

University of Memphis

University of Memphis Digital Commons

Electronic Theses and Dissertations

11-22-2016

On Some Exact Nonparametric Conditional Test for Clustered Binary Data

Nana Akwasi Abayie Boateng

Follow this and additional works at: <https://digitalcommons.memphis.edu/etd>

Recommended Citation

Boateng, Nana Akwasi Abayie, "On Some Exact Nonparametric Conditional Test for Clustered Binary Data" (2016). *Electronic Theses and Dissertations*. 1534.
<https://digitalcommons.memphis.edu/etd/1534>

This Dissertation is brought to you for free and open access by University of Memphis Digital Commons. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of University of Memphis Digital Commons. For more information, please contact khhgerty@memphis.edu.

ON SOME EXACT NONPARAMETRIC CONDITIONAL TESTS FOR
CLUSTERED BINARY DATA

by

Nana Akwasi Abayie Boateng

A Dissertation

Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Major: Mathematical Sciences

The University of Memphis

December 2016

Copyright © 2016 Nana Akwasi Abayie Boateng

All rights reserved

ACKNOWLEDGMENTS

I am thankful to my Lord Jesus Christ for granting me the strength to make this work a possibility. I cannot forget about the enormous help I have received from my advisor Dr. E. O. George throughout the period of writing this thesis. I thank him so much for all his advice, suggestions and directions. I am particularly grateful to Dr. Dale Bowman for her tremendous support, guidance and incisive suggestions through out the period of writing this dissertation and all of the time that I have spent here as a student in the Mathematics Department at the University of Memphis. I also would like to say a special thank you to my dissertation committee members Dr. Albert Okunade and Dr. Su Chen for their role in making this whole work a success.

ABSTRACT

Boateng, Nana Akwasi Abayie . Ph.D. The University of Memphis.
December, 2016. On Some Exact Nonparametric Conditional Test for Clustered
Binary Data. Co-Major Professors: E. O. George, Ph.D. and D. Bowman Ph.D.

The development of exact, non-asymptotic, procedures for analyzing clustered discrete data has remained a challenging problem for research statisticians due to the dearth of tractable probability models for describing complex dependence structures of discrete data points within clusters. Under an assumption of de Finetti's [15] definition of infinite exchangeability, several models have been developed. However, these procedures are not valid when data come from intrinsically finite exchangeable population. Moreover, the procedures invariably conduct inference based on asymptotic distribution of estimators or test statistics. For example, for testing association between the rows and columns of a contingency table the asymptotic distribution of the likelihood ratio statistic under null hypothesis of independence is widely used in many applications. Such asymptotic distributions are known to be inaccurate when sample size is moderate or when data are unbalanced or sparse[34]. The goal of this dissertation is to develop and implement exact conditional nonparametric testing procedures for sparse, unbalanced correlated clustered binary data under an assumption of finite exchangeability. The methods proposed in this dissertation takes advantage of the increasing availability of computational power and the development of efficient procedures for enumerating tables, such as the network algorithm [33] and Patefield's algorithm [36]. These methods have facilitated the computation of exact p-value in many cases. We show that for fixed cluster sizes, the distribution of marginal sums of responses can be modeled as a multinomial distribution and that these marginal sums are jointly sufficient statistics [10]. We construct an exact test by conditioning on the sufficient statistics of the multinomial distribution, thereby eliminating all nuisance parameters. By conditioning on the cluster sizes, we

develop an exact nonparametric test for linear a trend. We illustrate the performance of the tests proposed in comparison to those based on unconditional and asymptotic methods by using Monte-Carlo and other stochastic simulations. Our research demonstrates that the exact tests that we have developed can have significantly different p-values, and in many cases superior power, when compared to those using large sample theory approximation when data are sparse or cluster sizes are unbalanced.

TABLE OF CONTENTS

Chapter	Page
Acknowledgments	iii
Abstract	v
List of Tables	viii
1 Introduction	1
2 Literature Review	6
Overview of Nonparametric Models for Categorical Data	6
Unconditional Distributions for Categorical Data	6
Multinomial Distribution	6
Full Multinomial Sampling	6
Product Multinomial Sampling	7
Poisson Sampling	8
Conditional Distributions for Unordered $g \times K$ Contingency Tables	9
Fishers Exact Test	9
Choosing a Test Statistic	12
Singly Ordered $R \times C$ Contingency Tables : Kruskal-Wallis Test	14
Doubly Ordered $R \times C$ Contingency Tables : Jonckheere-Terpstra Test	16
Analysis of $2 \times K$ Tables	18
Cochran-Armitage Trend Test for $2 \times K$ Tables	18
Cochran-Mantel-Haenszel Test for $S \times 2 \times 2$ Tables	20
Generalized Cochran-Mantel-Haenszel Association Statistic for $s \times g \times K$ Tables	21
Generalized Cochran-Mantel-Haenszel Mean Score Statistic for $S \times g \times K$ Tables	23
Generalized Cochran- Mantel-Haenszel Correlation Statistic for $s \times g \times K$ Tables	23
3 Exact Conditional Test for Exchangeable Binary Responses	25
Formulation of Exact Test	25
Reference Set and P-value	28
Monte Carlo Method Using Likelihood Function	29
An Algorithm for Generating Tables	31
Simulation Study I	31
Simulation Study II	38
The p-power and q-power Models	38
Inverse Transform Method	40
Application	47
Summary Statistics for Edge Data	48
Treatment Test of Heterogeneity	51

4	Exact Stratified Linear Rank Test	54
	Stratified Representation of Data	54
	Reference Set	56
	Exact Linear Trend Test	56
	Exact Two-sided P-value Computation	63
	Application using Edge Data	65
	Results	65
	Application using 2,4,5-T Data	67
5	Future Research: Adjusting for Multiplicity in Exact Tests	70
	Introduction	70
	Application with EDGE Data	70
	Conclusion	73
	Appendices	80
A	Computer Programming Codes	81

LIST OF TABLES

Table	Page
1 Fisher's Exact Test for 2×2 Tables	9
2 Fisher's Exact Test for $g \times k$ Tables	11
3 Cochran- Armitage Trend Test For $2 \times K$ Tables	18
4 Cochran- Mantel-Haenszel Test for $S \times 2 \times 2$ Tables	20
5 Summary of Data	26
8 Data from Ethylene glycol ether(EDGE) Laboratory experiment: A Frequency Table Representation	48
9 Summary Statistics for EDGE Data	48
10 Exact Conditional Test	52
11 Comparison with Other Test	52
12 Stratified Representation of Clustered Data	55
13 Stratified Exact Linear Trend Test for 2×2 Tables	57
14 Size of Reference Sets (Mehta, Patel, Senchaudhuri, 1992)	64
15 EDGE Data, Dose groups (0, 25, 50, 100)	66
16 First three columns from Aniko Szabo,2010	66
17 Frequency Distribution of the Number of Implant(s) and Number of Combined Endpoints(t) Following Exposure to 2,4,5-T (Bowman and George 1995)	68
19 Results of T-245 Data	69
20 Raw p-values with equally spaced scores	71
21 Adjusted p-values	72

CHAPTER 1

INTRODUCTION

The validity of statistical inference based on large sample theory is often questionable whenever data are clustered (correlated), sparse and unbalanced or the sample size is small. One of the common mistakes made by practitioners of data science in this era of big data is to assume that big data correspond to a large sample. This assumption often leads to the use of software with algorithms constructed for large sample for inference with multidimensional sparse big data sets that have intrinsically small samples. In dealing with data (big or small) in which data points are discrete and categorical, parametric distributional assumptions are usually difficult to formulate or verify. Exact Nonparametric procedures have long been known to be suitable for inference involving such data.

Significant progress has been made in the development and implementation of exact hypothesis testing in the field of uncorrelated categorical data. Agresti [2] gives an overview of both exact conditional and exact unconditional inference. The same cannot be said for correlated binary data. Very few alternatives exist for making valid statistical inference on clustered binary data with a small sparse or unbalanced samples. Exact methods guarantee that the size of a hypothesis test does not exceed the nominal level and also that the coverage probability for a confidence interval is at least the nominal confidence coefficient [2]. Exact methods have often been criticized for their conservativeness, especially when relevant conditional distribution is highly discrete leading to the conservativeness [2]. One way to address the discreteness is to use the mid-p-value which can smooth out the P – value and consequently reduce the discreteness.

Correlated clustered data occur frequently in biomedical research settings such as teratological experiments, ophthalmologic, otolaryngic and developmental studies. What constitutes a cluster depends on the set up of the experiment. A cluster in a

developmental toxicity study consists of the fetuses of a single dam, while in familial studies a cluster could be a set of family members. The sampling units in a developmental study are the pregnant dams and the observational units are the fetuses of each dam. Since fetuses in a cluster share similar genetic traits and environmental exposure, they tend to exhibit similar individual responses. Failure to account for these intra-litter correlations among fetuses could potentially lead to erroneous inferences. Several approaches for dealing with this problem of intra-cluster correlation have been proposed. Some of the early approaches include the use of quasi-likelihood methods such as generalized estimating equations methods [37], [50], a saturated model [10], Beta-Binomial, Compound-Beta Binomial and Corrected Binomial models [48], [39]. Marginal models such as the GEE were popularized for clustered binary data due to the availability of software and relative computational ease [41]. Expectation maximization (EM) algorithms have also been used to analyze clustered binary data with unequal cluster sizes [43]. Kuk and Pang [51] proposed smoothing methods for unequal cluster sizes under marginal compatibility assumption. Their model accounts for the variability of the estimated null expectation under marginal compatibility. The assumption of marginal compatibility allows estimation to be done over different cluster sizes. When data are sparse, the probability function estimated using a saturated model can be very jagged and some kind of smoothing is needed [51]. Kuk *et al* extended the penalized kernel method to obtain parameter estimates for unequal cluster sizes using an EM-type algorithm.

Luta *et al* [30] proposed an exact conditional logistic regression for correlated binary data that conditions on nuisance parameters and treats the clusters as fixed effects. Their approach however, results in over conditioning as the number of clusters increases [12]. Log-linear models have also been used in analyzing clustered data. Log-linear models may estimate parameters with different standard errors and

different covariances making interpretation of the analysis difficult [8]. Corcoran *et al* [12] proposed an exact linear trend test for correlated binary data based on an exponential model proposed by Molenberghs and Ryan [41]. By conditioning on sufficient statistics, they eliminate the nuisance parameters under the null hypothesis of no treatment effect, leading to a conditional test analogous to Fisher's Exact test. The exponential model from which the conditional test is obtained however, only accounts for pairwise interactions and assumes a linear logistic model for the marginal response probability using the Cochran-Armitage trend statistic [46]. Higher order interactions in the clusters are set to zero.

Difficulty with using many procedures arise when data are sparse, unbalanced or when sample size is small. For example when data are sparse, statistical procedures that involve maximization of the likelihood may suffer convergence problems [2]. Valid statistical inference can be conducted whenever one is presented with a small sample, sparse or unbalanced data by conducting exact inference. The justification for exact conditional inference lies in three main principles [32]:the sufficiency, ancillary and randomization principles. An exact conditional test can be obtained by conditioning on sufficient statistic to eliminate nuisance parameters in the model under the null hypothesis of no association between response and treatment. The data can be represented in a contingency table format with treatment groups as rows (or columns) and response types as columns (or rows). To test for treatment effect using exact methods, a reference set is specified. The reference set contains all tables with marginal sums equal to that of the observed table. The exact $P - value$ can consequently be computed by comparing the observed table with tables in the reference set. If the observed table is unlikely under the null hypothesis when compared to the reference set, the null hypothesis is rejected.

Exact inference has gained increased popularity in recent times due to the availability of computational power and efficient table generation algorithms. Some

of the various table enumeration methods include the network algorithm [33], Fourier transform and Algorithm AS 159 [36]. Mehta and Patel [33], with their introduction of the network algorithm, greatly extended the bounds of computational feasibility for exact inference. The network algorithm implicitly lists all tables in the reference set through a series of nodes and arcs. Each table in the reference set is represented by the sum of lengths of a distinct path in the network. The network representation is used in computing the exact distribution by a stage-wise recursion process through the network path. The speed of computation of $P - values$ is increased by computing at each node, lower and upper bounds on the test statistic value for each table that passes through that node [3].

Explicit enumeration of all tables becomes impractical as the sample size increases. For example, tables with sample sizes 20 and 100, have about 40000 and 7×10^9 tables respectively [2]. For large sample sizes, an alternative is to use Monte-Carlo methods to sample a large number of tables from the reference set and make inference on the sample of tables. This reduces the computational time significantly for large sample sizes.

The remainder of this dissertation is organized as follows. In chapter 2, we provide a review of various nonparametric approaches to modeling categorical data. The Fisher's exact test is among the most popular conditional exact test [18]. Its popularity has been easily spurred on by accessible computer software. In chapter 3, we formally introduce the problem of modeling exchangeable clustered binary data by conditioning on complete sufficient statistics. We discuss the formulation of the problem, computation of the $P - values$ and various algorithms that allow sampling from the reference set to be computationally feasible. The Monte Carlo approach provides a convenient way to overcome the computational challenge of explicitly listing every single table in the reference set. Two simulation studies are performed to examine the performance of the test under several hypothetical cluster

sizes, intra-litter correlations and sample sizes. We also implement the test using real data obtained from the EDGE toxicology experiment.

In chapter 4, we introduce the exact stratified linear rank test for clustered data. We assign non decreasing monotone Weights to the ranked row and column sums for each stratum. By conditioning on the cluster sizes, we order each stratum by a linear trend statistic. The overall test statistic is obtained by summing the individual linear trend test statistics over all the available strata.

In chapter 5, we extend the exact test to the field of multiple testing. Simultaneous testing of multiple hypothesis introduces type-I error probability which approaches one as the number of test increase. The adjusted $P - value$ is used to make inference in multiple testing to control for both the Familywise Error Rate and False Discovery rate. Various approaches to compute the adjusted p-values are discussed including resampling methods to estimate the exact adjusted $P - value$.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview of Nonparametric Models for Categorical Data

2.1.1 Unconditional Distributions for Categorical Data

The exact probability of any categorical data \mathbf{X} , depends on the sampling scheme that is used to generate \mathbf{X} . Three key distributions for modeling categorical responses are the full multinomial, product multinomial and Poisson distributions.

2.1.2 Multinomial Distribution

Suppose that each of n independent, identical trials has an outcome one of K possible categories. Let X_i be the number of times an outcome of category i is observed in n independent trials, then the vector $\underline{\mathbf{X}} = (X_1, X_2, \dots, X_K)$ has multinomial distribution. Let π_i denote the probability of outcome in category i for each trial. The probability mass function of the multinomial distribution is given as:

$$P(X_1 = x_1, X_2 = x_2, \dots, X_K = x_K) = \begin{cases} \frac{n!}{x_1!x_2!\dots x_k!} \prod_{i=1}^K \pi_i^{x_i} & , \text{ if } \sum_{i=1}^K x_i = n, \quad \sum_{i=1}^K \pi_i = 1 \\ 0 & \text{otherwise} \end{cases}$$

The binomial distribution is a special case where $K = 2$. The expected value, covariance and variance for the multinomial distribution is given as: $E(X_i) = n\pi_i$, $cov(X_i, X_j) = -n\pi_i\pi_j$ and $var(X_i) = n\pi_i(1 - \pi_i)$ respectively.

2.1.3 Full Multinomial Sampling

Consider two factors with g and K categories respectively. Suppose n items are sampled independently from the population, classify each X_{ij} as the cell count in the i th row of the j th column, $i = 1, \dots, g$ and $j = 1, \dots, K$. Let the probability of an outcome in the i th category of factor 1 and the j th category of factor 2 be denoted as π_{ij} for $i = 1, \dots, g$ and $j = 1, \dots, K$. In this approach the sample size n is assumed to be fixed, and the cell counts X_{ij} are random.

$(X_{11}, \dots, X_{gK}) \sim Mult(n, \pi_{11}, \dots, \pi_{gK})$. The hypothesis of interest is independence of factor 1 and factor 2, $H_0 : \pi_{ij} = \pi_i \pi_j$. Where π_i is the marginal probability of being in the i th category of factor 1 and π_j is the marginal probability of being in the j th category of factor 2. Binomial sampling is a special case when each of the factors has only two categories.

2.1.4 Product Multinomial Sampling

Product multinomial sampling is generated if for any table \mathbf{X} either the marginal sums of factor 1 are fixed and the margins sums of factor 2 are allowed to vary or vice versa, for $i = 1, \dots, g$ and $j = 1, \dots, K$. Let m_i be the fixed marginal sum of the i th category of factor 1 which is independently sampled. Each cell count X_{ij} is classified into j category of factor 2. Each row of the table with m_i observations, $i = 1, \dots, g$ has a multinomial distribution with parameters, $(X_{i1}, \dots, X_{iK}) \sim Mult(m_i, \pi_{i1}, \dots, \pi_{iK})$. The distribution of any table \mathbf{X} is therefore the product of independent multinomial distributions from the rows.

$$P(\underline{X} = \underline{x}) = P(\underline{\mathbf{x}}) = \prod_{i=1}^g \frac{m_i! \prod_{j=1}^K \pi_{ij}^{x_{ij}}}{\prod_{i=1}^g \prod_{j=1}^K x_{ij}!} \quad (2.1)$$

Independent binomial sampling is a special case of product multinomial sampling where the row and column variables, factors 1 and 2 have two levels. An example of a product multinomial sampling is if each of g different treatments is administered to m_i patients, $i = 1, \dots, g$ and the responses to each treatment level i is recorded in K categories. The hypothesis of interest would be whether the different treatments are equivalent thus whether the probability π_{ij} of treatment i with response j is the same across all treatment levels i , specifically $H_0 : \pi_{ij} = \pi_j$, for $j = 1, 2, \dots, K$.

2.1.5 Poisson Sampling

In Poisson sampling the number trials/sample size is not fixed but random. Each cell count X_{ij} is considered as an independent Poisson random variable, $X_{ij} \sim Poisson(\lambda_{ij})$, where λ_{ij} is the rate of an occurrence of an event in cell (i, j) . The sampling scheme is under the assumption that the data generation process follows the Poisson distribution. Poisson sampling scheme is useful in modeling counts where the probability of success of an event is very small in a very large number of trials. It is usually used in modeling counts over a fixed period of time or space. The distribution of a table \mathbf{X} with independent cell counts X_{ij} , $i = 1, \dots, g$ and $j = 1, \dots, K$ is

$$P(\mathbf{x}) = \prod_{i=1}^g \prod_{j=1}^K \frac{(\lambda_{ij})^{x_{ij}} e^{-\lambda_{ij}}}{x_{ij}!} \quad (2.2)$$

The hypothesis of interest is independence of each cell, $H_0 : \pi_{ij} = \pi_i \cdot \pi_j$. An example of Poisson sampling is a situation in which there are $g \times K$ groups of people, one group for each cell of the $g \times K$ table. The members of each group arrive randomly at a hospital for a medical check up over a period of time. There exists a connection between the Poisson and multinomial distributions. Suppose $X_{ij} \sim Poisson(\lambda_{ij})$, $i = 1, \dots, g$ and $j = 1, \dots, K$. Let

$$n = \sum_i \sum_j X_{ij} \quad \text{then} \quad \sum_i \sum_j X_{ij} \sim Poisson(\lambda) \quad \text{where} \quad \lambda = \sum_i \sum_j \lambda_{ij}$$

$$\begin{aligned} P(X_{11} = x_{11}, \dots, X_{gK} = x_{gK} | \sum_i \sum_j x_{ij} = n) &= \frac{P(X_{11} = x_{11}, \dots, X_{gK} = x_{gK})}{P(\sum_i \sum_j x_{ij} = n)} \\ &= \frac{\prod_{i=1}^g \prod_{j=1}^K \frac{\lambda_{ij}^{x_{ij}} \exp^{-\lambda_{ij}}}{x_{ij}!}}{\frac{\lambda^n \exp^{-\lambda}}{n!}} = \frac{\exp\left(-\sum_{i=1}^g \sum_{j=1}^K \lambda_{ij}\right) \prod_{i=1}^g \prod_{j=1}^K \frac{\lambda_{ij}^{x_{ij}}}{x_{ij}!}}{\frac{\lambda^n \exp^{-\lambda}}{n!}} = \frac{\exp(-\lambda) \prod_{i=1}^g \prod_{j=1}^K \frac{\lambda_{ij}^{x_{ij}}}{x_{ij}!}}{\frac{\lambda^n \exp^{-\lambda}}{n!}} \end{aligned}$$

$$= n! \prod_{i=1}^g \prod_{j=1}^K \frac{1}{x_{ij}!} \left(\frac{\lambda_{ij}}{\lambda} \right)^{x_{ij}}$$

Hence the distribution of Poisson counts conditional on their sum is a multinomial distribution.

$$= \frac{n!}{\prod_{i=1}^g \prod_{j=1}^K x_{ij}} \prod_{i=1}^g \prod_{j=1}^K \pi_{ij}^{x_{ij}} \text{ where } \pi_{ij} = \frac{\lambda_{ij}}{\lambda}$$

2.2 Conditional Distributions for Unordered $g \times K$ Contingency Tables

2.2.1 Fishers Exact Test

Consider N observations of two dichotomous factors, factor 1 and factor 2 with possible combinations given in a 2×2 contingency table 1 below.

Table 1: Fisher's Exact Test for 2×2 Tables

		Factor 1		RowTotal
		1	2	
Factor 2	1	A_{11}	A_{12}	m_1
	2	A_{21}	A_{22}	m_2
Column Total		A_1	A_2	N

For each fixed row, the counts A_{11} and A_{21} are distributed as two independent binomial distributions with parameters, (m_1, π_1) and (m_2, π_2) respectively. Where π_1 and π_2 are the probability of success in row 1 and 2 respectively. Under the null hypothesis of row independence, $H_0 : \pi_1 = \pi_2 = \pi$, the distribution of A_{ij} conditioned on the row and column marginals is the hypergeometric distribution. Let $X = (m_1, m_2, A_1, A_2)$. Then

$$p_X = \frac{\binom{m_1}{A_{11}} \binom{m_2}{A_{21}}}{\binom{N}{A_1}} \quad (2.3)$$

Two common sampling schemes for 2×2 tables are the Binomial and

Multinomial. Binomial sampling is obtained by fixing the row sums. The hypothesis of interest is $H_0 : \pi_1 = \pi_2$ versus $H_a : \pi_1 \neq \pi_2$. In the multinomial case the cell counts $(A_{11}, A_{12}, A_{21}, A_{22})$ are multinomial distributed with a fixed sample size N and probability of cell count in cell (i, j) is $\pi_{ij}, i = 1, 2$ and $j = 1, 2$. The hypothesis of interest in the test of independence of the cells is $H_0 : \pi_{ij} = \pi_{i.}\pi_{.j}$ versus $H_a : \pi_{ij} \neq \pi_{i.}\pi_{.j}, i = 1, 2$ and $j = 1, 2$.

