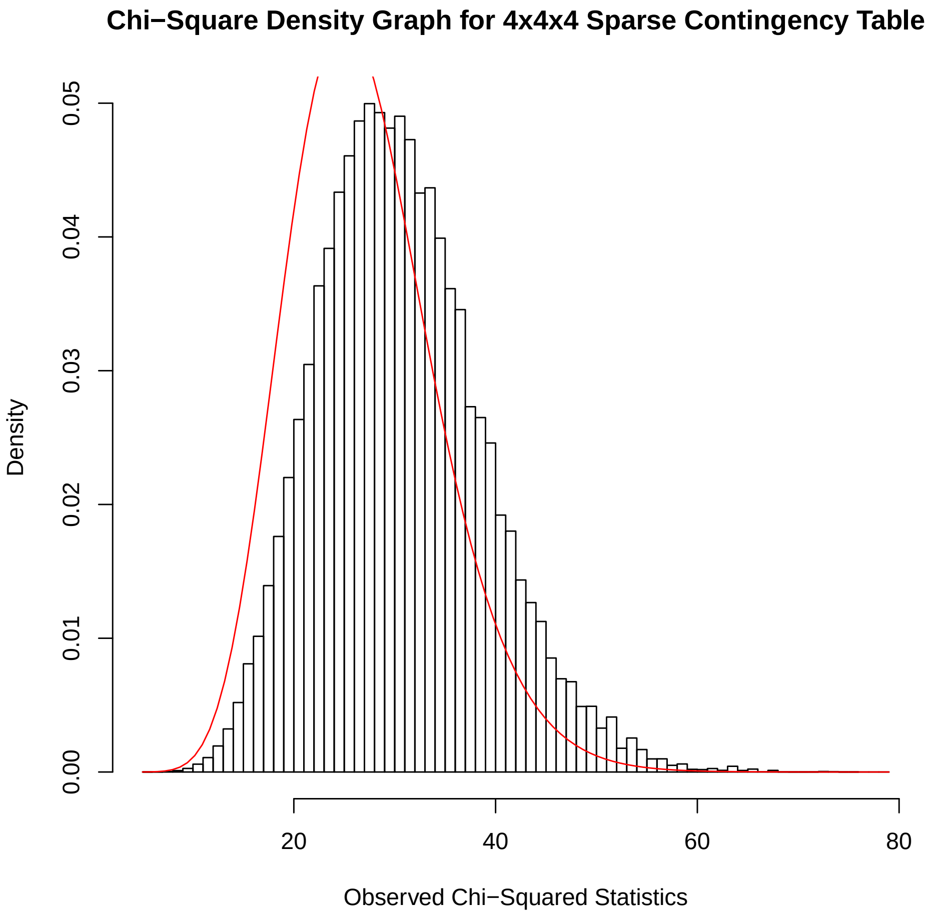


Figure B.13. This figure displays the distribution of the χ^2 Test Statistics for 10 million sampled tables. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

B.2.7 $5 \times 5 \times 5$ Table

Figure B.14 displays the distribution of χ^2 test statistics for a single trial.

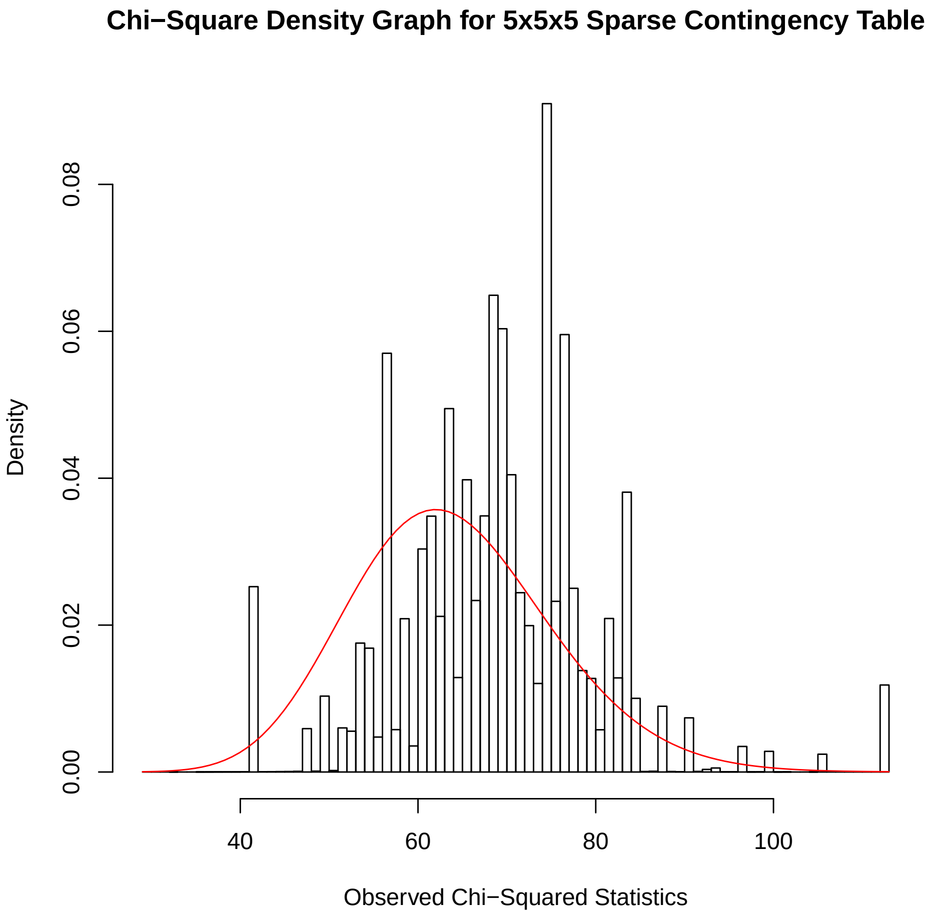


Figure B.14. This figure displays the distribution of the χ^2 Test Statistics for 10 million sampled tables. The red curve indicates the χ^2 distribution. Since the distribution does not converge to the χ^2 distribution, we should not use the χ^2 distribution as the null distribution of test statistics for sparse data.