The hypothesis may equivalently be expressed in terms of the odds ratio. $H_0 : OR = 1$ and $H_a : OR \neq 1$. where,

$$\frac{\pi_{11}\pi_{22}}{\pi_{12}\pi_{21}}$$

An odds ratio equal to one indicates independence of the cells.

Let the observed table be x with marginals (A_1, A_2, m_1, m_2) . Define the reference set

$$\Gamma = (Y : Y \text{ is } 2 \times 2; A_1, A_2, m_1, m_2)$$

is defined as the set of all tables with marginal sums equal to the observed. The exact two sided p-value P is defined as :

$$\sum_{Y \in \Gamma^*} P_Y \quad \text{where} \quad \Gamma^* = (Y : Y \in \Gamma \text{ is } 2 \times 2; P_Y) \geq P_X)$$

The exact p-value P is then obtained as the sum of distributions P_Y of all tables in the reference set at least as extreme as the observed table P_X .

The 2×2 Fishers exact test can be easily extended to tables of size $g \times K$. Suppose patients are now assigned to g treatments and response to treatment is recorded in K categories as shown in Table 2. For each fixed row, (A_{i1}, \dots, A_{iK}) , is distributed as a multinomial distribution with parameters $(m_i, \pi_{i1}, \dots, \pi_{iK})$ for $i = 1, \dots, g$. The null hypothesis can be expressed as

$$H_0 : \pi_{1j} = \pi_{2j} = \dots = \pi_{gj} = \pi_j \quad j = 1, \dots, K$$

Under the null hypothesis of independence of rows, the row sums

Table 2: Fisher's Exact Test for $g \times k$ Tables

		Factor 1				RowTotal
		1	2	...	K	
Factor 2	1	A_{11}	A_{12}	...	A_{1K}	m_1
	2	A_{21}	A_{22}	...	A_{2K}	m_2
	3	A_{31}	A_{32}	...	A_{3K}	m_3
	\vdots	\vdots	\vdots		\vdots	\vdots
	g	A_{g1}	A_{g2}	...	A_{gK}	m_g
	Column Total	A_1	A_2	...	A_g	N

$(A_1, \dots, A_g | H_0)$ are also multinomial distributed with parameters
 $(N, \pi_1, \dots, \pi_k | H_0)$

$$P(A_{i1} = a_{i1}, \dots, A_{gK} = a_{gK}, i = 1, \dots, g | H_0; A_1, \dots, A_g) =$$

$$\begin{aligned} & \frac{P(A_{i1} = a_{i1}, \dots, A_{gK} = a_{gK}; i = 1, \dots, g | H_0)}{P(A_1 = a_1, A_2 = a_2, \dots, A_K = a_K)} \\ &= \frac{\prod_{i=1}^g \binom{m_i}{a_{i1}, \dots, a_{iK}} \prod_{j=1}^K \pi_{ij}^{a_{ij}}}{\binom{N}{a_1, \dots, a_K} \prod_{j=1}^K \pi_j^{a_j}} \\ &= \frac{\prod_{i=1}^g \binom{m_i}{a_{i1}, \dots, a_{iK}} \prod_{j=1}^K \pi_j^{\sum_i a_{ij}}}{\binom{N}{a_1, \dots, a_K} \prod_{j=1}^K \pi_j^{a_j}} = \frac{\prod_{i=1}^g \binom{m_i}{a_{i1}, \dots, a_{iK}}}{\binom{N}{a_1, \dots, a_K}} \\ &= \frac{\prod_{i=1}^g \prod_{j=1}^K m_i! a_j}{N! \prod_{i=1}^g \prod_{j=1}^K a_{ij}} \end{aligned}$$

Conditioning on the row and column marginals is a convenient way to eliminate parameters which in this case are the probabilities π_{ij} from the distribution of P_X of

the $g \times K$ table.

The reference set Γ of the $g \times K$ table is defined by restricting the sample space of the observed table X to the set of tables with fixed row and column marginal sums. Specifically the reference set is defined as:

$$\Gamma = \left\{ Y : Y \text{ is } g \times K \quad \sum_{i=0}^g A_{ij} = A_j; \sum_{j=1}^K A_{ij} = m_i; \text{ for all } i, j \right\}$$

Fisher's exact test orders the tables in Γ according to it's hypergeometric distribution P_Y . The test statistic for each $Y \in \Gamma$ is defined as [34]:

$$T_Y = -2 \log(\gamma P_Y)$$

where

$$\gamma = (2\pi)^{\frac{(g-1)(K-1)}{2}} N^{-\frac{(gK-1)}{2}} \prod_{i=1}^g m_i^{\frac{K-1}{2}} \prod_{j=1}^K A_j^{\frac{g-1}{2}}$$

Freeman and Halton(1951) extended the asymptotic distribution of a 2×2 table by Fisher(1925) to $g \times K$. Under the null hypothesis of independence of row and columns, T_Y has a chi-square distribution with $(g - 1)(K - 1)$ degrees of freedom.

The exact p-value is found as the sum of probabilities all tables in the reference set that are at least as extreme as the observed table. Specifically the exact p-value is defined below :

$$\sum_{Y \in \Gamma^*} P_Y \quad \text{where } \Gamma^* = (Y : Y \in \Gamma; P_Y \geq P_X)$$

where P_Y and P_X are the probability distributions of the tables in the reference set and the observed table respectively computed under the null hypothesis.

2.2.2 Choosing a Test Statistic

Several tests are available for $g \times k$ contingency tables with row and column marginals m_i and n_j respectively, $i = 1 \dots, g$ and $j = 1, \dots, k$. If there is no ordering of either the columns or rows, then Fishers exact test, Likelihood Ratio and the Pearson chi-squared test are appropriate and are the most powerful against any other alternative test of the null hypothesis of no row and column interaction.

For discrepancy measure $D(Y)$, large absolute values of $D(Y)$ provide greater evidence against the null hypothesis, while small absolute values are consistent with it. Fishers test orders each table $Y \in \Gamma$ by a hypergeometric distribution $P(Y)$. Discrepancy measures of various tests are given below, where y_{ij} is the count in the i th row and the j th column:

For the Pearson Chi-square test:

$$D(y) = \sum_{i=1}^g \sum_{j=1}^K \frac{(y_{ij} - \frac{m_i n_j}{N})^2}{\frac{m_i n_j}{N}} \sim X_{(g-1)(K-1)}^2 \quad (2.4)$$

The likelihood ratio test orders every table $y \in \Gamma$ according to the likelihood ratio statistic:

$$D(y) = 2 \sum_{i=1}^g \sum_{j=1}^K y_{ij} \log \left(\frac{y_{ij}}{\frac{m_i n_j}{N}} \right) \quad (2.5)$$

The Kullback and Leibler modified log-likelihood ratio statistic

$$D(y) = 2 \sum_{i=1}^g \sum_{j=1}^K \frac{m_i n_j}{N} \log \left(\frac{\frac{m_i n_j}{N}}{y_{ij}} \right) \quad (2.6)$$

The N statistics of Neyman

$$D(y) = \sum_{i=1}^g \sum_{j=1}^K \frac{(y_{ij} - \frac{m_i n_j}{N})^2}{y_{ij}} \quad (2.7)$$

The Freeman-Tukey statistic

$$D(y) = 4 \sum_{i=1}^g \sum_{j=1}^K \left(\sqrt{y_{ij}} - \sqrt{\frac{m_i n_j}{N}} \right)^2 \quad (2.8)$$

The modified Freeman-Tukey statistic

$$D(y) = \sum_{i=1}^g \sum_{j=1}^K \left(\sqrt{y_{ij}} + \sqrt{y_{ij} + 1} - \sqrt{4 \left(\frac{m_i n_j}{N} \right) + 1} \right)^2 \quad (2.9)$$

The Cresie and Read statistic

$$D(y) = \frac{9}{5} \sum_{i=1}^g \sum_{j=1}^K y_{ij} \left[\left(\frac{y_{ij}}{\frac{m_i n_j}{N}} \right)^{\frac{2}{3}} - 1 \right] \quad (2.10)$$

A general representation of most of the test statistics above is given below:

$$D(y)^\delta = \frac{2}{\delta(1+\delta)} \sum_{i=1}^g \sum_{j=1}^K y_{ij} \left[\left(\frac{y_{ij}}{\frac{m_i n_j}{N}} \right)^\delta - 1 \right] \quad (2.11)$$

Where $\delta = 1, 0, -2, -1, -0.5$ and $\frac{2}{3}$ corresponds to the Pearson-Chi-squared statistic, log likelihood ratio statistic, Neyman statistics, Kullback and Leibler, Freeman-Tukey and Cresie -Read respectively.

2.3 Singly Ordered $R \times C$ Contingency Tables : Kruskal-Wallis Test

The Kruskal-Wallis (KW) test is a rank based method to compare K independent samples. Kruskal and Wallis(1952) list several advantages of rank based methods. Specifically, they simplify calculations and only general assumptions are made about the distribution of observations (independence). The null hypothesis that the K independent samples originate from the same distribution is tested against the alternate that at least one sample is stochastically larger than another. It's the nonparametric analogue of the parametric one-way analysis of variance (ANOVA) test. In KW test, the population mean of ranks is compared instead of comparing population means in ANOVA. The Mann-Whitney test is a special case of Kruskal-Wallis test for two samples. $H_0 : F_1 = F_2 = \dots = F_k$ vs $H_1 : F_i < F_j$ for some $i \neq j$

Where F_i is the distribution of sample i . Under the null hypothesis, the observations are a combined sample of size N from the common population. The combined sample is ranked from $1, \dots, N$. The total sum of ranks is $\sum_i^N i = \frac{N(N+1)}{2}$. The expected proportion of the i th sample i, \dots, K in the combined sample is $\frac{n_i}{N} \frac{N(N+1)}{2} = \frac{n_i(N+1)}{2}$ under the null hypothesis.

A test statistic \mathbf{t} based on a function of deviations between the observed and expected rank sum is given as $\mathbf{t} = \sum_{i=1}^K \left[r_i - \frac{n_i(N+1)}{2} \right]^2$, where r_i is the sum of ranks assigned to the elements of the i th sample. The null hypothesis of equal distribution of the K samples is rejected for large values of \mathbf{t} . The number of ways to assign N observations into K columns of size n_i , $i = 1, \dots, k$ is $\frac{N!}{\prod_{i=1}^K n_i!}$. Each $\frac{N!}{\prod_{i=1}^K n_i!}$ possible table assignment is evaluated and \mathbf{t} is calculated for each table. The distribution of the test statistic is then obtained as $f_{\mathbf{t}}(\mathbf{t}) = \mathbf{t} \frac{N!}{\prod_{i=1}^K n_i!}$

Kruskal and Wallis (1952) proposed a modified test statistic H that is weighted sum of square deviations that uses the the reciprocals of the sample sizes n_i as weights.

$$H = \frac{12}{N(N+1)} \sum_{i=1}^K \frac{1}{n_i} \left[r_i - \frac{n_i(N+1)}{2} \right]^2 \quad (2.12)$$

An equivalent computationally preferred version of H can be expressed as:

$$H = \frac{12}{N(N+1)} \sum_{i=1}^K \frac{r_i^2}{n_i} - 3(N+1) \quad (2.13)$$

When the sample sizes n_i are the same, the statistic \mathbf{t} is equivalent to H [20]. Tables exist for exact probabilities of \mathbf{t} and H for small sample sizes and the number of columns K . Due to the large number of computations to carry out to obtain these exact probabilities, large sample approximations have been developed for relatively large samples.

$$H^* = \sum_{i=1}^K \frac{12n_i \left[r_i - \frac{(N+1)}{2} \right]^2}{N(N+1)} \quad (2.14)$$

The statistic H^* is approximately distributed chi squared with $(K-1)$ degrees of freedom (Detailed proof can be found in [20]). The null hypothesis H_0 is rejected in favor of the alternative if $H^* \geq X_{(\alpha, K-1)}^2$. A correction for ties i uses the midrank

method. A modified test statistic for correcting for the number of ties present in the population, t is:

$$H' = \frac{H}{1 - \frac{\sum t(t^2-1)}{N(N^2-1)}} \quad (2.15)$$

The correction statistic for ties H' , does not significantly alter H for relatively small number of ties. When the null hypothesis is rejected, multiple pairwise comparisons could be made for any two groups say i and j ($1 \leq i \leq j \leq K$) by the statistic:

$$Z_{ij} = \frac{|\bar{r}_i - \bar{r}_j|}{\sqrt{\frac{N(N+1)}{2} \left[\frac{1}{n_i} + \frac{1}{n_j} \right]}} \quad (2.16)$$

The critical region is $Z_{ij} \leq Z_{\frac{\alpha}{K(K-1)}}$, where z is the standard normal quantile. The Type 1 error rate α is corrected by dividing α by the number of pairwise comparisons $\binom{K}{2}$. This kind of Type-I correction is the conservative Bonferroni method. The null hypothesis that the two populations are the same is rejected if $P(Z_{ij} \leq Z_{\frac{\alpha}{K(K-1)}}) \leq \frac{\alpha}{k(k-1)}$. The Wilcoxon test is a special case of the Kruskal-Wallis test for the location parameter of two independent samples.

2.4 Doubly Ordered $R \times C$ Contingency Tables : Jonckheere-Terpstra Test

The Jonckheere-Terpstra test is a nonparametric test for ordered alternative within independent samples. The null hypothesis tests for homogeneity among independent samples. It exhibits greater statistical power over comparable tests like the Kruskal-Wallis test if the samples are ordered from the population from which they are drawn. The Jonckheere-Terpstra Test hypothesis can be states as:

$$H_0 : \theta_1 = \theta_2 = \dots = \theta_k$$

$$H_a : \theta_1 \leq \theta_2 \leq \dots \leq \theta_k, i = 1, \dots, K, (\text{at least one strict inequality})$$

Where θ_i is the median of the i th population. An alternative expression of the alternate hypothesis is $\binom{K}{2}$ multiple comparisons of $\theta_i \leq \theta_{i+1}$ with at least one

$\theta_i \leq \theta_{i+1}$. This then reduces the problem of multiple comparison to a two sample comparison problem. The statistic for combining these multiple two sample comparisons into a single final statistic is the Mann-Whitney statistic U_{ij} , $i, j = 1, 2, \dots, K$ with $i \leq j$. The addition of all U_{ij} results in the statistic B . The Mann-Whitney statistic U_{ij} is defined for any two samples X_i and X_j with sample sizes n_i and n_j respectively as:

$$U_{ij} = \begin{cases} 1 & \text{if } x_{js} < x_{ir} \text{ for } s = 1, \dots, n_j, r = 1, \dots, n_i \\ 0 & \text{if } x_{js} > x_{ir} \end{cases}$$

$$\begin{aligned} B &= U_{12} + U_{13} + \dots + U_{1k} + U_{23} + \dots + U_{2k} + \dots + U_{(k-1)k} \\ &= \sum_{1 < i < j \leq K} U_{ij} = \sum_{i=1}^{K-1} \sum_{j=i+1}^K \sum_{r=1}^{n_j} \mathbf{I}(x_{ir} < x_{js}), \quad \text{where } \mathbf{I} \text{ is the indicator function.} \end{aligned}$$

where x_{ir} is the r th observation in the i th sample and x_{js} is the s th observation in the j th sample. The rejection region is $B \geq B(\alpha, K, n_1, n_2, \dots, n_k)$, where α is the nominal significance level. The null hypothesis is rejected if $P[B \geq B(\alpha, K, n_1, n_2, \dots, n_k)] \leq \alpha$. Tables of exact probabilities for small sample sizes and small populations is available. For sufficiently large sample sizes and large populations, the computation of exact probabilities becomes computationally challenging. An approximate large sample distribution for B is the chi-square distribution with one degree of freedom.

$$\frac{[B - E(B)]^2}{\text{var}(B)} \sim \chi_{(1)}^2 \quad (2.17)$$

where

$$E(B) = \sum_{1 < i < j < k} \sum_{r=1}^{n_j} \frac{n_i n_j}{2} = \frac{N^2 - \sum_{r=1}^k n_r}{2} \quad \text{and} \quad \text{Var}(B) = \frac{N^2(2N + 3) - \sum_{i=1}^k n_i(2n_i + 3)}{72}$$

The asymptotic approximation test rejects the null hypothesis if

$$P(\chi_{(1)}^2 \geq \chi_{(\alpha,1)}^2) \leq \alpha$$

The Mann-Whitney statistic U_{ij} used in computation of the Jonckheere-Terpstra test is modified for tied observations as $U_{ij}^* = \sum_{r=1}^{n_i} \sum_{s=1}^{n_j} D_{rs}$ where

$$D_{rs} = \begin{cases} 1 & \text{if } x_{ir} < x_{js} \\ \frac{1}{2} & \text{if } x_{ir} = x_{js} \\ 0 & \text{if } x_{ir} > x_{js} \end{cases}$$

The modified Jonckheere-Terpstra test statistic $B^* = \sum_{i=1}^{K-1} \sum_{j=1}^K U_{ij}^*$

2.5 Analysis of $2 \times K$ Tables

2.5.1 Cochran-Armitage Trend Test for $2 \times K$ Tables

Consider N subjects who are each exposed to some treatment levels 0 and 1. Assume the responses to treatment is progressively increasing from 1 to K . Let p_j be the probability of response of a subject exposed to treatment level 1 with response level j , $j = 1, 2, \dots, K$. In table 3, X_{ij} is an observation in the i th row of

Table 3: Cochran- Armitage Trend Test For $2 \times K$ Tables

		Treatment	Response					
		Row score	1	2	3	...	K	RowTotal
Treatment	Row 1	1	x_{11}	x_{12}	x_{13}	...	n_{1k}	m_1
	Row 2	0	x_{21}	x_{22}	x_{23}	...	x_{2k}	m_2
	Column Total		n_1	n_2	n_3	...	n_k	N
	Column Score		w_1	w_2	w_3	...	w_K	

the j th category, $i = 1, 2$ and $j = 1, \dots, K$, $w_1 < w_2 < \dots < w_k$ are column scores assigned to response categories $1, \dots, K$ and 0, 1 are the row scores of row 1 and 2 respectively. For $2 \times K$ tables with ordered columns and K independent $bin(n_i, p_i)$ variates, Armitage [4] proposed a linear trend test for testing association between

variable 1 (Treatment) with 2 categories and variable 2 (Response) with K categories. The proportion of observations in any category j is given by $p_j = \frac{x_{1j}}{n_j}$, whereas the overall proportion in row 1 is given as $p = \frac{m_1}{N}$. Each observation on row one is assigned a row score of 1 and each observation on row 2 is assigned a row score of 0.

The null hypothesis of the Cochran-Armitage trend test is that all of K independent binomial proportions are equal with response probability p :

$$H_0 : p_1 = p_2 = \dots = p_k = p \text{ where } p_j = \frac{x_{1j}}{n_j}$$

against an ordered alternative of the form

$$H_a : p_1 \leq p_2 \leq \dots \leq p_k$$

The test is equivalent to the null hypothesis $H_0 : \beta = 0$ for the linear probability regression model $p_i = \alpha + \beta w_i + \epsilon_i$.

The Pearson test statistic for the K category of responses X_K can be expressed in terms of variation among the K proportions by

$$\chi_K^2 = \frac{1}{p(1-p)} \sum_i^K n_i (p_i - p)^2.$$

The chi-squared statistic can be decomposed into two chi-squared statistic to test for the lack of goodness of fit of the model and the existence of a linear test [1].

$$\chi_K^2 = Z^2 + X_L^2$$

where $X_K^2 = \frac{1}{p(1-p)} \sum_i^K n_i (p_i - \hat{p})^2$ has asymptotic chi-squared distribution with $K - 2$ degrees of freedom. It test's the goodness of fit of the model.

$$Z^2 = \frac{b^2}{p(1-p)} \sum_i^K n_i (w_i - \bar{w})^2 = \left[\frac{\sum_i^K (w_i - \bar{w}) x_{1i}}{\sqrt{p(1-p)} \sum_i^K n_i (w_i - \bar{w})} \right]^2. \quad (2.18)$$

Z^2 has an asymptotic chi-square distribution with one degree of freedom and is used to test $H_0 : \beta = 0$ for the linear trend in the proportions.

2.5.2 Cochran-Mantel-Haenszel Test for $S \times 2 \times 2$ Tables

The Cochran-Mantel-Haenszel Test is non-model based approach to testing conditional independence. Data can be stratified to control for possible confounding variables. The Cochran-Mantel-Haenszel test provides a measure of association which is a summary of the weighted average of the risk or odds ratio across the different strata. The null hypothesis tests homogeneity of association between treatment and response groups across the S strata. The test provides a summary estimate of the exposure effect stratified by multiple studies. The odds ratio obtained from each of the stratified subgroups represents exposure effect in the group when the overall joint effect of the stratification variable has been held constant. The overall odds estimate across all the strata can be combined to form a summary estimate adjusted for effects of those variables used in the stratification. The Mantel-Haenszel test statistic $OR_{MH}(OddsRatio)$ or $RR_{MH}(RiskRatio)$ can be considered as weighted averages of odds ratios of each stratum in the data provided b_i and c_i are greater than 0, for b_i, c_i, a_i, d_i as in the table below

For subgroups $i = 1, \dots, S$. The stratum i is given below

Table 4: Cochran- Mantel-Haenszel Test for $S \times 2 \times 2$ Tables

Cases	Outcome		Row Total
Exposure	positive	negative	
Yes	a_i	b_i	m_{1i}
No	c_i	d_i	m_{2i}
Column Total	n_{1i}	n_{2i}	n_i

The test statistics are given by:

$$RR_{MH} = \frac{\sum_{i=1}^S \frac{a_i(c_i+d_i)}{n_i}}{\sum_{i=1}^S \frac{c_i(a_i+b_i)}{n_i}}. \quad (2.19)$$

$$OR_{MH} = \frac{\sum_{i=1}^S \frac{a_i d_i}{n_i}}{\sum_{i=1}^S \frac{b_i c_i}{n_i}}. \quad (2.20)$$

A major disadvantage of stratified analysis is the inability to control simultaneously for multiple confounding variables. The sampling schemes for cell counts (a_i, b_i, c_i, d_i) given row and column marginals $(n_{1i}, n_{2i}, m_{1i}, m_{2i})$ is the hypergeometric distribution. The mean and variance are given below

$$\mu_i = E(a_i) = \frac{m_{1i} n_{1i}}{N_i}. \quad (2.21)$$

$$Var(a_i) = \frac{m_{1i} m_{2i} n_{1i} n_{2i}}{N_i^2 (N_i - 1)}. \quad (2.22)$$

Cell counts from different strata are assumed to be independent. The test statistic for combining information from the S strata is obtained by comparing a_i to its expected value.

$$Q = \frac{\left[\sum_i (a_i - \mu_i) \right]^2}{\sum_i Var(a_i)} \quad \widetilde{\text{asympt}} \quad \chi_{(1)}^2 \quad (2.23)$$

The test gives similar results to a logit model with sufficiently large strata size n_i .

The set back in a parametric model such as the logit model is that, the maximum likelihood estimates of log odds ratio may be over estimated for sparse data [1].

Logit models may also fail to converge.

2.5.3 Generalized Cochran-Mantel-Haenszel Association Statistic for $s \times g \times K$ Tables

Let N_{ijh} , $i = 1, \dots, g, j = 1, \dots, K$ be the cell counts in the h th stratum of a $g \times K$ table, where i represents the levels of factor of interest such as treatment and

j represents another factor such as levels of response. Conditional on row and column marginals each stratum has $(g - 1) \times (K - 1)$ non-redundant cell counts [1].

For each stratum h the vector of cell counts

$n_h = (n_{11h}, \dots, n_{1(k-1)h}, n_{11h}, \dots, n_{1(k-1)h}, \dots, n_{(g-1)1h}, \dots, n_{(g-1)(k-1)h})$ has probability function (2.24) .

$$\frac{\prod_{i=1}^g n_{i..h}! \prod_{j=1}^K n_{.jh}!}{n_{..h}! \prod_{i=1}^g \prod_{j=1}^K n_{ijh}}. \quad (2.24)$$

Let $\mu_h = E(n_h)$ under H_0 : no row by column association. Then

$$\mu_h = \frac{n_h}{n_{..h}} = \frac{(n_{11h}, \dots, n_{1(k-1)h}, n_{11h}, \dots, n_{1(k-1)h}, \dots, n_{(g-1)1h}, \dots, n_{(g-1)(k-1)h})}{n_{..h}} \quad (2.25)$$

Let V_h be the covariance matrix of n_h where

$$Cov(n_{ijh}, n_{i'j'}) = \frac{n_{i.k}(\delta_{ii'}n_{..h} - n_{i'.h})n_{.jh}(\delta_{jj'}n_{..k} - n_{.j'h})}{n_{..h}^2(n_{..h} - 1)} \quad (2.26)$$

where

$$\delta_{ab} = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{otherwise} \end{cases}$$

Let $G_h = n_h - E(n) = n_h - \mu_h$. Now $G = \sum_h G_h = \sum_h (n_h - \mu_h)$ is the aggregation over all $(g - 1)(k - 1)$ strata. Let the covariance matrix of $V = \sum_h V_h$. The general association test statistic for the Cochran-Mantel-Haenszel(CMH) test under the null hypothesis is $Q_G = G^T V G$. The distribution of the test statistic approaches

$X_{(g-1)(K-1)}^2$ as the total sample size ($n = \sum_h n_{..h}$) approaches infinity. When the CMH statistic is significant at some level α , then there exists some association

between the row and column variables in at least one stratum.

2.5.4 Generalized Cochran-Mantel-Haenszel Mean Score Statistic for $S \times g \times K$ Tables

Consider a set of S independent $g \times K$ tables where cell counts in the h th stratum is given by $n_{ijh}, i = 1, \dots, g - 1$ and $j = 1, \dots, K$. Assume the column variable is ordinal and has assigned scores w_{1h}, \dots, w_{Kh} . The null hypothesis of no association between row and column variables in any of the S strata is tested against the alternative that there is a difference in the g mean scores on average across the S strata. Define $M_h = \sum_{j=1}^K w_{jh}(n_h - E(n_h))$, the vector of differences between the observed and expected mean scores under the null hypothesis. Let $M = \sum_{h=1}^S M_h$ which has expectation zero and covariance matrix $V_M = \sum_{h=1}^S V_h$. Then $Q_M = M'V_M^{-1}M$, is the CMH mean score statistic which is approximately $X_{(g-1)}^2$ if the null hypothesis is true. The statistic is used only when the column variable is ordinal or when the variable is an interval column variable. Special cases of the CMH mean statistic arise when $S = 1$ and $g = 2$, Q_M is the Wilcoxon-Mann-Whitney statistic. If $S = 1$ and $r > 2$, Q_M is the Kruskal-Wallis statistic. If $S > 1$ and $n_{hi} = 1$ for $i = 0, \dots, g$ and $h = 1, \dots, S$, then the Q_M is the Friedman's chi-square statistic.

2.5.5 Generalized Cochran- Mantel-Haenszel Correlation Statistic for $s \times g \times K$ Tables

Consider a set of S independent $g \times K$ tables with ordinal row and column variables. Let the scores assigned to the row and column be u_{1h}, \dots, u_{gh} and w_{1h}, \dots, w_{Kh} respectively. The null hypothesis that there is no association between row and column variables in any of the S strata is tested against the alternative that there is consistent positive or negative association between the rows and column scores across all stratum. Define $C_h = \sum_i \sum_j u_{ih}w_{jh}(n_h - E(n_h))$ as the vector of difference between the observed and expected association scores in the h th

stratum under the null hypothesis. Let $C = \sum_h C_h$, then C has expectation zero and variance denoted by V_C . The CMH correlation statistic $Q_C = C'V_C C$ has asymptotic distribution as χ_1^2 for sufficiently large total sample size ($n_{..} = \sum_h n_{..h}$). The Pearson correlation coefficient ρ between row and column scores is a special case of of CMH correlation statistic $Q_c = (n_{..} - 1)\rho^2$ where $S = 1$.

CHAPTER 3

EXACT CONDITIONAL TEST FOR EXCHANGEABLE BINARY RESPONSES

3.1 Formulation of Exact Test

A finite sequence of random variables (X_1, \dots, X_k) is exchangeable if for any k

$$P(X_{\pi(1)} = x_1, \dots, X_{\pi(k)} = x_k) = P(X_1 = x_1, \dots, X_k = x_k)$$

for any permutation $\pi(1) \dots \pi(k)$ of indices $1, \dots, k$. Let (X_1, \dots, X_n) be exchangeable random variables with $\lambda_{0,n} = 1$ and $\lambda_{k,n} = P\{X_1, \dots, X_k = 1\}$, then

$$P\{X_1 = x_1, \dots, X_n = x_n\} = \sum_{k=0}^{n-r} (-1)^k \binom{n-r}{k} \lambda_{r+k,n}$$

where

$$r = \sum_{i=1}^n x_i$$

In this thesis we consider an experiment involving g treatment groups and exchangeable binary responses. For the i th treatment group, let $(X_{ij1}, \dots, X_{ijn_{ij}})$ be the set of binary data from the j th cluster of the i th treatment group, where $j = 1, \dots, m^{(i)}, i = 1, \dots, g$, and let

$$P_{r_{ij}, n_{ij}}^{(i)} = \sum_{k=0}^{n_{ij}-r_{ij}} (-1)^k \binom{n_{ij}-r_{ij}}{k} \lambda_{r_{ij}+k, n_{ij}}$$

$$\text{where } \sum_{k=1}^{n_{ij}} x_{ijk} = r_{ij}$$

If $A_{r,n}^{(i)}$ represents the number of clusters of size n with r responses in the i th treatment group

$$P\{X_{ij1} = x_{ij1}, \dots, X_{ijn_{ij}} = x_{ijn_{ij}}, \quad j = 1, \dots, m^{(i)}, \quad i = 1, \dots, g\} =$$

$$\prod_{i=1}^g \prod_{n=1}^K \prod_{r=0}^n [P_{r,n}^{(i)}]^{A_{r,n}^{(i)}} \quad (3.1)$$

where K denotes the maximum possible cluster size in the experiment

It is clear from (3.1) and the factorization theorem that $A_{r,n}^{(i)}$'s form a set of joint sufficient statistic for $P_{r,n}^{(i)}$'s. Thus we can summarize the dataset as given in Table 5 with each fixed cluster size.

Table 5: Summary of Data

Treatment	Response	Cluster size					Row Total
		1	2	3	...	K	
Control	0	$A_{0,1}^{(1)}$	$A_{0,2}^{(1)}$	$A_{0,3}^{(1)}$...	0	n_0^1
	1	0	$A_{1,2}^{(1)}$	$A_{1,3}^{(1)}$...	$A_{1,K}^{(1)}$	n_1^1
	2	0	0	$A_{2,3}^{(1)}$...	$A_{2,K}^{(1)}$	n_2^1
	⋮	⋮	⋮	⋮	⋮	⋮	
	n	0	0	0	...	$A_{n,K}^{(1)}$	n_n^1
Column Total		$m_1^{(1)}$	$m_2^{(1)}$	$m_3^{(1)}$...	$m_K^{(1)}$	$m^{(1)}$
Treatment 1	0	$A_{0,1}^{(2)}$	$A_{0,2}^{(2)}$	$A_{0,3}^{(2)}$...	$A_{0,K}^{(2)}$	n_0^2
	1	$A_{1,1}^{(2)}$	$A_{1,2}^{(2)}$	$A_{1,3}^{(2)}$...	$A_{1,K}^{(2)}$	n_1^2
	2	0	$A_{2,2}^{(2)}$	$A_{2,3}^{(2)}$...	$A_{K,3}^{(2)}$	n_2^2
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	n	0	0	0	...	$A_{n,K}^{(2)}$	n_n^2
Column Total		$m_1^{(2)}$	$m_2^{(2)}$	$m_3^{(2)}$...	$m_K^{(2)}$	$m^{(2)}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	
Treatment g	0	$A_{0,1}^{(g)}$	$A_{0,2}^{(g)}$	$A_{0,3}^{(g)}$...	$A_{0,K}^{(g)}$	n_0^g
	1	$A_{1,1}^{(g)}$	$A_{1,2}^{(g)}$	$A_{1,3}^{(g)}$...	$A_{1,K}^{(g)}$	n_1^g
	⋮	⋮	⋮	⋮	⋮	⋮	
	n	0	0	0	...	$A_{n,K}^{(g)}$	n_n^g
Column Total		$m_1^{(g)}$	$m_2^{(g)}$	$m_3^{(g)}$...	$m_K^{(g)}$	$m^{(g)}$

where

$$\sum_{n=1}^K A_{r,n}^i = n_r^i \quad \text{and} \quad \sum_{r=0}^n A_{r,n}^i = m_n^i$$

$$\sum_{n=1}^K A_{r,n}^i = n_r^i \quad \text{and} \quad \sum_{r=0}^n A_{r,n}^i = m_n^i$$

Now for each fixed $n, n = 1, \dots, K, \{A_{r,n}^{(i)}, r = 0, \dots, n, i = 0, \dots, g\} \sim$

$$\text{Multinomial} \left(m_n^{(i)}, P_{0,n}^{(i)}, \binom{n}{1} P_{1,n}^{(i)}, \dots, \binom{n}{n-1} P_{n-1,n}^{(i)}, P_{n,n}^{(i)} \right)$$

Now consider testing the hypotheses: H_0 : No treatment effect vs H_a : There is a treatment effect and let T be the test statistic for testing these hypothesis and suppose that we reject H_0 if $T \in C$, where C is the critical region.. Then under $H_0, P_{r,n}^{(i)} = P_{r,n}, i = 1, \dots, g, r = 0, \dots, n, n = 1, \dots, K$ and $A_{r,n} = \sum_{i=1}^g A_{r,n}^{(i)}$, where $\{A_{r,n}, r = 0, 1, \dots, n\}$ form a set of joint complete sufficient statistics for $(P_{0,n}, \dots, P_{n,n})$.

Thus the null hypothesis of no treatment effect reduces to

$$H_0 : P_{r,n}^{(i)} = P_{r,n}, i = 1, \dots, g, r = 0, \dots, n, n = 1, \dots, K.$$

Under $H_0, \left\{ \sum_{i=1}^g A_{r,n}^{(i)} = a_{r,n} | H_0, r = 0, \dots, n, i = 1, \dots, g, n = 1, \dots, K \right\}$ has a Multinomial $(M_n, P_{0,n}, \binom{n}{1} P_{1,n}, P_{n,n} \dots, \binom{n}{r} P_{r,n} \dots P_{n,n})$, and

$$\frac{P \left(A_{0,n}^{(i)} = a_{0,n}^{(i)}, \dots, A_{n,n}^{(i)} = a_{n,n}^{(i)} | H_0, A_{0,n} = a_{0,n}, \dots, A_{n,n} = a_{n,n}, n = 1, \dots, K, i = 0, \dots, g \right)}{P \left(A_{0,n}^{(i)} = a_{0,n}^{(i)}, \dots, A_{K,K}^{(i)} = a_{K,K}^{(i)}, A_{0,n} = a_{0,n}, \dots, A_{K,K} = a_{K,K}, n = 1, \dots, K, i = 0, \dots, g \right)}$$

$$= \begin{cases} \frac{\prod_{i=1}^g \prod_{n=1}^K \left(P \left(A_{0,n}^{(i)} = a_{0,n}^{(i)}, \dots, A_{n,n}^{(i)} = a_{n,n}^{(i)}, A_{0,n} = a_{0,n}, \dots, A_{n,n} = a_{n,n} \right) \right)}{\prod_{n=1}^K P(A_{0,n} = a_{0,n}, \dots, A_{n,n} = a_{n,n})} & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} = a_{r,n} \\ 0 & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} \neq a_{r,n} \end{cases}$$

$$= \begin{cases} \frac{\prod_{i=1}^g \prod_{n=1}^K \binom{m_n^{(i)}}{a_{0,n}^{(i)}, \dots, a_{n,n}^{(i)}} \prod_{r=0}^K P_{r,n}^{\sum_{i=1}^g A_{r,n}^{(i)}}}{\prod_{n=1}^K \binom{M_n}{a_{0,n}, \dots, a_{n,n}} \prod_{r=0}^K P_{r,n}^{\sum_{i=1}^g A_{r,n}^{(i)}}} & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} = a_{r,n}, r = 0, \dots, n \\ 0 & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} \neq a_{r,n} \end{cases}$$

$$= \begin{cases} \frac{\prod_{i=1}^g \prod_{n=1}^K \binom{m_n^{(i)}}{a_{0,n}^{(i)}, \dots, a_{n,n}^{(i)}}}{\prod_{n=1}^K \binom{M_n}{a_{0,n}, \dots, a_{n,n}}} & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} = a_{r,n}, r = 0, \dots, n \\ 0 & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} \neq a_{r,n} \end{cases}$$

3.2 Reference Set and P-value

Define a reference set Γ as the set of all possible contingency tables with fixed marginal column and row sums as the observed table sums for treatment groups $i = 1, \dots, g$, cluster sizes $n = 1, \dots, K$, with responses $r = 0, \dots, n$, that is

$$\Gamma = \left\{ Y : Y \text{ is } n \times K; \sum_{n=1}^K A_{r,n}^i = n_r^i, \sum_{r=0}^n A_{r,n}^i = m_n^i \right\}$$

Then for an observed value t of a test statistic T , the P -value of a test which rejects H_0 , for large value of T is given by

$$P\text{-value}=p(T > t_{obs}|H_0) = \sum_{T>t_{obs}} \left[\frac{\prod_{i=1}^g \prod_{n=1}^K \binom{m_n^{(i)}}{a_{0,n}^{(i)}, \dots, a_{n,n}^{(i)}}}{\prod_{n=1}^K \binom{M_n}{a_{0,n}, \dots, a_{n,n}}} \right]$$

3.3 Monte Carlo Method Using Likelihood Function

Monte Carlo methods, provide powerful tools to compute unbiased estimates of the exact $P - value$ for small data sets , sparse ,unbalanced tables, non normal data and when the data that does not to conform to any particular parametric distribution. For any observed table, a large number of tables with the same column and marginal sums are randomly sampled from the reference set to obtain a set of $P - values$. The enumeration of all possible tables in a reference set can be computationally expensive for large tables. The accuracy of the Monte Carlo estimate of the exact $P - value$ increases as the number of tables sampled from the reference set increases.

The procedure for Monte-Carlo computation may be summarized as follows

- Let the critical region of the reference set be defined as:

$$\Gamma^* = \{Y : Y \in \Gamma \text{ and } P_Y \geq P_X\}$$

and P_X is the probability of the observed table and P_Y is the probability of the sampled table Y . The probability of each table $P(Y)$ is given by,

$$P(Y) = \begin{cases} \frac{\prod_{i=1}^g \prod_{n=1}^K \binom{m_n^{(i)}}{a_{0,n}^{(i)}, \dots, a_{n,n}^{(i)}}}{\prod_{n=1}^K \binom{M_n}{a_{0,n}, \dots, a_{n,n}}} & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} = a_{r,n}, r = 0, \dots, n \\ 0 & \text{if } \sum_{i=1}^g A_{r,n}^{(i)} \neq a_{r,n} \end{cases}$$

$n = 1, \dots, K, i = 1, \dots, g$

- A Monte Carlo estimate of the $P - value$ is obtained by sampling N tables from the reference set Γ . Each of the N tables sampled from the reference set Γ is ordered by the modified Freeman-Tukey statistic,

$$D_Y = \sum_{n=1}^K \sum_{i=1}^g \sum_{r=0}^n \left(\sqrt{a_{ij}} + \sqrt{a_{ij} + 1} - \sqrt{4 \left(\frac{m_n^i n_r^i}{N} \right) + 1} \right)^2 \quad (3.2)$$

to detect those tables that are at least as extreme as the observed table X .

- For each table $Y_i \in \Gamma$ that is sampled, define W_i as:

$$W_i = \begin{cases} 1 & D_{Y_i} \geq D_X \\ 0 & \text{otherwise} \end{cases}$$

The Monte-Carlo $P - value$ is then given by

$$\hat{P} - value = \frac{1}{N} \sum_{i=1}^N W_i$$

Since variable W_i is a Bernoulli random variable. The asymptotic standard error of the Monte Carlo estimate of the $P - value$ is given by:

$$\text{s.e.}(\hat{p}) = \sqrt{\frac{\hat{p}(1 - \hat{p})}{N - 1}}$$

and the asymptotic $100(1 - \alpha)\%$ confidence intervals is given by:

$$\hat{p} \pm z_{\frac{\alpha}{2}} \text{s.e.}(\hat{p})$$

Although \hat{P} is an unbiased estimate of the exact $P - value$, simply replacing the exact $P - value$ with \hat{P} fails to control correctly for type I error rate[42]. \hat{P} underestimates the $P - value$ if the null hypothesis is true when the number of W_i is very small relative to N , and it can result in obtaining a $P - value$ of zero if the observed statistic is greater than all the permuted test statistics. For this reason an adjusted Monte Carlo $P - value$ estimate was suggested [13] as is:

$$\hat{p} = \frac{1}{(N + 1)} \left(\sum_{i=1}^N W_i + 1 \right) \quad (3.3)$$

The Curtis adjusted $P - value$ has expectation, $\frac{NP+1}{N+1}$, and is biased and overestimates the $P - value$ by $\frac{1-P}{N+1}$.

The discreteness in small sample distributions can be adjusted by using the mid- $p - value$ approach suggested by Lancaster [27]. The mid P-value behaves much more like the P-value of a test statistic with a continuous distribution. For a test statistic T with observed value t_{obs} and one-sided alternate hypothesis H_a . The mid P-value is given by :

$$\text{mid } P - value = \frac{1}{2}P(T = t_{obs}) + P(T > t_{obs})$$

This adjustment causes the mid $P - value$ to be less than the ordinary P-value by about half of the observed result, but it is less conservative compared to the ordinary $P - value$ of an exact test [1].

3.4 An Algorithm for Generating Tables

The tables from the reference set are generated by the Algorithm AS 159 Patefield (1981). Each table with the same row and column marginal sums is generated under the null hypothesis assumption of no association between the rows and columns categories. The conditional probability distribution of each entry $A_{r,n}$ and its expected value in a table given the previous entries in the rows and columns is found. The probability distribution of $A_{r,n}$ is accumulated starting with the $A_{r,n}$ equal to the nearest integer of it's expected value. The c.p.u. times required for this algorithm unlike the algorithm by Boyett(1979), depends more on the dimension of the table rather than the sample size. Patefield's algorithm performs considerably better than Boyett's Algorithm AS 144 for large dimension tables.

3.5 Simulation Study I

A simulation study was conducted to determine the operating characteristics of the conditional exact test. The $P - values$ of a homogeneity test between two groups (control and treatment) were computed using various sample sizes by varying the difference in success probabilities between the two groups. The marginal

probability for the control group p_1 was held constant at 0.1 and the probability of the treatment group was varied from $p_2 = 0.1, \dots, 0.9$. The sample sizes for both the control and treatment groups were equal and varied from 50, 100 and 200.

The method proposed by Lunn and Davies [29] was used to generate correlated binary random variables with exchangeable correlation structure ρ_j within the j th cluster. Although this method is very simple to implement, it seems to be faster than other existing algorithms for generating clustered binary data.

To give a brief description of the algorithm, let X_{ijk} be the k th binary random variable in the j th cluster of the i th treatment group for $i = 1, 2, k = 1, \dots, n_{ij}, j = 1, \dots, N_i$. Thus N_1 and N_2 are the number of clusters in Control and Treatment group respectively. The cluster sizes n_{ij} were chosen to follow a random Poisson distribution with mean $\lambda = 5$ and $\lambda = 10$. X_{ijk} 's and were generated as follows:

Let Y_{ijk} and Z_{ij} be independent binary random variables with $p_i = P(Y_{ijk} = Z_{ij} = 1)$. Further let U_{ijk} be a binary random variable with $P(U_{ijk} = 1) = \theta_i$ independent of Y_{ijk} and Z_{ij} . Generate $X_{ijk} = (1 - U_{ijk})Y_{ijk} + U_{ijk}Z_{ij}$. Then $E(X_{ijk}) = (1 - \theta_i)p_i + \theta_i p_i = p_i$ and $var(X_{ijk}) = E(X_{ijk})^2 - (E(X_{ijk}))^2 = p_j(1 - p_j)$. Further $cov(X_{ijk}, X_{ijl}) = E(X_{ijk}, X_{ijl}) - p_i^2 = \theta^2_i p_i(1 - p_i)$ for $k \neq l$ and $cov(X_{ijk}, X_{ijl}) = 0$ for $j \neq l$ so that $\rho_i = \theta_i^2$.

The results from the simulation are displayed in Table 6. The exact conditional approach detects a significant change in treatments as the marginal probabilities of the two treatment groups begin to change. The level of significance increases as the difference in probabilities of the two groups increases. Generally, it was also observed that an increase in sample size is more likely to detect significance. The p - values appear to increase as the intra cluster correlation is varied from 0.3 to 0.9. This demonstrates that there is some intra cluster effect. The results of the

simulation study also show that the size of the clusters did not have any significant effect on the p – *values*. Increasing the mean cluster size from 5 to 10, the results remained generally the same.