List of References

[1] A. Agresti, *Categorical Data Analysis*, 2nd ed., ser. Wiley Series in Probability and Statistics. Hoboken, NJ, USA: Wiley-Interscience, 2002.

[2] T. Epps, *Probability and Statistical Theory for Applied Researchers*, ser. World Scientific Books. 27 Warren Street, Suite 401-402, Hackensack, NJ, USA: World Scientific Publishing Co. Pte. Ltd., January 2013, no. 8831. [Online]. Available: https://ideas.repec.org/b/wsi/wsbook/8831.html

[3] P. Diaconis and B. Sturmfels, “Algebraic algorithms for sampling from conditional distributions,” *Ann. Statist.*, no. 1, pp. 363–397, 2002.

[4] F. Bunea and J. Besag, “MCMC in $i \times j \times k$ contingency tables,” National Research Center for Statistics and the Environment, University of Washington, Seattle, WA, USA, Tech. Rep. NRCSE-TRS-037, 1996. [Online]. Available: http://www.nrcse.washington.edu/pdf/trs37_mcmc.pdf

[5] Y. Chen, I. Dinwoodie, and R. Yoshida, *Markov Chains, Quotient Ideals, and Connectivity with Positive Margins*, P. Gibilisco, E. Riccomagno, M. P. Rogantin, and H. P. Wynn, Eds. New York, USA: Cambridge University Press, 2009. [Online]. Available: http://polytopes.net/pdf/pistone6.pdf

[6] R. Scheaffer and L. Young, *Introduction to Probability and Its Applications*, 3rd ed., ser. Advanced series, M. Taylor, D. Seibert, and C. Ronquillo, Eds.

[7] “Lesson 10: Log-linear models,” online class notes for Analysis of Discrete Data., Eberly College of Science, Penn State University, University Park, PA, USA, 2018. [Online]. Available: https://onlinecourses.science.psu.edu/stat504/node/117

[8] M. Drton, B. Sturmfels, and S. Sullivant, *Markov Bases*. Basel, Switzerland: Birkhäuser Verlag, Basel, 2009. [Online]. Available: https://math.berkeley.edu/~bernd/owl.pdf

[9] S. Ross, *Stochastic Processes*, 2nd ed., ser. Wiley Series in Probability and Statistics: Probability and Statistics. New York, USA: John Wiley & Sons, Inc., 1996.

[10] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan, “An introduction to MCMC for machine learning,” *Machine Learning*, vol. 50, no. 1, pp. 5–43, Jan 2003. [Online]. Available: https://link.springer.com/article/10.1023/A:1020281327116

[11] P. Dizikes, "Explained: Monte Carlo simulations," MIT News Office, Cambridge, MA, USA, May 2010. [Online]. Available: http://news.mit.edu/2010/exp-monte-carlo-0517

[12] D. van Ravenzwaaij, P. Cassey, and S. D. Brown, "A simple introduction to Markov chain Monte–Carlo sampling," *Psychonomic Bulletin & Review*, Mar 2016. [Online]. Available: https://doi.org/10.3758/s13423-016-1015-8

[13] P. Diaconis, D. Eisenbud, and B. Sturmfels, *Lattice Walks and Primary Decomposition*, B. E. Sagan and R. P. Stanley, Eds. Boston, MA, USA: Birkhäuser Boston, 1998. [Online]. Available: https://doi.org/10.1007/978-1-4612-4108-9_8

[14] "Lesson 11.1.1: Sparse tables," online class notes for Analysis of Discrete Data., Eberly College of Science, Penn State University, University Park, PA, USA, 2018. [Online]. Available: https://onlinecourses.science.psu.edu/stat504/node/136

[15] Y. Chen, P. Diaconis, S. P. Holmes, and J. S. Liu, "Sequential Monte Carlo methods for statistical analysis of tables," *Journal of the American Statistical Association*, vol. 100, no. 469, pp. 109–120, 2005. [Online]. Available: http://statweb.stanford.edu/~susan/papers/chenJASA05.pdf

[16] The Markov Bases Database, accessed Feb. 1, 2018. [Online]. Available: https://markov-bases.de/show.php?name=no3way-04-04-04

[17] J. De Loera and S. Onn, "Markov bases of three-way tables are arbitrarily complicated," *Journal of Symbolic Computation*, vol. 41, no. 2, pp. 173–181, 2006. [Online]. Available: https://www.math.ucdavis.edu/~deloera/researchsummary/markovarecomplicated.pdf

[18] S. Aoki and A. Takemura, "Minimal basis for a connected Markov chain over $3 \times 3 \times k$ contingency tables with fixed two-dimensional marginals," *Australian & New Zealand Journal of Statistics*, vol. 45, no. 2, pp. 229–249, 2003. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1111/1467-842X.00278/epdf

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California