Table 6: Simulation Study I Results

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$p_1 = p_2 = 0.1$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.5036603	0.4883057	0.4906973
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.2881089	0.1443886	0.03172927
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.07149251	0.005467532	0.001004995
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.001715285	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.5102378	0.503006	0.4948531
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.3466993	0.238026	0.1015674
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.1478362	0.03147852	0.001747253
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.00674026	0.001015984	0.000999001
$p_1 = 0.1, p_2 = 0.7$			
Continued on next page			

Table 6 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.001010989	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = p_2 = 0.1$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.5095644	0.509015	0.4906503
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.4020539	0.3217502	0.1935874
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.2189271	0.09046254	0.01030569
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.001564436	0.001564436	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.001114885	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.5124665	0.4985754	0.5069441
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.2722398	0.1042607	0.01398202
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.05241459	0.002683317	0.000999001
Continued on next page			

Table 6 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.001330669	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.4909351	0.4971698	0.4943417
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.3182138	0.2037013	0.06839161
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.1065634	0.01919381	0.001151848
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.004274725	0.001	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.001004995	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.4761019	0.4751439	0.5038921
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.9$	0.4080639	0.3138551	0.1967662
Continued on next page			

Table 6 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$\lambda = 10$			
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.2180729	0.09042458	0.09042458
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.02597003	0.001347652	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.001376623	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.000999001	0.000999001	0.000999001

3.6 Simulation Study II

Theorem 3.6.1 (de Finetti). *An infinite sequence of binary random variables (X_1, X_2, \dots) is exchangeable if and only if there exists a probability distribution function F on $[0, 1]$ such that for all n [15]*

$$P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = \int_0^1 \theta^{s_n} (1 - \theta)^{n - s_n} dF(\theta), \quad (3.4)$$

$$P\left(\sum_{i=1}^n X_i = s_n\right) = \binom{n}{s_n} \int_0^1 \theta^{s_n} (1 - \theta)^{n - s_n} dF(\theta) \quad (3.5)$$

where, $\sum_{i=1}^n X_i = s_n$

de Finetti's theorem assumes that any finite set under consideration is a subset of an infinite sequence. This assumption is not necessarily true in many practical applications. The Beta-Binomial may be derived from De Finetti's theorem when the mixing distribution F is a beta distribution. The intra-litter correlation is restricted to be positive in a Beta-Binomial model [10]. Diaconis [15], showed how the De Finetti theorem fails in a finite sequence scenario and also obtained finite sequence representation of de Finetti's theorem.

3.6.1 The p-power and q-power Models

In the interest of formulating parametric models, Kuk [26] introduced p-power for exchangeable binary random variables. Kuk [26] showed that the power family model is a completely monotone function. This constraint is needed in order that the forward differences of all marginal response probabilities is positive for a valid probability model [10]. Define

$$\lambda_{r+k} = P(X_1 = \dots = X_{r+k} = 1) = p^{(r+k)\gamma}, \quad 0 \leq p, \gamma \leq 1. \quad (3.6)$$

Thus,

$$P(R = r) = \binom{n}{r} \sum_{k=0}^{n-r} (-1)^k \binom{n-r}{k} p^{(r+k)\gamma}, \text{ where } \sum_{k=1}^n X_k = R. \quad (3.7)$$

Kuk also gave the q-power model as

$$\lambda'_k = P(X'_1 = \dots = X'_k = 1) = P(X_1 = 0 = \dots = X_k = 0) = q^{k^\gamma} \quad (3.8)$$

where

$$P(R = r) = P(R' = n - r) = \binom{n}{r} \sum_{k=0}^{n-r} (-1)^k \binom{r}{k} q^{(n-r+k)^\gamma} \quad (3.9)$$

Using this ,we can show that the intra cluster correlation is given as

$$\rho = \frac{q^{2^\gamma} - q^2}{q(1 - q)}, \text{ where } 0 < q < 1 \quad (3.10)$$

where γ is the measure of positive intra-cluster correlation. $\gamma = 1$ represents the case of independent observations and $\gamma = 0$ specifies complete intra-cluster dependence. $\gamma > 1$ results in negative intra-cluster correlation ($\rho < 0$). $0 < \gamma < 1$ indicates positive intra cluster correlation.

A reparametrization gives

$$\gamma = \frac{\log \left[\frac{\log(q^2 + \rho q(1-q))}{\log(q)} \right]}{\log(2)} \quad (3.11)$$

The p-power distribution given by (3.7) under-estimates the probability of at least one positive response $P(R \geq 1)$ by assigning high probability mass at zero[26]. This leads to overestimation of a safe dose in a litter-based approach to quantitative risk assessment [26]. The q-power distribution given by (3.9) does not have this problem. Kuk [26] showed that the probability distribution functions of the q-power distribution in cases where the response probability or intra-cluster correlation is small closely follows bell shape in comparison to their p-power counterparts.[26]. The q-power model in (3.9) does not model positive intra-litter correlation only. In most practical applications that involve clustered binary data, intra-litter correlation is positive. It can be shown that $\frac{-1}{n_{max}} \leq \rho \leq 1$, where n_{max} is the maximum cluster size [17]. If $\sum_{r=0}^n P(R \geq r) = 1$ the q-power model (3.9) defines a valid distribution even when $\gamma > 1$ as long as $P(R \geq r) \geq 0$. For $r = 0, \dots, n$, this

leads to negative intra-cluster correlation defined by Equation 3.10. A value of $\gamma > 1$ results in negative intra-cluster correlation. Thus when these simple conditions are satisfied, the q-power distribution can be adapted to model data with negative intra-litter correlation. The q-power distribution is convenient in dose response modeling when the probability that at least one cluster variable is affected in contrast to the p-power distribution, since

$$P(R \geq 1) = 1 - P(R = 0) = 1 - P(X_1 = \dots, X_n = 0) = 1 - q^{n\gamma} \quad (3.12)$$

The corresponding p-power family is given by

$$P(R \geq 1) = 1 - P(R = 0) = 1 - \binom{n}{r} \sum_{k=0}^{n-r} (-1)^k \binom{n}{k} p^{k\gamma} \quad (3.13)$$

A second simulation study was conducted to determine the characteristics of the conditional exact tests. The $P - values$ of a homogeneity test between a control and a treatment groups were computed at various sample sizes by varying the difference in success probabilities between the two groups. The marginal probability for the control group p_1 was held constant at 0.1 and the probability of the treatment group was varied from $p_2 = 0.1, \dots, 0.9$. The number of clusters for both the control and treatment groups were made equal and varied at 50, 100 and 200. λ is the mean of the Poisson distribution used to generate random cluster sizes. For each set of parameters, the simulation was repeated a thousand times and the mean p-value recorded.

3.6.2 Inverse Transform Method

The Inverse Transform method was used to generate correlated binary random variables with intra-litter correlation ρ . To generate a random variable R with probability distribution given by (3.9), we follow these two steps:

Steps

1. Generate probabilities $p_0 = P(R \leq 0)$, $p_1 = P(R \leq 1)$, $p_2 = P(R \leq 2)$, \dots

2. Generate $U \sim u(0, 1)$ then set $R = k$ if $p_{k-1} < U < p_k$.

The results from the simulation are summarized in Table 7. The exact conditional approach detects a significant change between control and treatment group as the marginal probabilities of the two treatment groups begin to change. The p-values decrease as the difference in marginal probabilities between the control and treatment group increases. Generally, the increase in sample size has an effect on significance. Larger sample sizes were more likely to detect significance for a fixed set of parameters as expected, while for a fixed set of parameters, increasing intra-cluster correlation resulted in increasing p-values. This suggests that intra-cluster correlation has an effect on the significance of the test. Larger sample sizes are needed to detect significant differences as intra-litter correlations increase.

Table 7: Simulation Study II Results

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$p_1 = p_2 = 0.1$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.5075455	0.5068741	0.4948332
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.3366064	0.2118432	0.07347453
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.1192218	0.01818382	0.001256743
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.00458042	0.001003996	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.001010989	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.4972358	0.5091479	0.5035944
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.3752328	0.2706613	0.156002
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.176035	0.06263437	0.00662038
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.01225175	0.001110889	0.000999001
$p_1 = 0.1, p_2 = 0.7$			
Continued on next page			

Table 7 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.001063936	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.5154116	0.5028581	0.5005315
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.3893546	0.3132098	0.1918781
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.222013	0.08045654	0.01461738
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.02365135	0.001455544	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.001165834	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 5$	0.000999001	0.000999001	0.000999001
$p_1 = p_2 = 0.1$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.4975125	0.5038791	0.5048911
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.3001578	0.1708402	0.04642657
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.08166134	0.01356244	0.001064935
Continued on next page			

Table 7 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.003024975	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.001018981	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.3$ $\lambda = 10$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.5004855	0.4963666	0.5109191
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.3816813	0.2802967	0.1594915
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.1927493	0.06670829	0.00782018
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.01223177	0.001152847	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.001191808	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.6$ $\lambda = 10$	0.000999001	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.1$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.50203	0.5016913	0.5188941
$p_1 = 0.1, p_2 = 0.2$ $\rho_1 = \rho_2 = 0.9$	0.4172587	0.3215764	0.2148541
Continued on next page			

Table 7 – continued from previous page

Simulation Parameters	P- Value, $N_1 = N_2 = 50$	P- Value, $N_1 = N_2 = 100$	P- Value, $N_1 = N_2 = 200$
$\lambda = 10$			
$p_1 = 0.1, p_2 = 0.3$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.2392268	0.1025734	0.01953247
$p_1 = 0.1, p_2 = 0.5$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.03052048	0.001665335	0.000999001
$p_1 = 0.1, p_2 = 0.7$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.001525475	0.000999001	0.000999001
$p_1 = 0.1, p_2 = 0.9$ $\rho_1 = \rho_2 = 0.9$ $\lambda = 10$	0.000999001	0.000999001	0.000999001

The Lunn and Davis approach to generating correlated binary data results in p-values that are in general slightly less than that of the Inverse Transform of the q-power distribution for any fixed set of parameters. The two simulation methods share all other similar characteristics with each other.

3.7 Application

The exact test procedure was applied to the analysis of a developmental toxicity study of ethylene glycol diethyl ether (EDGE). The data was obtained in an experiment in which pregnant New Zealand white rabbits were exposed to the compound EGDE. The effect of ethylene glycol diethyl ether on their fetal development was then studied. In the study, four groups of pregnant does were randomly assigned to dose levels 0, 25, 50, and 100 milligrams per kilogram body weight of EGDE. For each litter and at each dose level, the adverse response used is the combined number of fetal malformation and fetal death. The data are presented in Table 8. The frequency distribution of live fetuses with malformations are grouped into cluster sizes ranging from 2 to 15. The data appears to be relatively sparse with no indication that dose level has impact on cluster size. From the table, for each fixed cluster size, frequency of malformations appears to increase with increasing dose level. Figure 1 shows the probability of at least one, two, three and four responses across the various dose groups. In general, these probabilities increase with increasing dose demonstrating the existence of a trend. Figure 2 shows a strip chart plot of the responses. It is observed that dose group 100 has the highest proportion fetuses with malformations. Lower dose groups 0, 25 and 50 have relatively lower proportion of fetuses with malformations. Figure 3 shows that the marginal proportion of fetuses affected also increase with increasing dose.

Table 8: Data from Ethylene glycol ether(EDGE) Laboratory experiment: A Frequency Table Representation

Dose Level	No. malformed <i>i</i>	No. of live fetuses													
		2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	2													
	1		1		1	1		1							
	2														
	3		1												
	4				1					2					
	5										1				
	6											1			
	7							1							
	8								1						
25	0		1	1	4	2	1	2	2	1		1			
	1					1	1		1	2					
	2											2		1	
	3								1	2					
	4					1				1					
	5														
	6														
	7							1							
	8								1						
50	0	2	1		1		1	1	1	1					
	1				1		1	1	1	1		1			
	2			1		1	1	1	1	1		1			
	3														
	4													1	
	5														
	6														
	7														
	8								1			1			
100	0	1	2												
	1	1													
	2														
	3														
	4				1	1	1			1					
	5					1	1								
	6														
	7														
	8								1						
	9														
	10							1		2					
	11									1					
	12										2		1		

3.7.1 Summary Statistics for Edge Data

Table 9: Summary Statistics for EDGE Data

	Dose Level				
	0	25	50	100	Total
Clusters	21	23	25	25	94
Mean Cluster Size	7.79	8.28	8.31	7.71	8.02
Sample Size	218	265	216	239	938
Malformations	40	47	54	171	312
Response Rate (%)	18.35	17.74	25	71.55	33.26
Probability of at least one fetus affected	0.46	0.53	0.69	0.87	0.64

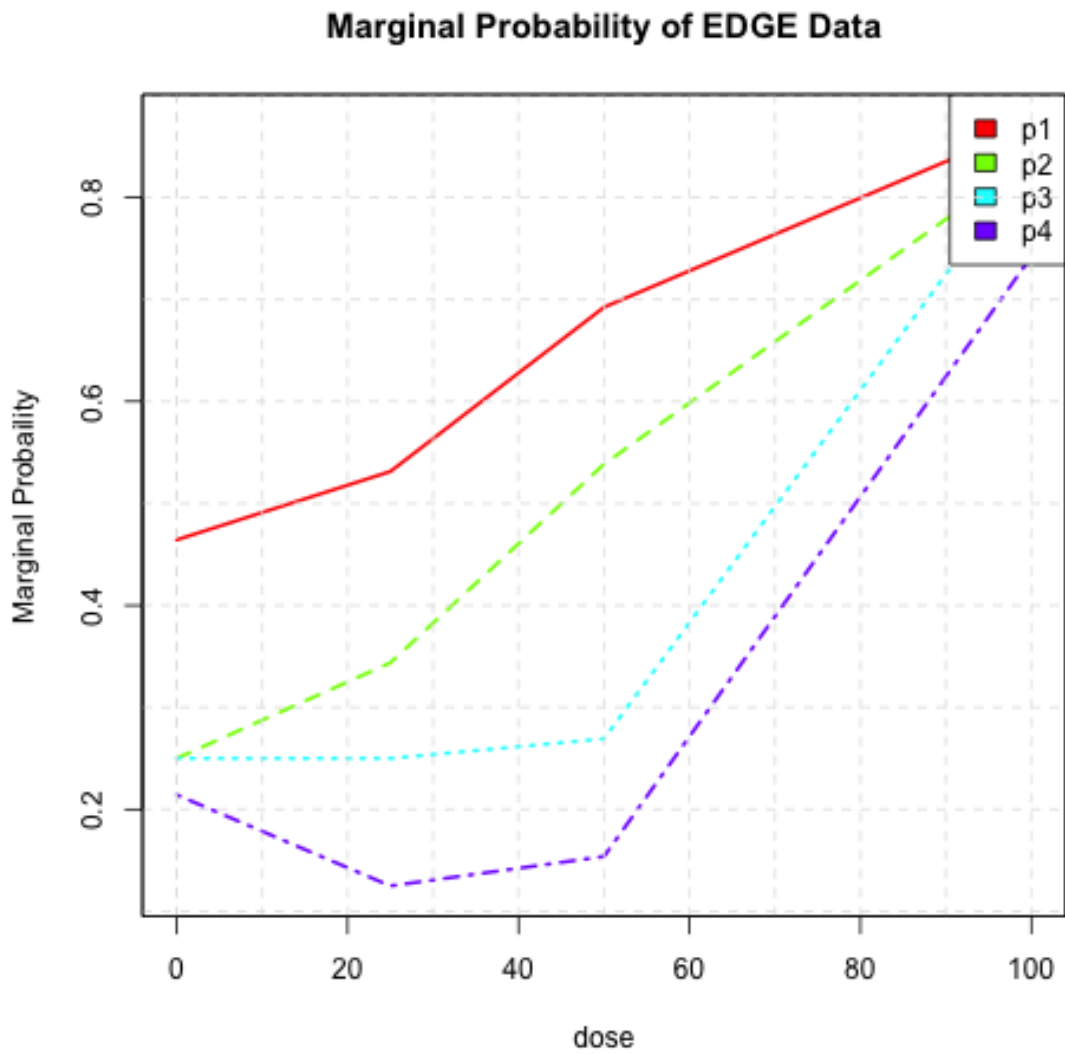


Figure 1: Marginal Response Probability Plot for EDGE Data

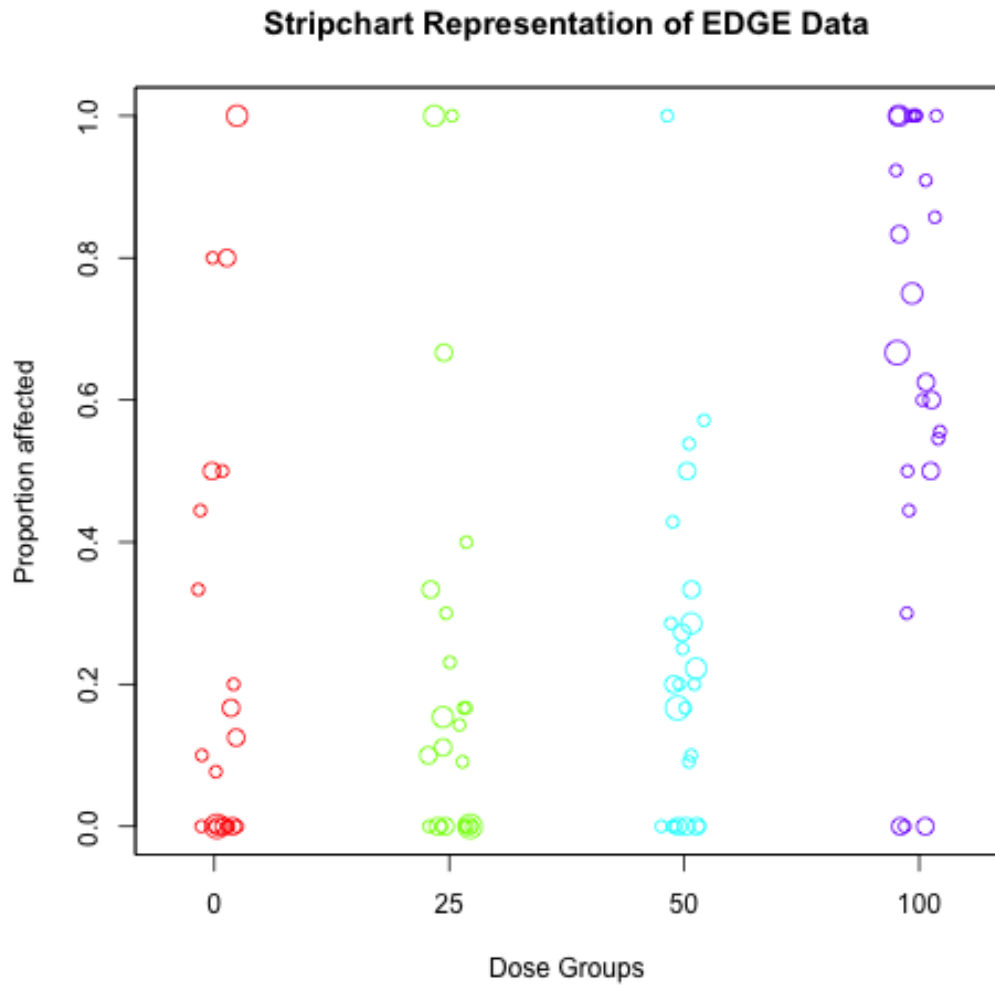


Figure 2: Strip chart Representation for EDGE Data

Marginal Proportion and Estimated Marginal Proportions

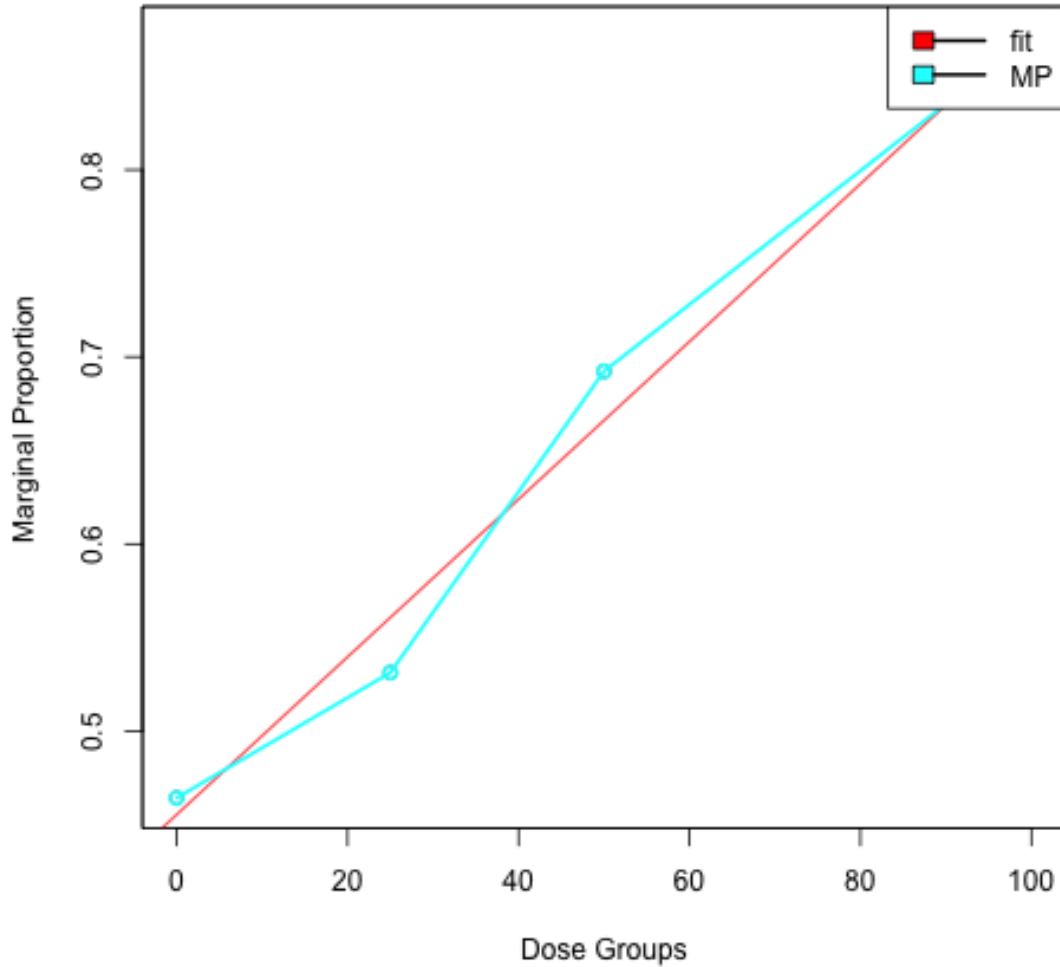


Figure 3: Fit of Marginal Probability for Edge Data

3.7.2 Treatment Test of Heterogeneity

To test the hypothesis of no adverse effect of EDGE against possible adverse effect. We test the null hypothesis

$$H_0 : P_{r,n}^{(i)} = P_{r,n}, i = 1, \dots, 4 \quad r = 0, \dots, 12 \quad n = 2, \dots, 15 \quad \text{versus the alternate}$$

$$H_a: \text{At least one } P_{r,n}^{(i)} \neq P_{r,n}^{(i+1)}$$

In Table 10, we summarize the results of this test across all four treatment groups. The results indicate a significant difference in response rates between the four dose groups (p -value = 0.02894971). The estimates are based on 10,000

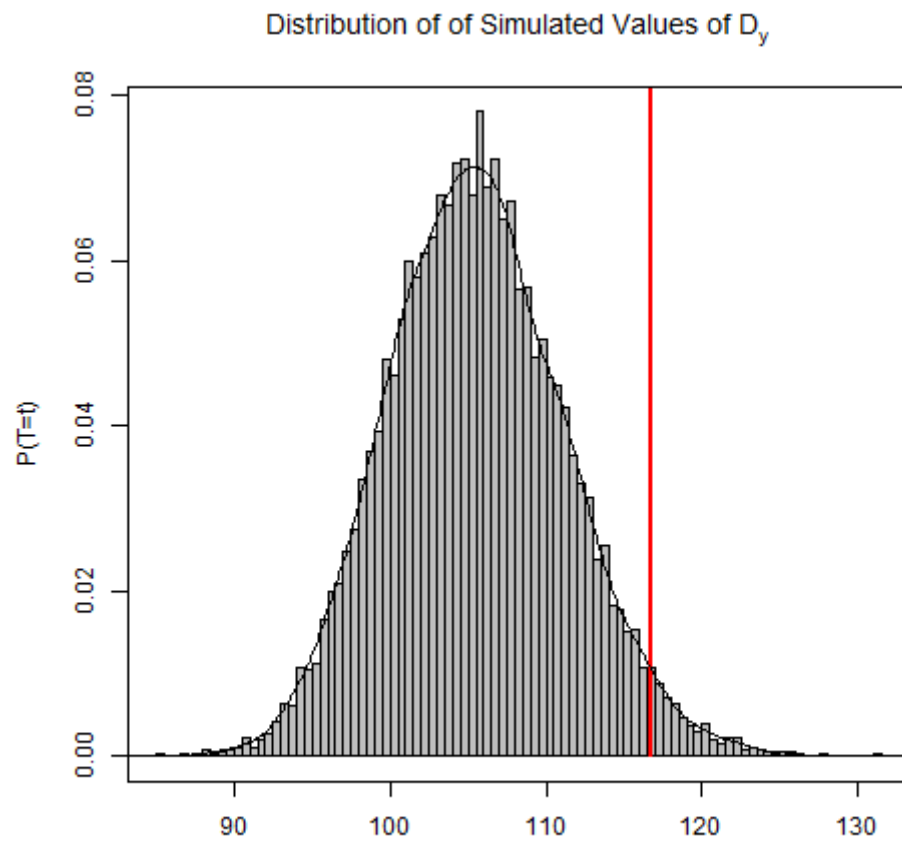
random samples from the reference set. The distribution of the test statistic D_Y displayed by Figure 4 shows, the test statistic appears to be unimodal and skewed to the left.

Table 10: Exact Conditional Test

Treatment Groups	Mid P-value	P-value	99% Confidence Interval
0,25,50,100	0.028945	0.02894971	(0.02758443, 0.03031499)

Table 11: Comparison with Other Test

Treatment Groups	Exact Conditional	Rao-Scott Homogeneity Test(Asymptotic)
0,25,50,100	0.028945	< 0.001



red line is the observed value of D_x

Figure 4: 10000 Simulated Tables of Eq 3.2 for Edge data

CHAPTER 4

EXACT STRATIFIED LINEAR RANK TEST

4.1 Stratified Representation of Data

Stratification of clustered data allows analysis to be made on data with the same cluster sizes. When cluster sizes are fixed, each row in a stratum of the g treatment groups can be viewed as g independent multinomial distributions with K categories. Each observation $A_{r,j}^i$ represents the number of clusters with r responses $r = 0, \dots, K$ and stratum/cluster size $j, j = 1, \dots, K$ in the i th treatment group, $i = 1, \dots, g$.

The stratum scores $w_{i,j}, i = 0, 1, \dots, K$ and $j = 1, \dots, K$ are monotone non-decreasing weights assigned ($w_{0,j} \leq w_{1,j} \leq \dots \leq w_{K,j}$) to the j th stratum with r responses. The weights $w_{i,j}$ are assigned to the ranked column in each stratum. Let $(u_{0i}, u_{1i}, u_{2i}, \dots, u_{gi})$ be monotone non-decreasing weights that is $(u_{0i} \leq u_{1i} \leq u_{2i} \leq \dots \leq u_{gi})$. The weights $u_{ij}, i = 1, \dots, g$ and $j = 1, \dots, K$ are weights assigned to the ranked row sum in each stratum. The stratified representation is shown in Table 12.

Table 12: Stratified Representation of Clustered Data

Cluster size	Treatment	Response					Row Total	Row Weights
		0	1	2	...	n		
		Stratum 1						
1	0	$A_{0,1}^{(1)}$	$A_{1,1}^{(1)}$	0	...	1	m_1^1	u_{11}
	2	$A_{0,1}^{(2)}$	$A_{1,1}^{(2)}$	0	...	0	m_1^2	u_{21}

	g	$A_{0,1}^{(g)}$	$A_{1,1}^{(g)}$	$A_{2,1}^{(g)}$...	$A_{r,1}^{(g)}$	m_1^g	u_{g1}
Column Total		$n_{0,1}$	$n_{1,1}$	$n_{2,1}$...	$n_{r,1}$	N_1	
Column Weights		w_{01}	w_{11}	w_{21}	...	w_{r1}		
		Stratum 2						
2	1	$A_{0,2}^{(1)}$	$A_{1,2}^{(1)}$	$A_{2,2}^{(1)}$...	0	m_2^1	u_{12}
	2	$A_{0,2}^{(2)}$	$A_{1,2}^{(2)}$	$A_{2,2}^{(2)}$...	0	m_2^2	u_{22}

	g	$A_{0,2}^{(g)}$	$A_{1,2}^{(g)}$	$A_{2,2}^{(g)}$...	$A_{r,2}^{(g)}$	m_2^g	u_{g2}
Column Total		$n_{0,2}$	$n_{1,2}$	$n_{2,2}$...	$n_{r,2}$	N_2	
Column Weights		w_{02}	w_{12}	w_{22}	...	w_{r2}		
		Stratum 3						
3	1	$A_{0,3}^{(1)}$	$A_{1,3}^{(1)}$	$A_{2,3}^{(1)}$...	0	m_3^1	u_{13}
	2	$A_{0,3}^{(2)}$	$A_{1,3}^{(2)}$	$A_{2,3}^{(2)}$...	0	m_3^2	u_{23}

	g	$A_{0,3}^{(g)}$	$A_{1,3}^{(g)}$	$A_{2,3}^{(g)}$...	$A_{r,3}^{(g)}$	m_3^g	u_{g3}
Column Total		$n_{0,3}$	$n_{1,3}$	$n_{2,3}$...	$n_{r,3}$	N_3	
Column Weights		w_{03}	w_{13}	w_{23}	...	w_{r3}		
...	
...	
...	
...	
...	
		Stratum K						
K	1	$A_{0,K}^{(1)}$	$A_{1,K}^{(1)}$	$A_{2,K}^{(1)}$...	$A_{K,K}^{(1)}$	m_K^1	u_{1K}
	2	$A_{0,K}^{(2)}$	$A_{1,K}^{(2)}$	$A_{2,K}^{(2)}$...	$A_{K,K}^{(2)}$	m_K^2	u_{2K}

	g	$A_{0,K}^{(g)}$	$A_{1,K}^{(g)}$	$A_{2,K}^{(g)}$...	$A_{r,K}^{(g)}$	m_K^g	u_{gK}
Column Total		$n_{0,K}$	$n_{1,K}$	$n_{2,K}$...	$n_{r,K}$	N_K	
Column Weights		$w_{0,K}$	$w_{1,K}$	$w_{2,K}$...	$w_{r,K}$		

Where $A_{r,j}^{(i)} = 0$, whenever $r > j$.

4.2 Reference Set

Denote the reference set for each stratum j as the set of all possible contingency tables with fixed marginal column and row sums as the observed table sums for treatment groups $i = 1, \dots, g$, cluster sizes $j = 1, \dots, K$, where K is the maximum cluster size in the experiment with responses $r = 0, \dots, n$. The reference set in the j th stratum is:

$$\gamma_j = \left\{ y_j : y_j \text{ is } g \times K \text{ contingency table; } \sum_{r=0}^n A_{r,j}^i = m_r^i, \sum_{i=1}^g A_{r,j}^i = n_{r,j} \right\}$$

The reference set across all the strata can therefore be represented by the Cartesian product of all reference sets from each of the j strata .

$$\Gamma = \gamma_1 \times \gamma_2 \cdots \times \gamma_K = \{Y : y_j \in \gamma_j, j = 1, \dots, K\}$$

Thus Y is made of γ_j stratums, $j = 1, \dots, K$. Each Y can be seen as $K \times (g \times K)$ dimensional array.

4.3 Exact Linear Trend Test

The g rows of each stratum represent increasing doses of a drug and the K distinctly ordered columns represent the progressively increasing responses. We wish to investigate whether response increases with increasing dose. We consider a general linear rank statistic T , which detects departure from the null hypothesis of no dose effect. T is the sum of linear rank statistics across the K strata. The asymptotic distribution of the standardized test statistic T^* is the standard normal distribution with mean and standard deviation of 0 and 1 respectively. The square of the standard test statistic is also asymptotically distributed as a chi-square with one degree of freedom. Suppose each response falls into one of the K categories of a multinomial distribution, Let $\Pi_j^i = (\pi_{j0}^i, \pi_{j1}^i, \dots, \pi_{jK}^i)$ be the multinomial probabilities along each row in stratum j for dose group i . The null hypothesis for stratum j can be written as

$$H_0^j : \Pi_j^1 = \Pi_j^2 = \dots = \Pi_j^g$$

and the hypothesis of trend as $H_a^j : \Upsilon_j^1 \leq \Upsilon_j^2 \leq \dots \leq \Upsilon_j^g$ with at least one

strict inequality

where for each fixed stratum $j, j = 1, \dots, K, \Upsilon_j^i = \sum_{r=0}^n \pi_{jr}^i$.

The linear trend test statistic is given by:

$$T = \sum_{j=1}^K T_j = \sum_{k=1}^K \sum_{i=1}^g \sum_{r=0}^n u_{ij} w_{rj} A_{r,j}^{(i)} \quad (4.1)$$

where $w_{r,j}$ is the weight assigned to the r th response within the j th stratum and u_{ij} is the weight assigned to the i th treatment group in the j th stratum. There exist several possible choices of weights, with each choice resulting in a different test. The weights can be chosen to make the test cluster size invariant or cluster size specific. For cluster size invariant scores, define $n_j = \sum_{k=1}^K n_{j,k}$, thus the sum of response is used in weight computation in each response group.

Theorem 4.3.1. *Under the null hypothesis of no row by column interaction, for the j th stratum $j = 1, \dots, K$, the test statistic is T_j as given in (4.1) has mean and variance as:*

$$E(T_j) = \frac{\left(\sum_{i=1}^g u_{ij} m_j^i \right) \left(\sum_{r=0}^n w_{rj} n_{rj} \right)}{N_j}$$

$$Var(T_j) = \frac{\left[\sum_{i=1}^g u_{ij}^2 m_j^i - \frac{\left(\sum_{i=1}^g u_{ij} m_j^i \right)^2}{N_j} \right] \left[\sum_{r=0}^n w_{rj}^2 n_{rj} - \frac{\left(\sum_{r=0}^n w_{rj} n_{rj} \right)^2}{N_j} \right]}{N_j - 1}$$

Proof. : We give the proof 2×2 case. The $g \times K$ case follows as a natural extension.

Table 13: Stratified Exact Linear Trend Test for 2×2 Tables

Treatment	Response 0	1	Row Total	Row Weights
0	$A_{0,k}^0$	$A_{1,k}^0$	m_k^0	u_k^0
1	$A_{0,k}^1$	$A_{1,k}^1$	m_k^1	u_k^1
Column Total	$n_{0,k}$	$n_{1,k}$	N_k	
Column Weights	$w_{0,k}$	$w_{1,k}$		

If each is treatment is fixed, the the number of 0 responses in treatment group 0, $A_{0,k}^0$ and the number of 0 responses in treatment group 1, $A_{0,k}^1$ are independently distributed as Binomial distributions $B(m_k^0, \pi_{00})$ and $B(m_k^1, \pi_{10})$ respectively.

π_{00} and π_{10} are probability of 0 response in treatment groups 0 and 1 respectively.

Let $A_0^k = A_{0,k}^0 + A_{0,k}^1$ and $A_1^k = A_{1,k}^0 + A_{1,k}^1$, then under the null hypothesis that $\pi_{00} = \pi_{01} = \pi_{10} = \pi_{11} = \pi$ and $A_0^k \sim B(N_k, \pi)$ and $A_1^k \sim B(N_k, \pi)$. The distribution of each $A_{r,k}^i$ stated in terms of $A_{0,k}^0$

$$\begin{aligned}
& P(A_{0,k}^0 = a_{0,k}^0, A_{1,k}^0 = a_{1,k}^0, A_{0,k}^1 = a_{0,k}^1, A_{1,k}^1 = a_{1,k}^1 | H_0, A_0^k = a_0^k, A_1^k = a_1^k) \\
&= \frac{P(A_{0,k}^0 = a_{0,k}^0, A_{1,k}^0 = a_{1,k}^0, A_{0,k}^1 = a_{0,k}^1, A_{1,k}^1 = a_{1,k}^1 | H_0)}{P(A_0^k = a_0^k, A_1^k = a_1^k)} \\
&= \frac{\binom{m_k^0}{a_{0,k}^0} \pi^{a_{0,k}^0} \pi^{m_k^0 - a_{0,k}^0} \binom{m_k^1}{a_{0,k}^1} \pi^{a_{0,k}^1} \pi^{m_k^1 - a_{0,k}^1}}{\binom{N_k}{a_0^k} \pi^{a_0^k} (\pi)^{N_k - a_0^k}} \\
&= \frac{\binom{m_k^0}{a_{0,k}^0} \binom{m_k^1}{a_{0,k}^1}}{\binom{N_k}{a_0^k}} = \frac{\binom{m_k^0}{a_{0,k}^0} \binom{N_k - m_k^0}{n_{0,k} - a_{0,k}^0}}{\binom{N_k}{a_0^k}}
\end{aligned}$$

That is $A_{r,k}^{(i)} \sim \text{Hypergeometric}(a_0^k, m_k^{(0)}, m_k^{(1)})$. Hence $E(A_{0,k}^0) = \frac{m_k^0 n_{0k}}{N_k}$ and $\text{Var}(A_{0,k}^0) = \frac{m_k^0 n_{0k} m_k^1 n_{1k}}{N_k^2 (N_k - 1)}$. Let $\pi_{00} = P(0 \text{ responses in dose group } 0)$ and $\pi_{01} = P(0 \text{ responses in dose group } 1)$.

Under the null hypothesis that $\pi_{00} = \pi_{10} = \pi$ the conditional distribution of any Table i , with counts $A_{r,k}^i$ given row and column marginals has a hypergeometric distribution with mean and variance given below:

$$E(A_{r,k}^i) = \frac{m_k^i n_{rk}}{N_k}$$

$$\text{Var}(A_{r,k}^i) = \frac{m_k^i (N_k - m_k^i) n_{rk} (N_k - n_{rk})}{N_k^2 (N_k - 1)}$$

$$T_k = \sum_{i=1}^g \sum_{r=0}^n u_{ik} w_{rk} A_{r,k}^{(i)}$$

$$E(T_k) = \sum_{i=1}^g \sum_{r=0}^n u_{ik} w_{rk} E(A_{r,k}^{(i)})$$

$$E(T_k) = \sum_{i=1}^g \sum_{r=0}^n u_{ik} w_{rk} \frac{m_k^i n_{rk}}{N_k}$$

$$E(T_k) = \frac{1}{N_k} \sum_{i=1}^g (u_{ik} m_k^i) \sum_{r=0}^n (w_{rk} n_{rk})$$

$$\text{Var}(A_{r,k}^i) = \frac{m_k^i (N_k - m_k^i) n_{rk} (N_k - n_{rk})}{N_k^2 (N_k - 1)} = \frac{m_k^i n_{rk}}{N_k - 1} \left(\frac{N_k - m_k^i}{N_k} \right) \left(\frac{N_k - n_{rk}}{N_k} \right)$$

$$= \frac{m_k^i n_{rk}}{N_k - 1} \left(1 - \frac{m_k^i}{N_k} \right) \left(1 - \frac{n_{rk}}{N_k} \right)$$

$$= \frac{1}{N_k - 1} \left(m_k^i - \frac{(m_k^i)^2}{N_k} \right) \left(n_{rk} - \frac{n_{rk}^2}{N_k} \right)$$

Under the null hypothesis of no row by column interaction:

$$\begin{aligned}
\text{Var}(T_k) &= \sum_{i=0}^g \sum_{r=0}^K u_{ik}^2 w_{rk}^2 \text{Var}(A_{r,k}^{(i)}) \\
\text{Var}(T_k) &= \sum_{i=0}^g \sum_{r=0}^K u_{ik}^2 w_{rk}^2 \frac{1}{N_k - 1} \left(m_k^i - \frac{(m_k^i)^2}{N_k} \right) \left(n_{rk} - \frac{n_{rk}^2}{N_k} \right) \\
&= \frac{1}{N_k - 1} \left(\sum_{i=0}^g u_{ik}^2 m_k^i - \frac{\sum_{i=0}^g u_{ik}^2 (m_k^i)^2}{N_k} \right) \left(\sum_{r=0}^K w_{rk}^2 n_{rk} - \frac{\sum_{r=0}^K w_{rk}^2 n_{rk}^2}{N_k} \right) \\
&= \frac{1}{N_k - 1} \left(\sum_{i=0}^g u_{ik}^2 m_k^i - \frac{\left(\sum_{i=0}^g u_{ik} m_k^i \right)^2}{N_k} \right) \left(\sum_{r=0}^K w_{rk}^2 n_{rk} - \frac{\left(\sum_{r=0}^K w_{rk} n_{rk} \right)^2}{N_k} \right)
\end{aligned}$$

□

$$T_k^* = \frac{T_k - E(T_k)}{\sqrt{\text{Var}(T_k)}} \underset{\text{asympt.}}{\sim} N(0, 1)$$

The standardized linear trend statistic T_k^* of each stratum k , is thus asymptotically standard normal distribution with mean 0 and variance = 1. Hence, the square of the standardized linear trend statistic T_k^* is asymptotic chi-squared distribution with one degree of freedom.

Theorem 4.3.2. *The asymptotic distribution of the statistic across k strata can be summarized by the M^2 statistic.*

$$M^2 = \frac{\left[\sum_{j=1}^k (T_j - E(T_j)) \right]^2}{\sum_{j=1}^k \text{var}(T_j)} \sim \chi_{(1)}^2 \quad (4.2)$$

Proof. :

Let T be the sum of K independent random variables with a from each of K strata

$$T = \sum_{k=1}^K T_k$$

Then $E(T)$ is the sum of expectations of K independent random variables with hypergeometric distribution.

$$E(T) = \sum_{k=1}^K E(T_k)$$

$$\text{Var}(T) = \sum_{k=1}^K \text{Var}(T_k)$$

Then

$$\frac{(T - E(T))^2}{\text{Var}(T)} \overset{\text{asympt.}}{\sim} \chi_{(1)}^2$$

□

For $g \times K$ tables without column ordering but with ordering of the rows, the nonparametric Kruskal Wallis test and generalized one way ANOVA provide the uniformly most powerful invariant, or UMP unbiased against the null hypothesis of no column and row association. When the number of rows $g = 2$, the Kruskal Wallis reduces to the Wilcoxon rank sum test. Both test are asymptotically chi-squared with $g - 1$ degrees of freedom.

The Linear by Linear and Jonckheere-Terpstra test has the highest power for doubly ordered $g \times K$ contingency tables. These test do well in detecting existence of a progressive trend in the alternate hypothesis. The flexibility in arbitrarily selecting scores, makes the linear by Linear test very powerful. A special case in which the original observations are used in place of the weights u_{ij} and w_{rj} results in a significance test for Pearson's correlation coefficient. However, if the weights are replaced by the mid-rank scores of the the observed data, Spearman correlation

coefficient between the ordered responses and ordered treatment groups is obtained [32]. A special case of the linear Trend statistic where there are only 2 dose groups and K strata is the simple linear rank statistic:

$$T_j = \sum_{r=0}^n w_{r,j} A_{r,j}^{(1)}$$

This test statistic T_j is also equal to the Spearman correlation between treatment levels $0, 1, \dots, g$ and the Responses $0, 1, \dots, K$ in stratum k multiplied by $\sqrt{N_k - 1}$. ([1]; [5]). When the observed data are used as weights, then the linear rank test is a significance test of the Pearson correlation coefficient. However, if the ranks of the data are used then the result is a significance test of the Spearman correlation coefficient [32].

Some of the commonly used linear rank weights are listed below:

- Wilcoxon Scores: The Wilcoxon scores are the ranks of the underlying responses. Mid-ranks are used in the case of tied observations.

$$w_{r,k} = n_{0,k} + n_{1,k} + \dots + n_{j-1,k} + \frac{(n_{j,k}+1)}{2}$$

$$u_{ik} = m_k^0 + m_k^1 + \dots + m_k^{j-1} + \frac{m_k^j+1}{2}$$

In the special case where we have only two treatment groups, using the Wilcoxon scores reduces to the rank sum and the Wilcoxon-Mann-Whitney test statistic [5].

- Normal Scores: The normal scores are defined as the percentile of the standard normal distribution.

$$w_{rk} = \frac{1}{n_{r,k}} \left(\sum_{j=v_{r-1,k}+1}^{v_{r,k}} \Phi^{-1} \left(\frac{j}{N_i+1} \right) \right)$$

$$u_{ik} = \frac{1}{m_k^i} \left(\sum_{l=w_{i-1,k}+1}^{w_{i,k}} \Phi^{-1} \left(\frac{l}{N_i+1} \right) \right)$$

where $v_{j,k} = n_{1,k} + n_{2,k} + \dots + n_{j,k}$,

$w_{j,k} = m_k^0 + m_k^1 + \dots + m_k^g$ and $\Phi^{-1}(\alpha)$ is the 100α th percentile of the standard normal distribution.

- Logrank Scores: The logrank scores are the same as the scores for Savage

(exponential scores) when they is no censoring [34]. They are used for survival data.

$$w_{rk} = \frac{1}{n_{r,k}} \left(\sum_{t=v_{r-1,k}+1}^{v_{r,k}} \sum_{l=1}^t \left(\frac{1}{N_i-l+1} \right) \right) - 1$$

$$u_{ik} = \frac{1}{m_k^i} \left(\sum_{t=w_{i-1,k}+1}^{w_{i,k}} \sum_{l=1}^i \left(\frac{1}{N_i-l+1} \right) \right) - 1$$

where $v_{r,k} = n_{0,k} + n_{1,k} + \dots + n_{r,k}$

$$w_{i,k} = m_k^0 + m_k^1 + \dots + m_k^g$$

- Equally Spaced Scores:

$$w_{rk} = (1, \dots, K)$$

$$u_{ik} = (1, \dots, g)$$

The Cochran-Armitage trend test is a special case of equally spaced scores

$w_{j,k} = j - 1$ and $u_{i,k} = i - 1$; when there are only two treatment groups [34].

4.4 Exact Two-sided P-value Computation

Let P_Y be the exact probability distribution of each table Y in the reference set Γ . The distribution of the test statistic can be found empirically by limiting the sample space of each table Y to the reference set Γ and randomly permuting the probability distribution P_Y a large number of times. Define the discrepancy measure \mathcal{D} as a real valued function $\mathcal{D} : \Gamma \rightarrow \mathcal{R}$ mapping $g \times K$ tables in Γ onto the real number line \mathcal{R} . An exact test is formed by ordering the tables in Γ according to some test statistic or discrepancy measure D_Y . The discrepancy measure quantifies the extent of deviation of a table $Y \in \Gamma$ from the null hypothesis of no row by column interaction. Various choices of \mathcal{D} include the Linear Rank, Pearson, Chi-Square, Fisher's Exact and Likelihood Ratio test statistics. The p-value is found as the sum of all null probabilities of all tables Y in Γ which are at least as extreme as the observed table X with respect to \mathcal{D} .

$$\text{p-value} = \sum_{D_Y \geq D_X} P_Y = P_r[D_Y \geq D_X]$$

Classical nonparametric methods rely on the large sample/asymptotic

distribution of \mathcal{D} such as chi-squared distribution to estimate the p-value. With the increased computational power and efficient enumerating algorithms now available, the exact distribution of \mathcal{D} can be found and the p-value subsequently computed. An adjustment for correcting the discreteness that arises in small samples distributions is to make inferences based on the *mid p-value* (Lancaster 1961). This adjustment reduces the ordinary p-value by half the probability of the observed result. The mid p-value is found as:

$$\text{mid P-value} = \frac{1}{2}P_r[D_Y = D_X] + P_r[D_Y > D_X]$$

Classical non-parametric methods rely on the asymptotic distribution of D to approximate the p-value. This approximation can be considerably different from the exact distribution when presented with unbalanced or sparse data or data with a small sample size. Mehta *et al* (1998) suggest that the main advantage of the exact p-value over its asymptotic approximation is that it is guaranteed to bound the type I error rate of the hypothesis testing procedure to any desired level [32].

Table 14: Size of Reference Sets (Mehta, Patel, Senchaudhuri, 1992)

Sample Size(N)	Tables in Reference Set Γ
20	1.8×10^5
30	1.5×10^8
40	1.4×10^{11}
50	1.3×10^{14}
100	1.0×10^{29}

Explicit enumeration of all the tables in the reference set Γ would be computationally expensive and infeasible for relatively large tables. An example of the size of reference sets for different N values is given in Table 14. Mehta, Patel, and Tsiatis (1984) developed a network algorithm that implicitly enumerates all tables in the reference set by connecting a series of networks and nodes. Some of these tables in the reference sets are also rather sparse and not likely to yield accurate p-values. This is the motivation for the Monte Carlo approach. For all

practical purposes, the Monte Carlo exact p-value is equivalent to the exact p-value obtained by enumeration of all tables in the reference set. The expected value of the Monte Carlo p-value is equal to that in which all tables are enumerated.

4.5 Application using Edge Data

The procedure was implemented using the Edge Toxicology Laboratory data available in the R statistical programming language; the data was obtained from a developmental toxicity experiment in which pregnant New Zealand white rabbits exposed to ethylene glycol diethyl ether (EGDE). The effect of ethylene glycol diethyl ether on their fetal development were then studied. In the study, four groups of pregnant does were randomly assigned to dose levels 0, 25, 50, and 100 milligrams per kilogram body weight of EGDE. For each litter and at each dose level, the adverse response used is the combined number of fetal malformation and fetal death. The data are presented in Table 8. The frequency distribution of live fetuses with malformations are grouped into cluster sizes ranging from 2 to 15. The data appears to be relatively sparse with no indication that dose level has impact on cluster size. From the table, for each fixed cluster size, frequency of malformations appears to increase with increasing dose level.

4.5.1 Results

In Tables 15 and 16 we summarize the results of the tests of homogeneity of response rates across all four treatment groups with the linear rank statistic as the discrepancy measure \mathcal{D} . The results for all four dose groups is displayed in Table 15 and individual comparison between dose group 0 and the remaining dose groups is displayed in Table 16, where SO is the stochastic ordering test [46], RS is the Rao-Scott linear trend test [38] and GEE is the Generalized Estimating Equation [50]. The results indicate significant difference in response rates when all four dose groups are considered together. There is significant difference between dose group 100 and the control group when pairwise hypothesis are considered. There is no

significant difference between dose groups (0, 25) and (0, 50). The estimates are based on 10,000 random samples from the reference set Γ . The distribution of the test statistic displayed shows, the test statistic appears to be approximately normally distributed with some discreteness at the peak of the histogram of the sample. Each of the two sets of weights used (midranks and equal spacing) lead to significant outcomes at 0.05 level. The asymptotic approximate distribution of the test statistic using Wilcoxon weights and Equally spaced weights also resulted in significant p-values. The Pearson Chi-Squared family of test statistics are inapplicable because of the level of sparsity in the data. Many of the expected cell frequencies are zero.

$$H_0^j : \Pi_j^0 = \Pi_j^1 = \dots = \Pi_j^g$$

$$H_a^j : \Upsilon_j^0 \leq \Upsilon_j^1 \leq \dots \leq \Upsilon_j^g \quad \text{with at least one strict inequality}$$

Table 15: EDGE Data, Dose groups (0, 25, 50, 100)

Weights	P value	Mid P-value	Asymptotic
	Exact	Exact	P-value
Equal spacing	9.99×10^{-6}	5.0×10^{-6}	1.97×10^{-8}
Midranks	9.99×10^{-6}	5.0×10^{-6}	9.6×10^{-8}

Table 16: First three columns from Aniko Szabo,2010

Dose	SO	RS	GEE	Exact Linear Trend (Equally Spaced Weights)
0,25	0.116	0.536	0.610	0.485
0,50	0.014	0.150	0.272	0.259
0,100	< 0.001	2×10^{-7}	4×10^{-11}	5×10^{-6}

4.5.2 Application using 2,4,5-T Data

The 2,4,5-T Data was obtained from a study conducted at the National Center for Toxicological Research, Food and Drug Administration [10]. Several strains of mice were used in a multiplicative teratology experiment. About 100 pregnant mice in each strain were daily exposed to ordered dose levels 0, 30, 60, 45, 60, 75 and 90 *mg/kg* of the herbicide 2,4,5-trichlorophenoxyacetic acid from day 6 to day 14 of gestation. The number of fetal implantation sites, deaths, resorptions, cleft palate malformations, and fetal weights of the live fetuses were recorded for each pregnant female. The data from the experiment are summarized in Table 18. For each pregnant mouse, s represents the number of implantation sites and t , the number of combined endpoints, is the number of reabsorbed embryos or dead fetuses together with the number of fetuses with cleft palate malformation. Table 18 gives the frequency of each combination of s and t .

Table 17: Frequency Distribution of the Number of Implant(s) and Number of Combined Endpoints(t) Following Exposure to 2,4,5-T (Bowman and George 1995)

Dose	t	Birth Defects No. of live fetuses(s)				Number of Implants																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	21		
0	0	1																				
	1			1																		
	2					1														1		
	3									2	1	7	6	5	2					1		
30	4									1	5	3	1							1		
	0			1	1	2																
	1								2	2	1	2	3	2	3	2	1					
	2									1	1	2	5	5	2	2						
	3									2	2		2	1	2	1						
	4																					
	5												1	1								
	6																					
	7																					
	8																					
	15																					
	45	0	1			1																
		1			1		1															
		2					1															
		3																				
4																						
5																						
6																						
7																						
8																						
9																						
10																						
11																						
12																						
13																						
14																						
18																						
60		0	1																			
		1																				
	2																					
	3																					
	4																					
	5																					
	6																					
	7																					
	8																					
	9																					
	10																					
	11																					
	12																					
	13																					
14																						
75	0																					
	1																					
	2																					
	3																					
	4																					
	5																					
	7																					
	8																					
	9																					
	10																					
	11																					
	12																					
	13																					
	14																					
	90	0	1																			
1		1																				
3																						
4																						
6																						
7																						
10																						
11																						
12																						
13																						
14																						

Table 19: Results of T-245 Data

Weights	P value Exact(Monte Carlo)	Mid P-value Exact(Monte Carlo)	Asymptotic P-value
Midranks	0.000149985	0.0000999	< 0.000001
Equal spacing	0.000149985	0.0000999	< 0.000001
Exact Conditional Likelihood	0.000449955	0.00039996	—

CHAPTER 5

FUTURE RESEARCH: ADJUSTING FOR MULTIPLICITY IN EXACT TESTS

5.1 Introduction

Multiple testing occurs in the analysis of data when several hypotheses are tested simultaneously. For example in genomics, one may be interested in simultaneously testing thousands of hypotheses to determine which genes are differentially expressed. In clinical trials, one may also be interested in simultaneously comparing a control group with different levels of treatment groups. In a single hypothesis test, a level of significance α , is chosen to control the type-I error rate, otherwise known as a false positive. This is the probability of incorrectly rejecting the null hypothesis. When multiple hypotheses are tested simultaneously to make inference on a global hypothesis of interest, the probability of incorrectly declaring significance when no effect exist, is greatly increased. Specifically, suppose n independent hypotheses H_{01}, \dots, H_{0n} are tested each at a significance level α , then the probability of incorrectly rejecting at least one null hypothesis is $1 - (1 - \alpha)^n$. Thus, as n increases, the probability of type-I error increases and tends to one. The probability of making at least one type-I error in a multiple test is called a familywise error rate denoted *FWER*.

5.2 Application with EDGE Data

In a plan to explore the use of various procedures that adjust P - values in the context of multiple exact test we have done a preliminary analysis of the EDGE data. We simultaneously tested for trend between the control group(dose 0) and other dose groups 25, 50 and 100. We implemented the exact linear trend test procedure described in chapter 4 using the EDGE dataset. Table 20 shows the unadjusted p-values. There is no significant difference between dose groups (0, 25), (0, 50) and (25, 50). In Table 21, the p-values are adjusted using the

familywise error rate and false discovery rates approaches for controlling type-I error described earlier in this chapter. As expected the Bonferroni procedure is the most conservative among all the procedures. The Familywise Error control procedures are conservative in comparison to the False Discovery Procedure. The exact linear trend test detects significant difference between dose groups 0 and 100. There is no significant difference between dose groups (0, 25) and (0, 50) even with multiple adjustment correction.

Table 20: Raw p-values with equally spaced scores

Dose	p-value	Mid p-value
0,25	0.5383462	0.49135
0,50	0.3055694	0.26335
0,100	9.999×10^{-5}	5×10^{-5}
25,50	0.3264674	0.27615
25,100	9.999×10^{-5}	5×10^{-5}

Table 21: Adjusted p-values

Dose Group	Raw p-values	Adjusted p-values							q-values
		Holm	Hochberg	Hommel	Bonferroni	Šidák	Benjamini Hochberg (2000)	Benjamini Yekutieli (2001)	
0,25	0.5383462	0.526700	0.491350	0.491350	1.000000	0.8683996	0.491350	0.9008083	0.3275667
0,50	0.3055694	0.526700	0.491350	0.491350	0.790050	0.6002545	0.395025	0.7242125	0.2633500
0,100	0.00005	0.000015	0.000015	0.000015	0.000015	0.000015	0.000015	0.0000275	0.0000100

5.3 Conclusion

In this dissertation, we have introduced an exact test for testing for trend using binary and multinomial exchangeable data for developmental toxicity studies. Our procedure generalized the well-known Fisher's Exact test and the version for correlated data introduced by Cochran *et al* [12]. We implemented an exact conditional test for exchangeable clustered binary data analogous to a generalized Fisher's exact test. We eliminate the nuisance parameters(in this case the probabilities of response) by conditioning on joint sufficient statistics. Conditional tests are known to be conservative, a test of Homogeneity of responses across different treatment groups using the EDGE data of the exact test implemented here compared with the Rao-Scott Homogeneity (asymptotic) test validates this belief. We also implemented an exact linear rank test for correlated binary data. We have conducted two simulation studies to demonstrate the performance of the exact conditional test. The results from the two simulation studies are very close to each other. The simulation study generating random values from the q-power model has slightly larger p-values than that of the Lunn and Davis approach for any set of fixed parameters. We have also shown that an exact linear trend test can be simultaneously performed to test multiple hypotheses and the p-values accordingly adjusted to control for multiplicity errors. Our future work will be to develop an exact conditional stochastic trend test to capture trends which are not necessarily linear like the one developed in this dissertation.

REFERENCES

- [1] Agresti, A. (2013). *Categorical Data Analysis*, Wiley & Sons, New York.
- [2] Agresti, A. (2001). *Exact inference for categorical data: recent advances and continuing controversies*. *Statistics in Medicine*, 20:27092722.
- [3] Agresti, A. (1992). *A Survey of Exact Inference for Contingency Tables*. *Statistical Science*. Vol. 7, No. 1., pp. 131-153.
- [4] Armitage, P. (1955). *Tests for linear trends in proportions and frequencies*. *Biometrics*, vol. 11, 375386.
- [5] Baso, D., Pesarin, F., Salmaso, L. and Solari, A. (2009). *Permutation Test for Stochastic Ordering and ANOVA, Theory and Applications with R* . Lecture Notes in Statistics, Springer, 194.
- [6] Benjamini, Y. and Yekutieli, D. (2001). *The Control of the False Discovery Rate in Multiple Testing under Dependency*. *The Annals of Statistics*, Vol. 29, No. 4 , pp. 1165-1188.
- [7] Bernard, R. (1989). *Multivariate Methods for Clustered Binary Data with More Than One Level of Nesting*. *Journal of the American Statistical Association*. Vol. 84, No. 406, pp. 373- 380.
- [8] Best, J. D. (1999). *Test of Fit and Other Nonparametric Data Analysis*. Doctor of Philosophy Thesis. University of Wollongong Thesis Collection.
- [9] Best, J. D., Rayner, W. C. J. and Thas, O. (2006) , *Nonparametric Analysis of Blocked Ordered Categories Data: Some Examples Revisited*. *Journal of Applied Mathematics and Decision Sciences*. Vol. 2006, Article ID 31089, 19.

- [10] Bowman, D. and George, E. O. (1995). *A Saturated Model for Analyzing Exchangeable Binary Data*. Journal of the American Statistical Association, Vol. 91, No. 436, pp. 1602-1610.
- [11] Bretz, F., Hothorn, T. and Westfall, P. (2011). *Multiple Comparisons using R*. Taylor and Francis.
- [12] Cocoran, C., Ryan, L., Senchaudhuri, P. , Mehta, C., Patel, N. and Molenberghs, G. (2001). *An Exact Trend Test for Correlated Binary Data*. Biometrics 57, 941-948.
- [13] Curtis, D. (2002). *Letters to the Editor*. The American Society of Human Genetics, Am. J. Hum. Genet. 71:439441.
- [14] Davis, S. C. (2002). *Statistical Methods For The Analysis of Repeated Measurements*, Springer Text in Statistics.
- [15] Diaconis, P. (1977). *Finite Forms of de Finetti Theorem on Exchangeability*. Synthese, 36: 271. doi:10.1007/BF00486116
- [16] George, E. O. and Kodell, L. R. (1995). *Tests of Independence, Treatment Heterogeneity, and Dose-Related Trend With Exchangeable Binary Data*. Journal of the American Statistical Association, Vol. 90, No. 431, pp. 871-879.
- [17] Eldridge, M. S., Ukoumunne, C. O., Carlin, B. J. (2009). *The Intra-Cluster Correlation Coefficient in Cluster Randomized Trials: A Review of Definitions*. International Statistical Review, 77, 3, 378394.
- [18] Fisher, R. A. (1925). *Statistical Methods for Research Workers*, Oliver and Boyd, Edinburg.

- [19] Freeman, G. H. and Halton, J.H. (1951). *Note on an Exact Treatment of Contingency Tables : Goodness of Fit and other Significance Problems*. *Biometrika*, 38:141-149.
- [20] Gibbons, D. J. and Chakraborti, S. (2003). *Nonparametric Statistical Inference*. Fourth Edition, Revised and Expanded.
- [21] Hochberg, Y. (1988). *A Sharper Bonferroni Procedure for Multiple Test of Significance*. *Biometrika*. Vol. 75, No. 4 (Dec., 1988), pp. 800-802.
- [22] Holm, S. (1979). *A Simple Sequentially Rejective Multiple Test Procedure*. *Scandinavian Journal of Statistics*, Vol. 6,, pp. 65-70.
- [23] Hommel, G. (1988). *A Stagewise Rejective Multiple Test Procedure Based on a Modified Bonferroni Test*. *Bimetrika*, Vol. 75, pp. 383-386.
- [24] Jan, S. , Jiří, V. and Ivan, S. (2000) ,*On the Life and Work of Zbyněk Šidák (1933-1999)*. *Applications of Mathematics*. Vol. 45, No. 5, 321–336.
- [25] Kruskal, H. W. and Wallis, A. W. (1952). *Use of Ranks in One-Criterion Variance Analysis*. *Journal of the American Statistical Association*. Vol. 47, pp. 583-621.
- [26] Kuk, Y. C. A. (2004). *A Litter-Based Approach to Risk Assessment in Developmental Toxicity Studies via a Power Family of Completely Monotone Functions*. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*. Vol. 53, pp. 369-386.
- [27] Lancaster, H. O. (1961). *Significance Tests in Discrete Distributions*. *Journal of the American Statistical Association*, Vol. 56, No. 294, pp. 223-234

- [28] Leek, T. J. and Storey, D. J. (2008). *A General Framework for Multiple Testing Dependence*. Proceedings of the National Academy of Sciences of the United States of America. Vol. 105, pp. 18718-18723.
- [29] Lunn D. A. and Davies, J. S. (1998). *A Note on Generating Correlated Binary Variables* . Biometrika , Vol. 85, No. 2 , pp. 487-490.
- [30] Luta, G., Koch, G., Cascio, W., and Smith, W. (1996). *Categorical data analysis strategies for a cardiovascular study with small sample size and clustered binary responses*. ASA Proceedings of the Section on Statistical Education,18-27.
- [31] Mantel, N. (1980). *Assessing Laboratory Evidence for Neoplastic Activity*. International Biometric Society. Vol. 36, pp. 381-399.
- [32] Mehta, R. C. (1994), *The Exact Analysis of Contingency Tables in Medical Research*. American Statistical Association. Vol. 1, pp. 21-40
- [33] Mehta, R. C. and Patel, R. N. (1983). *A Network Algorithm for Performing Fisher's Exact Test in $R \times C$ Tables with Given Row and Column Totals*. Journal of the American Statistical Association. Vol. 78, pp. 427-434.
- [34] Mehta, R. C., Patel, N. and Senchaudhuri, P. (1992). *Exact Stratified Linear Rank Tests for Ordered Categorical and Binary Data*. American Statistical Association. Vol. 1, No. 1 (1992), pp. 21-40.
- [35] Mehta, R. C., Patel, N. (2011). *IBM SPSS Exact Test*.
- [36] Patefield, M. W. (1981). *Algorithm AS 159: An Efficient Method of Generating Random $R \times C$ Contingency Tables*. Journal of the Royal Statistical Society. Series C (Applied Statistics), Vol. 30, No. 1 , pp. 91-97.

- [37] Prentice, R. L. and Zhao, L. P. (1991). *Estimating equations for parameters in means and covariances of multivariate discrete and continuous responses*. Biometrics, Vol. 47, No. 3 , pp. 825-839.
- [38] Rao, K. N. J. and Scott, J. A. (1992). *A Simple Method for the Analysis of Clustered Binary Data*. International Biometric Society. Vol. 48, No. 2 , pp. 577-585.
- [39] Rosner, B., Willet, W. C., Spiegelman, D. (1989). *Correction of logistic regression relative risk estimates and confidence intervals for systematic within-person measurement error*. Statistics in Medicine. Vol. 8, pp. 1051-1061.
- [40] Robert, P. C. and Casella, G. (2010). *Introducing Monte Carlo Methods with R*. Springer, New York.
- [41] Ryan, M. L. and Molenberghs, G. (1999). *An Exponential Family Model for Clustered Multivariate Binary Data*. Environmetrics 10,279-300.
- [42] Smyth, K. G. and Phipson, B. (2010). *Permutation P-values Should Never Be Zero: Calculating Exact P-values When Permutations Are Randomly Drawn*. Statistical Applications in Genetics and Molecular Biology. 9, No. 1, Article 39.
- [43] Stefanescu, C., Turnbull, W., B. (2003). *Likelihood Inference for Exchangeable Binary Data with Varying Cluster Sizes*. Biometrics. Vol. 59, 1, pp. 18-24
- [44] Storey, D. J. (2003). *The Positive False Discovery Rate: A Bayesian Interpretation and the q-Value*. The Annals of Statistics. Vol. 31, No. 6, pp. 2013-2035
- [45] Sulewski, P., Motyka, R. (2015). *Power Analysis of Independence Testing For Contingency Tables*. Scientific Journal of Polish Naval Academy. DOI: 10.5604/0860889X.1161260.

- [46] Szabo, A., George, E. O . (2010). *On The Use of Stochastic Ordering to Test for Trend with Clustered Binary Data*. *Biometrika*, vol. 97, pp. 95108.
- [47] Wenjian, Y. (2000). *On Some Exact Statistical Procedures for Analyzing Correlated Binary Data*. Doctor of Philosophy Degree Dissertation. University of Memphis.
- [48] Williams, D. A. (1975). *The Analysis of Binary Responses from Toxicological Experiments Involving Reproduction and Teratogenicity*. *Biometrics*. Vol. 31, No. 4 , pp. 949-952.
- [49] Zbynek, S. (1967). *Rectangular Confidence Regions for the Means of Multivariate Normal Distributions*. *Journal of the American Statistical Association*. Vol. 62, pp. 626-633.
- [50] Zeger, S. L., Liang, Y. K. and Albert, S. P. (1988). *Models for Longitudinal Data: A Generalized Estimating Equation Approach*. *Biometrics*. Vol. 44, pp. 1049-1060.
- [51] Zhen, P., Kuk, Y. C. A. (2007). *Test of Marginal Compatibility and Smoothing Methods for Exchangeable Binary Data with Unequal Cluster Sizes*. *Biometrics*. Vol. 63, pp. 218-227.

Appendices

APPENDIX A
COMPUTER PROGRAMMING CODES

Stratified Linear Exact Test

```
1 rm(list=ls())
2 library(CorrBin)
3 library(vcd)
4 library(coin)
5 data(shelltox)
6 #input data
7 data(egde)
8 edge <- egde # rename dataset because of typo in the code
9
10 shell=(xtabs(Freq~Trt+NResp+ClusterSize , data=shelltox))
11
12 c=shell
13
14 c=xtabs(Freq~Trt+NResp+ClusterSize , data=edge)
15 #c=c[c(1,2) , ,]
16
17
18 dim(c)
19
20
21 ## rank each stratum#####
22
23 ##### Wilxon ranks of nij ,midranks for ties#####
24
25 midrankscores=function(c){
26
27   d=matrix(0,dim(c)[1],dim(c)[2])
28
```

```

29  d[1,] =0.5*(c[1,]+1 )
30
31  d[2,]=c[1,]+0.5*(c[2,]+1)
32
33  for (i in 3:dim(c)[1]) {
34
35
36      d[i,]=colSums(c[1:(i-1),])+0.5*(c[i,]+1)
37
38
39  }
40  return(d)
41 }
42
43
44
45 #####
46 ##### mid rank weights of the column sums
47 ##### in each stratum
48
49 v=midrankscores(apply(colSums(c),2,rank))
50
51 #####
52 ##### mid rank weights of the row sums
53 ##### in each stratum
54
55 u=midrankscores(apply(apply(c,3,rowSums),2,rank))
56
57
58 multiply2=function(c,u,v) {
59     ts=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
60
61     for (k in 1:dim(c)[3]) {

```

```

62
63     ts [, , k]=t(u [, , k]*c [, , k]) *v [, , k]
64
65 }
66
67 tss=sum(ts)
68 return(list(tss))
69 }
70
71
72
73 tobs=multiply2(c,u,v) [[1]]
74
75 ts=c()
76 nsim=100000
77 for (j in 1:nsim){
78
79     t=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
80
81
82
83     for (i in 1:dim(c)[3]){
84
85         t [, , i]=r2dtable(1, rowSums(c [, , i]), colSums(c [, , i])) [[1]]
86
87     }
88     v=midrankscores(apply(colSums(t),2,rank))
89
90     u=midrankscores(apply(apply(t,3,rowSums),2,rank))
91
92     ts [j]=multiply2(t,u,v) [[1]]
93
94

```

```

95 }
96
97
98 p.value1=(length(((ts[ts>=tobs])))+1)/(nsim+1)
99 p.value1
100
101
102 p.value1+p.value2
103 ### mid p-value #####
104 pval=0.5*(length(((ts[ts=tobs])))/(nsim)+(length(((ts[ts>tobs])))/(
      nsim)
105 pval
106
107 # tobss=abs(tobs-mean(ts))
108 # tnn=abs(ts-mean(ts))
109 # twosidedpvalue=(length(((tnn[tobss<=tnn])))+1)/(nsim+1)
110 # twosidedpvalue
111 # midpvalue=(length(((tnn[tobss<=tnn])))/(nsim)+0.5*mean(tobss==tnn)
112 # midpvalue
113 # cbind(tobss,tnn)
114 # which(tobss==tnn)
115 #hist(ts, breaks="fd", col="gray",main="Distribution of test statistics
      using midranks ",xlab="",prob=TRUE,ylab="P(T=t)")
116 #abline(v =tobs, col="blue")
117 #box(lty = 'solid', col = 'black')
118
119 #png("~/Documents/memphisclassesbooks/RESEARCH/Trend/w.png")
120 #hist(ts, breaks="fd", col="gray",main=" ",xlab="",prob=TRUE,ylab="P(T=t
      )")
121 #abline(v =tobs, col="blue")
122 #box(lty = 'solid', col = 'black')
123 #dev.off();
124

```

```

125
126
127
128
129
130
131
132
133
134 x=(apply(colSums(c),2,rank))
135 r=(apply(apply(c,3,rowSums),2,rank))
136 r[1,]=0.5*(r[1,]+1)
137 r[2,]=r[1,]+0.5*(r[2,]+1)
138
139 ##### Wilcoxon ranks of nij, midranks for ties#####
140
141 midrankscores=function(x){
142
143   d=matrix(0,dim(x)[1],dim(x)[2])
144
145   d[1,]=0.5*(x[1,]+1)
146
147   d[2,]=x[1,]+0.5*(x[2,]+1)
148
149   for (i in 3:dim(x)[1]){
150
151
152     d[i,]=colSums(x[1:(i-1),])+0.5*(x[i,]+1)
153
154
155   }
156   return(d)
157 }

```

```

158
159
160 #####
161 ##### mid rank weights of the column sums
162 ##### in each stratum
163
164 v=midrankscores(apply(colSums(c),2,rank))
165
166 #####
167 ##### mid rank weights of the row sums
168 ##### in each stratus
169
170 r=(apply(apply(c,3,rowSums),2,rank))
171 u=matrix(0,dim(r)[1],dim(r)[2])
172 u[1,]=0.5*(r[1,]+1)
173 u[2,]=r[1,]+0.5*(r[2,]+1)
174
175
176
177
178
179 multiply2=function(c,u,v) {
180   ts=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
181
182   for (k in 1:dim(c)[3]) {
183
184     ts[, ,k]=t(u[,k]*c[, ,k])*v[,k]
185
186   }
187
188   tss=sum(ts)
189   return(list(tss))
190 }

```

```

191
192
193
194 tobs=multiply2(c,u,v)[[1]]
195
196 ts=c()
197 nsim=1000
198 for (j in 1:nsim){
199
200   t=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
201
202
203
204   for (i in 1:dim(c)[3]){
205
206     t[, , i]=r2dtable(1, rowSums(c[, , i]), colSums(c[, , i]))[[1]]
207
208   }
209   v=midrankscores(apply(colSums(t),2,rank))
210
211
212   r=(apply(apply(t,3,rowSums),2,rank))
213   u=matrix(0,dim(r)[1],dim(r)[2])
214   u[1,]=0.5*(r[1,]+1)
215   u[2,]=r[1,]+0.5*(r[2,]+1)
216
217
218   ts[j]=multiply2(t,u,v)[[1]]
219
220
221 }
222
223

```



```

224 p.value1=(length(((ts[ts>=tobs])))+1)/(nsim+1)
225 p.value1
226
227
228
229 ### mid p-value #####
230 pval=0.5*(length(((ts[ts=tobs]))))/(nsim)+(length(((ts[ts>tobs]))))/(
      nsim)
231 pval

1 rm(list=ls())
2 library(CorrBin)
3 library(vcd)
4 library(coin)
5 data(shelltox)
6 shelltox=shelltox
7 head(shelltox)
8 #input data
9 data(egde)
10 edge <- egde # rename dataset because of typo in the code
11
12 shell=(xtabs(Freq~Trt+NResp+ClusterSize ,data=shelltox))
13
14 shell1=(xtabs(Freq~Trt+NResp+ClusterSize ,data=shelltox))[c(1,4) , ,]
15
16
17 c=shell
18 c=xtabs(Freq~Trt+NResp+ClusterSize , data=edge)
19 #c=c[c(1,2) , ,]
20 #####Cochran-Armitage Trend Test w=j-1###
21
22 ##### rank of columns ###
23

```

```

24 v=seq(1,dim(c)[2],1)-1
25 u=seq(1,dim(c)[1],1)-1
26
27
28 ## observed test statistic
29 tobs=apply(u*c,3,function(x) t(x))*v
30 tobs=apply(tobs,1,sum)
31 sum(tobs)
32 tob=(apply((apply(u*c,3,function(x) t(x))*v),2,sum))
33 sum(tob)
34 multiply3=function(c,u,v) {
35   ts=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
36
37   for (k in 1:dim(c)[3]){
38
39     ts[, ,k]=t(u[,k]*c[, ,k])*v[,k]
40
41   }
42
43   tss=(ts)
44   return(list(tss))
45 }
46
47
48
49
50
51 #####Asymptotic pvalue of Linear by Linear Association Test#####
52 ##### u=0,1,2,3 and v=0,1,2,3,4,5,6 #####
53 n=colSums(c)
54 m=apply(c,3,rowSums)
55 n=apply(c,3,colSums)
56 N=apply(c,3,sum)

```

```

57
58 v1=matrix(v,dim(c)[2],dim(c)[3])
59
60 u1=matrix(u,dim(c)[1],dim(c)[3])
61
62
63 tobs=apply(multiply3(c,u1,v1)[[1]],3,sum)
64 sum(tobs)
65 mean=colSums(u1*m)*colSums(v1*n)/N
66 sum(mean)
67 var=((colSums((u1^2)*m)-(colSums(u1*m)^2)/N)*(colSums(v1^2*n)-(colSums(
      v1*n)^2)/N))/N
68
69
70 tstar=(sum(tobs-mean)^2)/sum(var)
71
72 pchisq(tstar,df=1,ncp=0,lower.tail=F,log.p=FALSE)
73 pchisq(tstar,df=1,ncp=0,lower.tail=T,log.p=FALSE)
74
75 1-pchisq(tstar,df=1,ncp=0,lower.tail=T,log.p=FALSE)
76
77
78 #####
79 ##### wilcoxon scores#####
80
81 n=colSums(c)
82 m=apply(c,3,rowSums)
83 n=apply(c,3,colSums)
84 N=apply(c,3,sum)
85
86
87 midrankscores=function(c){
88

```

```

89  d=matrix(0,dim(c)[1],dim(c)[2])
90
91  d[1,]=0.5*(c[1,]+1 )
92
93  d[2,]=c[1,]+0.5*(c[2,]+1)
94
95  for (i in 3:dim(c)[1]){
96
97
98      d[i,]=colSums(c[1:(i-1),])+0.5*(c[i,]+1)
99
100
101  }
102  return(d)
103 }
104
105
106 multiply3=function(c,u,v) {
107     ts=array(0,c(dim(c)[1],dim(c)[2],dim(c)[3]))
108
109     for (k in 1:dim(c)[3]){
110
111         ts[,,k]=t(u[,k]*c[, ,k])*v[,k]
112
113     }
114
115     tss=(ts)
116     return(list(tss))
117 }
118
119
120
121

```

```

122
123 v1=midrankscores(apply(colSums(c),2,rank))
124
125 u1=midrankscores(apply(apply(c,3,rowSums),2,rank))
126
127 tobs=apply(multiply3(c,u1,v1)[[1]],3,sum)
128
129 mean=colSums(u1*m)*colSums(v1*n)/N
130
131
132 var=((colSums((u1^2)*m)-(colSums(u1*m)^2)/N)*(colSums(v1^2*n)-(colSums(
      v1*n)^2)/N))/N
133
134
135
136
137 tstar=sum(tobs-mean)^2/sum(var)
138
139 pchisq(tstar, df=1, ncp = 0, lower.tail = F, log.p = FALSE)
140
141 pchisq(tstar, df=1, ncp = 0, lower.tail = T, log.p = FALSE)

```

Simulation of Correlated Binary Data

```

1 set.seed(148)
2 N<-30
3 meanlambda<-5
4 p=.8
5 q=.1
6
7
8 #####
9 ##generating observed variable in a cluster

```

```

10
11 n<-rep(0,N) #size of clusters
12 r<-rep(0,N) #response type one
13
14 for (i in 1:N){
15     r[i]<-0
16
17     n[i]<-rpois(1,meanlambda)
18     Z1<-rbinom(1,1,p)
19
20     for (j in 1:n[i]){
21         Y1<-rbinom(1,1,p)
22
23
24         U<-rbinom(1,1,q)
25         x<-(1-U)*Y1+U*Z1
26         if (x==0) r[i]<-r[i]+1
27
28     }
29     obs1<-cbind(r,n)
30 }
31
32
33
34 obs1
35
36
37
38
39
40
41 #####
42 ##generating observed variable in a cluster

```

```

43 #set.seed(148)
44 N=100
45 n<-rep(0,N) #size of clusters
46 r<-rep(0,N) #response type one
47 treatment=rep(0,N)
48
49 for (i in 1:N){
50
51   #n[i]<-rpois(1,meanlambda)
52   n[i]=sample(seq(1,3))
53   Z1<-rbinom(1,1,p)
54
55   p= runif(n=1, min = 0, max = 0.5)
56   q=2*p
57
58   if(q <=0.25){
59     treatment[i] <- 0
60   } else if(q >0.25& q <=0.6){
61     treatment[i] <- 50
62
63
64   }else {
65     treatment[i] <- 100
66
67   }
68
69
70
71
72
73   x=c()
74   for (j in 1:n[i]){
75     Y1<-rbinom(1,1,p)

```

```

76
77
78     U<-rbinom(1,1,q)
79     x[j]<-(1-U)*Y1+U*Z1
80
81
82   }
83   r[i]<-sum(x)
84   obs<-data.frame(treatment,r,n)
85 }
86
87
88
89   obs
90
91   d= xtabs(~r+n+treatment,data=obs)
92   d
93   apply(d,3,sum)
94
95
96   c=xtabs(~r+treatment+n,data=obs)
97   c=array(d[, ,1],c(4,3,2))
98
99
100
101
102
103 #####
104 ##generating observed variable in a cluster
105 #set.seed(148)
106 N=100
107 n<-rep(0,N) #size of clusters
108 r<-rep(0,N) #response type one

```



```

109 treatment=rep(0,N)
110 meanlambda=3
111
112 for (i in 1:N){
113
114   n[i]<-rpois(1,meanlambda)
115   #n[i]=sample(seq(1,3))
116   Z1<-rbinom(1,1,p)
117
118   p= runif(n=1, min = 0, max = 0.5)
119   q=2*p
120
121   if(q <=0.25){
122     treatment[i] <- 0
123   } else if(q >0.25& q <=0.6){
124     treatment[i] <- 50
125
126
127   }else {
128     treatment[i] <- 100
129
130   }
131
132
133
134
135
136   x=c()
137   for (j in 1:n[i]){
138     Y1<-rbinom(1,1,p)
139
140
141     U<-rbinom(1,1,q)

```

```

142     x[j] <- (1-U)*Y1+U*Z1
143
144
145   }
146   r[i] <- sum(x)
147   obs <- data.frame(treatment, r, n)
148 }
149
150
151
152 obs
153
154 d = xtabs(~r+n+treatment, data=obs)
155 d
156 apply(d, 3, sum)
157
158
159 c = xtabs(~r+treatment+n, data=obs)
160 c
161
162 N <- 30
163 meanlambda <- 5
164 p = .8
165 q = .1
166
167 niter <- 100
168 LAMBDA = matrix(0, niter, 6)
169
170
171
172 # #####
173 # ##generating observed variable in a cluster
174 # N=100

```

```

175 # n<-rep(0,N) #size of clusters
176 # r<-rep(0,N) #response type one
177 # Treatment=rep(0,N)
178 # Frequency=rep(0,N)
179 # meanlambda=1
180 # p=0.2
181 # q=0.5
182 # for (i in 1:N){
183 #   r[i]<-0
184 #
185 #   #n[i]<-rpois(1,meanlambda)
186 #   n[i]<-sample(1:5,1,replace=T)
187 #   Z1<-rbinom(1,1,p)
188 #
189 #   for (j in 1:n[i]){
190 #     Y1<-rbinom(1,1,p)
191 #
192 #
193 #     U<-rbinom(1,1,q)
194 #     x<-(1-U)*Y1+U*Z1
195 #     if (x==0) r[i]<-r[i]+1
196 #
197 #   }
198 #   Frequency= sample(0:5,N,replace=T)
199 #   obs1<-cbind(r,n,Frequency)
200 # }
201 #
202 #
203 #
204 # c1=xtabs(Frequency ~ r+n, data=obs2)
205 # #c1=xtabs(~ r+n, data=obs1)
206 # c1
207 #

```

```

208 #
209 # #####
210 # ##generating observed variable in a cluster
211 # N=100
212 # n<-rep(0,N) #size of clusters
213 # r<-rep(0,N) #response type one
214 # Treatment=rep(0,N)
215 # Frequency=rep(0,N)
216 # sample(0:1,N,replace=T)
217 # meanlambda=1
218 # p=0.2
219 # q=0.5
220 # x=c()
221 # for (i in 1:N){
222 #   #r[i]<-0
223 #   r=c()
224 #   # n[i]<-rpois(1,meanlambda)
225 #   n[i]<-sample(1:5,1,replace=T)
226 #
227 #   Z1<-rbinom(1,1,p)
228 #
229 #   for (j in 1:n[i]){
230 #     Y1<-rbinom(1,1,p)
231 #
232 #
233 #     U<-rbinom(1,1,q)
234 #
235 #     x[j]<-(1-U)*Y1+U*Z1
236 #     #if (x==0) r[i]<-r[i]+1
237 #
238 #
239 #   }
240 #   r[i]=sum(x)

```

```

241 #     Frequency= sample(0:5 ,N, replace=T)
242 #     obs2<-cbind(r,n, Frequency)
243 #   }
244 #
245 #
246 #
247 #   c2= xtabs(Frequency~r+n, data=obs2)
248 #   #c2=xtabs(~r+n, data=obs2)
249 #   c2
250 #
251 # c= array(c(c1 , c2) ,c(dim(c2) [1] ,dim(c2) [2] ,2) )
252 # c
253 apply(c ,3 , fisher . test (c ,B=2000 ,workspace=2e+10))
254
255 a=matrix(0 ,2 ,2)
256 b=matrix(c(1 ,2 ,3 ,4) ,2 ,2)
257 array(c(a,b) ,c(2 ,2 ,2) )
258
259 set . seed (3486)
260 library (CorrBin)
261 library (lattice)
262 ss <- expand . grid (Trt=0:2, ClusterSize=1:10, Freq=5)
263 #Trt is converted to a factor
264 rd <- ran . CBDData(ss , p . gen . fun=function (g) 0.2+0.1 *g)
265 rd <- ran . CBDData(ss , p . gen . fun=function (g) 0.1 *g)
266 rd
267 c=xtabs(Freq~Trt+NResp+ClusterSize , data=rd)
268
269 ss$ClusterSize [1]
270 cs <- ss$ClusterSize [1]
271 trt <- unclass (ss$Trt) [1]
272 n <- ss$Freq [1]
273 p <- p . gen . fun (trt)

```

```

274 rho <- rho.gen.fun(trt)
275 probs <- pdf.fun(p, rho, cs)
276
277 r1= rmultinom(1,size = 17, prob = c(0.7,0.7,0.7,0.7,0.7))
278 r2=rmultinom(1, size = 10, prob = c(0.5,0.5,0.5,0.5,0.5))
279 r3=rmultinom(1, size = 10, prob = c(0.1,0.1,0.1,0.1,0.1))
280 apply(cbind(r1,r2,r3),2,sum)
281 pr <- c(1,3,6,10) # normalization not necessary for generation
282 rmultinom(10, 20, prob = pr)
283
284
285 rm(list=ls())
286 #set.seed(148)
287 N<-50
288 meanlambda<-10
289 p=.1
290 q=.3
291 n=c()
292 z=c()
293 x=c()
294 y=c()
295 u=c()
296 r=c()
297 s=c()
298 c=c()
299 for (j in 1:N){
300   y<-rbinom(1,1,p)
301   u<-rbinom(1,1,p)
302   n[j]=rpois(1,meanlambda)
303   #n[j]= sample(1:10,1,replace=T)
304   #g=NULL
305   g=c()
306

```

```

307   for (i in 1:n[j]) {
308     z<-rbinom(1,1,p)
309     x =(1-u) *y+u*z
310
311     g=c(g,x)
312     r[j] =sum(g)
313     c[j]=length(g)
314   }
315
316 }
317
318 obs1=cbind(r=r,n=c,t=rep(1,length(r)))
319
320 #xtabs(~r+n, data=obs1)
321 p=.5
322 q=.3
323 n=c()
324 z=c()
325 x=c()
326 y=c()
327 u=c()
328 r=c()
329 s=c()
330 c=c()
331 for (j in 1:N){
332   y<-rbinom(1,1,p)
333   u<-rbinom(1,1,p)
334   n[j]=rpois(1,meanlambda)
335   #n[j]= sample(1:10,1,replace=T)
336   #g=NULL
337   g=c()
338
339   for (i in 1:n[j]) {

```

```

340     z<-rbinom(1,1,p)
341     x =(1-u) *y+u*z
342
343     g=c(g,x)
344     r[j] =sum(g)
345     c[j]=length(g)
346   }
347
348 }
349
350 obs2=cbind(r=r,n=c,t=rep(2,length(r)))
351
352 obs= rbind(obs1,obs2)
353
354 c= xtabs(~t+r+n, data=obs)
355 c
356
357
358
359
360
361
362
363 s.pvalue=c()
364 for (i in 1:10) {
365   N<-50
366   meanlambda<-10
367   p1=.1
368   q1=.3
369   n=c()
370   z=c()
371   x=c()
372   y=c()

```



```

373  u=c()
374  r=c()
375  s=c()
376  c=c()
377  for (j in 1:N){
378    y<-rbinom(1,1,p1)
379    u<-rbinom(1,1,p1)
380    n[j]=rpois(1,meanlambda)
381
382    g=c()
383
384    for(i in 1:n[j]){
385      z<-rbinom(1,1,p1)
386      x =(1-u) *y+u*z
387
388      g=c(g,x)
389      r[j] =sum(g)
390      c[j]=length(g)
391    }
392
393  }
394
395  obs1=cbind(r=r,n=c,t=rep(1,length(r)))
396
397  p2=.5
398  q2=.3
399  n=c()
400  z=c()
401  x=c()
402  y=c()
403  u=c()
404  r=c()
405  s=c()

```

```

406   c=c()
407   for (j in 1:N){
408     y<-rbinom(1,1,p2)
409     u<-rbinom(1,1,p2)
410     n[j]=rpois(1,meanlambda)
411
412     g=c()
413
414     for(i in 1:n[j]){
415       z<-rbinom(1,1,p2)
416       x =(1-u) *y+u*z
417
418       g=c(g,x)
419       r[j] =sum(g)
420       c[j]=length(g)
421     }
422
423   }
424
425   obs2=cbind(r=r,n=c,t=rep(2,length(r)))
426
427   obs= rbind(obs1,obs2)
428
429   c= xtabs(~t+r+n, data=obs)
430   c
431
432   s.pvalue[i]=Homogeneity(c,nsim=1000)
433 }
434
435 s.pvalue
436
437
438

```

```

439
440
441
442 p=.5
443 q=.3
444
445 rm(list=ls())
446 genfunc=function(p,q,meanlambda,N){
447
448
449   #
450   # N<-30
451   # meanlambda<-5
452   # p=.8
453   # q=.1
454
455
456   #####
457   ##generating observed variable in a cluster
458
459   n<-rep(0,N) #size of clusters
460   r<-rep(0,N) #response type one
461
462   for (i in 1:N){
463     r[i]<-0
464
465     n[i]<-rpois(1,meanlambda)
466     Z1<-rbinom(1,1,p)
467
468     for (j in 1:n[i]){
469       Y1<-rbinom(1,1,p)
470
471

```

```

472     U<-rbinom(1,1,q)
473     x<-(1-U)*Y1+U*Z1
474     if (x==0) r[i]<-r[i]+1
475
476   }
477   obs1<-cbind(r,n)
478 }
479
480
481
482   obs1
483
484   return(obs1)
485
486 }
487
488
489   genfunc(p=0.1,q=0.2,meanlambda=5,N=30)
490
491   obs1=cbind(genfunc(p=0.1,q=0.2,meanlambda=5,N=30),Trt=rep(1,N=30))
492   obs2=cbind(genfunc(p=0.1,q=0.2,meanlambda=5,N=30),Trt=rep(2,N=30))
493
494
495
496   obs= data.frame(rbind(obs1,obs2))
497
498   c= xtabs(~Trt+r+n, data=obs)
499   c
500
501
502
503
504

```

```

505
506 s.pvalue=c()
507
508 for (i in 1:1000){
509
510     obs1=cbind(genfunc(p=0.1,q=0.9,meanlambda=10,N=50),Trt=rep(1,N=50))
511     obs2=cbind(genfunc(p=0.9,q=0.9,meanlambda=10,N=50),Trt=rep(2,N=50))
512
513
514
515     obs= data.frame(rbind(obs1,obs2))
516
517     c= xtabs(~Trt+r+n, data=obs)
518
519
520
521     s.pvalue[i]=Homogeneity(c,nsim=1000)
522
523     avg.pvalue=mean(s.pvalue)
524 }
525 avg.pvalue

```

Simulation of Correlated Binary Data by Inverse Transform of the q-power Distribution

```

1 rm(list=ls())
2 library(CorrBin)
3 data(egde)
4 edge <- egde
5 data(shelltox)
6
7
8

```

```

9 #
#####

10 ##### Generate random variables R with the
#####
11 ##### Inverse transform method from the logistic distribution
#####

12
13 beta1=0.4553394;
14 beta2=0.0042140;
15 dose=c(0,25)
16 #beta1=4.9568
17 #beta2=-5.54101
18 #beta=beta1+beta2*(dose/100);
19 beta=beta1+beta2*(dose);
20
21 logistic.pdf<-function(n, beta){
22   v=c()
23   for(r in 0:n){
24
25     k =0:(n-r)
26
27     v[r+1]=choose(n, r)*sum((-1)^k*choose(n-r, k)*(2/(1+exp((beta)*log(r
+k+1))))))
28
29   }
30   v = pmax(pmin(v,1),0)
31   return(v)
32 }
33
34
35 cdf=function(n, beta){c=c(); c=cumsum(logistic.pdf(n, beta)); return(c) }
36

```

```

37 cdf(50,2)
38
39
40
41 generate.r=function(beta,n){
42
43   c=c(0,cdf(n,beta));
44   u=runif(1);
45
46   for (i in 2:length(c)){
47     if ( (u>=c[i-1])&(u<c[i]) ) {
48       r=(i-2)
49
50
51     }
52   }
53   return(list(n=n,r=r,u=u))
54
55 }
56 generate.r(beta=beta[1],n=5)
57 generate.r(beta=beta[2],n=5)
58
59 s.pvalue=c()
60
61 for (i in 1:100){
62
63
64   n=50
65   s1=sapply(rpois(n,lambda = 5),generate.rq,p=0.1,rho=0.3)
66   s2=sapply(rpois(n,lambda = 5),generate.rq,p=0.1,rho=0.3)
67
68
69   obs1=cbind(r=unlist(s1[2,]),n=unlist(s1[1,]),Trt=rep(1,n))

```

```

70  obs2=cbind(r=unlist(s2[2,]),n=unlist(s2[1,]),Trt=rep(2,n))
71
72
73  obs= data.frame(rbind(obs1,obs2))
74
75  c= xtabs(~Trt+r+n, data=obs)
76
77
78
79
80  s.pvalue[i]=Homogeneity(c,nsim=1000)
81
82  avg.pvalue=mean(s.pvalue)
83 }
84 avg.pvalue
85
86
87 #
      #####
88 ##### Generate random variaavles R with the
      #####
89 ##### Inverse transform method from q power distribution
      #####
90
91 qpower_pdf <- function(p, rho, n){
92   q <- 1-p
93   gamma <- log2(log(q^2+rho*q*(1-q))/log(q))
94   pr <- numeric(n+1)
95   for (r in 0:n){
96     k <- 0:r
97     pr[r+1] <- choose(n,r) * sum((-1)^k * choose(r,k) * q^((n-r+k)^
gamma))

```



```

98   }
99   pr <- pmax(pmin(pr,1),0) #to account for numerical imprecision
100  pr
101  }
102
103
104
105  cdfq=function(p, rho, n){c=c();c=cumsum(qpower_pdf(p,rho,n)); return(c)
    }
106
107  generate.rq=function(p, rho, n){
108
109    c=c(0,cdfq(p, rho, n));
110    u=runif(1);
111
112    for (i in 2:length(c)){
113      if( (u>=c[i-1])&(u<c[i]) ){
114        r=(i-2)
115
116
117      }
118    }
119    return(list(n=n, r=r, u=u))
120
121  }
122
123
124 # generate.rq(p=0.9,rho=0.9,n=5)
125 # cdfq(p=0.9,rho=0.9,n=8)
126 # sapply(rpois(n=5,lambda = 5),generate.rq,p=0.1,rho=-0.9)
127 # p=0.9
128 # q=0.1
129 # rho=0.9

```

```

130 # log2(log(q^2+rho*q*(1-q))/log(q))
131
132
133
134 s.pvalue=c()
135
136 for (i in 1:1000){
137
138
139 n=50
140 s1=sapply(rpois(n,lambda = 5),generate.rq,p=0.1,rho=0.9)
141 s2=sapply(rpois(n,lambda = 5),generate.rq,p=0.9,rho=0.9)
142
143
144 obs1=cbind(r=unlist(s1[2,]),n=unlist(s1[1,]),Trt=rep(1,n))
145 obs2=cbind(r=unlist(s2[2,]),n=unlist(s2[1,]),Trt=rep(2,n))
146
147
148 obs= data.frame(rbind(obs1,obs2))
149
150 c= xtabs(~Trt+r+n, data=obs)
151
152
153
154
155 s.pvalue[i]=Homogeneity(c,nsim=1000)
156
157 avg.pvalue=mean(s.pvalue)
158 }
159 avg.pvalue
160
161
162

```

```

163
164
165
166 #
#####

167 ##### Generate random variaavles R with the
#####
168 ##### Inverse transform method from the Beta-Binomial distribution
#####

169
170
171 betabin_pdf <- function(p, rho, n){
172   a <- p*(1/rho-1)
173   b <- (1-p)*(1/rho-1)
174   k <- 0:n
175   pr <- choose(n, k)*beta(a+k, b+n-k)/beta(a,b)
176
177   return(pr)
178 }
179
180 betabin_pdf(0.2, .3, 5)
181
182 cdfbb=function(p, rho, n){c=c();c=cumsum(betabin_pdf(p,rho,n)); return(c
    ) }
183 cdfbb(p=0.2,rho=0.3,n=8)
184
185 generate.rbb=function(p, rho, n){
186
187   c=c(0,cdfq(p, rho, n));
188   u=runif(1);
189
190   for (i in 2:length(c)){

```

```

191     if ( (u>=c[i-1])&(u<c[i]) ) {
192         r=(i-2)
193
194
195     }
196 }
197 return( list (n=n, r=r, u=u) )
198
199 }
200
201
202 generate.rbb(p=0.2, rho=0.3, n=8)
203
204
205 s.pvalue=c()
206
207 for (i in 1:100){
208
209
210     n=50
211     s1=sapply( rpois(n, lambda = 5), generate.rbb, p=0.1, rho=0.3)
212     s2=sapply( rpois(n, lambda = 5), generate.rbb, p=0.1, rho=0.3)
213
214
215     obs1=cbind(r=unlist(s1[2,]), n=unlist(s1[1,]), Trt=rep(1, n))
216     obs2=cbind(r=unlist(s2[2,]), n=unlist(s2[1,]), Trt=rep(2, n))
217
218
219     obs= data.frame( rbind(obs1, obs2))
220
221     c= xtabs(~Trt+r+n, data=obs)
222
223

```

```

224
225
226   s.pvalue[i]=Homogeneity(c,nsim=1000)
227
228   avg.pvalue=mean(s.pvalue)
229 }
230 avg.pvalue
231
232
233 #
#####
234 ##### plots of intracluster correlation
#####
235 ##### #####
#####
236
237
238 p=c(0.4948332,0.5035944,0.5005315)
239 rho=c(0.3,0.6,0.9)
240 plot(rho,p)

```

Multiple Testing

```

1 alpha=c(0.1,0.05,0.01,0.001)
2 n=100
3 s=seq(1,n,1)
4 p1=1-(1-0.1)^s
5 p2=1-(1-0.05)^s
6 p3=1-(1-0.01)^s
7 p4=1-(1-0.001)^s
8
9 plot(s, p1, lwd=2, col="blue",pch=2)

```

```

10 lines(s, p2, type="l", lwd=2, col="red")
11 lines(s, p3, type="l", lwd=2, col="green")
12 lines(s, p4, type="l", lwd=2, col="pink")
13 # set up graph
14 xrange <- range(s)
15 yrange <- round(range(p))
16 colors <- rainbow(length(alpha))
17 p=cbind(p1,p2,p3,p4)
18
19 plot(xrange, yrange, type="n",
20       xlab="s",
21       ylab="P(at least one Type I Error)")
22 #panel.first=grid()
23 # add power curves
24 opts = c("p", "l", "o", "b")
25
26
27 for (i in 1:length(alpha)){
28   lines(s, p[,i], type=opts[i], lwd=1, col=colors[i], cex=0.5)
29 }
30
31 # add annotation (grid lines, title, legend)
32 abline(v=0, h=seq(0,yrange[2],0.10), lty=2, col="grey89")
33 abline(h=0, v=seq(0,xrange[2],10), lty=2,
34        col="grey89")
35 title("Graph of different type-I error rates levels\n")
36 #legend("topright",1,alpha, title="alpha", opts)
37 legend(xrange[1], yrange[2], alpha, cex=0.8, col=colors,
38        pch=plotchar, lty=linetype, title="alpha")
39
40 #png("~/Documents/memphisclassesbooks/RESEARCH/Trend/mult.png")
41 #matplot(s,p, pch=c(15,16,17,20), col = rainbow(4),ylab="P(at least one
      Type I Error)",xlab="n", lwd=1,cex=0.5,lty=1"4")

```

```

42 linetype <- c(1:length(alpha))
43 plotchar <- seq(17,17+length(alpha)-1,1)
44 opts = c("p","l","o","b")
45 matplot(s,p,lwd=1,lty=linetype,col=colors,pch=plotchar,type=opts,cex
      =0.6,xlab="s",ylab="P(at least one Type I Error)")
46 # add a legend
47 title("Graph of different type-I error rates levels", "")
48 # add a legend
49 legend(xrange[1], yrange[2], alpha, cex=0.8, col=colors,
50        pch=plotchar, lty=linetype, title="alpha")
51
52 # add annotation (grid lines, title, legend)
53 abline(v=0, h=seq(0,yrange[2],0.10), lty=2, col="grey89")
54 abline(h=0, v=seq(0,xrange[2],10), lty=2,
55        col="grey89")
56
57 box(lty = 'solid', col = 'black')
58
59
60
61
62 #png("~/Documents/memphisclassesbooks/RESEARCH/Trend/mult.png")
63 #x11()
64 getwd()
65 png("C:/Users/Gucci148/Desktop/Trend/mult.png")
66 colors <- rainbow(length(alpha))
67 linetype <- c(1:length(alpha))
68 plotchar <- seq(17:20)
69 opts = c("b","p","l","o")
70 plot(xrange, yrange,
71       xlab="s",
72       ylab="P(at least one Type I Error)")
73

```

```

74 # Plot solid circles with solid lines
75 points(s, p1, type=opts[1], pch=plotchar[1], lwd=1, cex=0.5, col=colors[1])
76 # Add open squares with dashed line, with heavier line width
77 points(s, p2, type=opts[2], pch=plotchar[2], lty=2, lwd=1, cex=0.5, col=
      colors[2])
78
79 points(s, p3, type=opts[3], pch=plotchar[3], # Diamond shape
80       lty="dotted", cex=0.5, col=colors[3], # Dotted line,
      double-size shapes
81       lwd=1)
82
83 points(s, p4, type=opts[4], pch=plotchar[4], lty=2, lwd=1, cex=0.5, col=
      colors[4])
84
85 #legend("topright", 8, alpha, fill = rainbow(4))
86 # add annotation (grid lines, title, legend)
87 # add a title and subtitle
88 title("Graph of various type-I error rates", "Number of Test")
89 # add a legend
90 legend(xrange[1]-0.5, yrange[2], alpha, cex=0.8,
91        pch=plotchar, lty=linetype, title=expression(alpha), col=colors)
92
93 abline(v=0, h=seq(0, yrange[2], 0.10), lty=2, col="grey89")
94 abline(h=0, v=seq(0, xrange[2], 10), lty=2,
95        col="grey89")
96
97 # add a legend
98
99 box(lty = 'solid', col = 'black')
100 dev.off()
101
102
103 #####makes you see the plot in r and saves the graph in the

```



```

104 ##### same folder as the r program
105 #dev.copy(jpeg,filename="mult.jpg");
106 #dev.off ();
107
108
109
110 # Create Line Chart
111
112 #png("~/Documents/memphisclassesbooks/RESEARCH/Trend/mult.png")
113 # get the range for the x and y axis
114 xrange <- range(s)
115 yrange <- round(range(p))
116 # set up the plot
117 plot(xrange, yrange, type="n", xlab="s",
118       ylab="P(at least one Type I Error)" )
119 colors <- rainbow(length(alpha))
120 linetype <- c(1:length(alpha))
121 plotchar <- seq(18,18+length(alpha),1)
122 opts = c("p","l","o","b")
123 # add lines
124 for (i in 1:length(alpha)) {
125
126   lines(s, p[,i], type=opts[i], lwd=1.0,
127         lty=linetype[i], col=colors[i], pch=plotchar[i],cex=0.5)
128
129 }
130
131 # add a title and subtitle
132 title("Graph of different type-I error rates levels", "")
133
134 # add a legend
135 legend(xrange[1], yrange[2], alpha, cex=0.8, col=colors,
136        pch=plotchar, lty=linetype, title="alpha")

```

```

137
138
139 # add annotation (grid lines, title, legend)
140 abline(v=0, h=seq(0,yrange[2],0.10), lty=2, col="grey89")
141 abline(h=0, v=seq(0,xrange[2],10), lty=2,
142         col="grey89")
143
144 box(lty = 'solid', col = 'black')
145
146
147
148
149
150
151
152 # Create Line Chart
153
154 # get the range for the x and y axis
155 xrange <- range(s)
156 yrange <- round(range(p))
157 # set up the plot
158 plot(xrange, yrange, type="n", xlab="s",
159       ylab="P(at least one Type I Error)" )
160 colors <- rainbow(length(alpha))
161 linetype <- c(1:length(alpha))
162 plotchar <- seq(18,18+length(alpha),1)
163 opts = c("p", "l", "o", "b")
164 # add lines
165 for (i in 1:length(alpha)) {
166
167     points(s, p[,i], type=opts[i], lwd=1.0,
168           lty=linetype[i], col=colors[i], pch=plotchar[i], cex=0.5)
169

```

```

170 }
171
172 # add a title and subtitle
173 title("Graph of different type-I error rates levels", "")
174
175 # add a legend
176 legend(xrange[1], yrange[2], alpha, cex=0.8, col=colors,
177        pch=plotchar, lty=linetype, title="alpha")
178
179
180 # add annotation (grid lines, title, legend)
181 abline(v=0, h=seq(0, yrange[2], 0.10), lty=2, col="grey89")
182 abline(h=0, v=seq(0, xrange[2], 10), lty=2,
183        col="grey89")
184
185 box(lty = 'solid', col = 'black')
186
187
188
189
190
191 #dev.off();
192
193 matplot(s,p, type="l", col = rainbow(4), ylab="P(at least one Type I Error
194        )", xlab="n", lwd=2, pch=1:4)
195 legend(80, 0.95, as.character(alpha), fill = rainbow(4))
196 # add annotation (grid lines, title, legend)
197 abline(v=0, h=seq(0, yrange[2], 0.10), lty=2, col="grey89")
198 abline(h=0, v=seq(0, xrange[2], 10), lty=2,
199        col="grey89")
200
201 dev.off();
202

```

```

202
203
204 set.seed(123)
205 x <- rnorm(50, mean = c(rep(0, 25), rep(3, 25)))
206 p <- 2*pnorm(sort(-abs(x)))
207 #p=runif(6)
208 p1=p.adjust(p, method = 'fdr', n = length(p))
209 p2=p.adjust(p, method = 'holm', n = length(p))
210 p3=p.adjust(p, method = 'hochberg', n = length(p))
211 p4=p.adjust(p, method = 'hommel', n = length(p))
212 p5=p.adjust(p, method = 'bonferroni', n = length(p))
213 p6=p.adjust(p, method = 'BH', n = length(p))
214 p7=p.adjust(p, method = 'BY', n = length(p))
215 p8=p.adjust(p, method = 'none', n = length(p))
216 pp=cbind(p1,p2,p3,p4,p5,p6,p7,p7)
217 pp
218
219 matplot(p, pp, ylab="p.adjust", type = "l", asp = 1, lty = 1:8,
220         main = "P-value adjustments")
221 legend(0.7, 0.6, c("fdr", "holm", "hochberg", "hommel", "bonferroni", "BH",
222                 "BY", "none"),
223        col = rainbow(8), lty = 1:8, fill = rainbow(8))
224
225 # set up graph
226 xrange <- range(p)
227 yrange <- round(range(pp))
228 # add annotation (grid lines, title, legend)
229 abline(v=0, h=seq(0,yrange[2],0.1), lty=2, col="grey89")
230 abline(h=0, v=seq(0,xrange[2],10), lty=2,
231        col="grey89")
232
233 plot(rank(p), p, lty=2, pch=20, type="l")

```

```

234
235
236
237 p.adjust(p, n = 70)
238
239
240
241 #####
242 ### Q-VALUE #####
243 install.packages("qvalue")
244 library(qvalue)
245 data(hedenfalk)
246 pvalues <- hedenfalk$p
247 qobj <- qvalue(p = pvalues)
248
249 Qvalue(pValues=p, lambda=seq(0, 0.9, 0.05), pi0.method="smoother",
250       fdr.level=NULL, robust=FALSE, smooth.df=3, smooth.log.pi0=FALSE,
251       silent=FALSE)
252
253
254
255 p=c(0.49135,0.26335,0.000005)
256 p.adjust(p, method = p.adjust.methods, n = length(p))
257 p.adj
258 p.adjust.M <- p.adjust.methods[p.adjust.methods != "none"]
259 p.adj <- sapply(p.adjust.M, function(meth) p.adjust(p, meth))
260 p.adj
261
262 plot(rank(p), p, lty=2, pch=20, type="l")
263
264
265 library(BiocInstaller)
266 biocLite()

```

```

267 library(qvalue)
268 data(hedenfalk)
269 pvalues <- hedenfalk$p
270 pvalues=c(0.49135,0.26335,0.000005)
271 qobj <- qvalue(p = pvalues)
272
273
274
275 ## try http:// if https:// URLs are not supported
276 source("https://bioconductor.org/biocLite.R")
277 biocLite()
278 install.packages("mutoss")
279 install.packages("multtest")
280 library(mutoss)
281 library(multtest)
282 install.packages("devtools")
283 library("devtools")
284 install_github("jdstorey/qvalue")
285 p=c(0.49135,0.26335,0.000005)
286 qvalue(p)
287
288 #p=runif(10)
289 p=c(0.49135,0.26335,0.000005)
290 qvalue(p, lambda=seq(0,0.99,0.05), pi0.method="smoother", fdr.level
      =0.05, robust=T, gui=FALSE,
291       smooth.df=3, smooth.log.pi0=T)
292
293
294 #####
295 ##### Adaptive Benjamini Hochberg2001
      #####
296 p=c(0.49135,0.26335,0.000005)
297 adaptiveBH(pValues=p, alpha=0.05, silent=FALSE)

```

```

298
299
300
301
302 #
#####

303 ##### Storey–Taylor–Siegmund’s (2004) adaptive step–up procedure
#####

304
305 ABH=adaptiveSTS(p, alpha=0.05, lambda=0.5, silent=TRUE)
306 ABH
307
308 adjPValuesPlot(adjPValues=ABH$adjPValues, alpha=0.05)
309
310 #
#####

311 ##### Benjamini–Hochbergs Linear Step–Up Procedure
#####

312 BH(pValues=p, alpha=0.05, silent=FALSE)
313
314 #####
315 ##### Benjamini Yekutieli 2001#####
316 p=c(0.49135,0.26335,0.000005)
317 BY(pValues=p, alpha=0.05, silent=FALSE)
318
319
320 hommel(p, alpha=0.05)
321 bonferroni(p, alpha=0.05)
322 holm(p, alpha=0.05)
323 hochberg(p, alpha=0.05)
324 sidak(pValues=p, alpha=0.05, silent=FALSE)

```

```
325 SidakSD(pValues=p, alpha=0.05, silent=FALSE)
326
327
328 install.packages("fdrtool")
329 library(fdrtool)
```