ROBUST TESTS USING WEIGHTED LIKELIHOOD ESTIMATION

By

CHANSOO KIM

Bachelor of Science
Chung-Ang University
Seoul, Korea
1985

Master of Science
Chung-Ang University
Seoul, Korea
1987

Master of Science
Southern Methodist University
Dallas, Texas
1993

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment
of the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
July, 1997

Thesis
1991D
K49r

ROBUST TESTS USING WEIGHTED LIKELIHOOD ESTIMATION

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

ACKNOWLEDGMENTS

I would like to thank and express my most indebted gratitude to my advisor, Dr. Sahadeb Sarkar, for guidance, support, and friendship throughout my tenure at Oklahoma State University. I express my sincere gratitude to Dr. William Warde for his encouragement, kindness and valuable advice. I also thank Dr. Barry Moser for his suggestions, comments and understanding. Thanks is also due to Dr. Kenneth Case, professor of industrial engineering and management, for his wisdom, sense of humor, and valuable time.

A special acknowledgments is given to Dr. Bruce Lindsay at the Pennsylvania State University who kindly sent me his manuscripts when I desperately needed them. Special thanks to Dr. N. Balakrishnan at the McMaster University, Canada and Dr. Rand Wilcox at University of Southern California who compassionately answered to my personal questions. I would also like to express my sincere appreciation to Dr. Ayanedranath Basu at the Indian Institute of Statistics, India, for his valuable suggestions.

I dedicate this work and give deepest thanks from my heart to my father, Chong-shik Kim, and my mother, Bohee Lee, for their endless love, encouragement, patience, and support. My lovely wife, Hyunsook, has been an inspiration throughout all of my educational endeavor, I thank her for her love, assistance, and confidence.

Finally, special thanks are extended to my in-laws for their love and support which I can never repay and to my best friend, Mr. Scott Richter, who generously proofread this dissertation even though he was busy enough to prepare his own.

TABLE OF CONTENTS

Chapter Page

I. INTRODUCTION 1

II. LITERATURE REVIEW 7

2.1 Fisher Information and Large Samples Criteria 7

2.1.1 Fisher's Information Measure 7

2.1.2 Fisher Consistency 8

2.1.3 Efficiency 9

2.1.4 *CAN* Estimators 10

2.1.5 Second Order Efficiency 12

2.2 von Mises Differentiable Statistical Functions 12

2.3 Influence Function 14

2.4 M-estimator 15

2.5 Nonparametric Kernel Density Estimation 17

2.6 Maximum Likelihood Estimation of Censored Data 19

2.7 Modified Maximum Likelihood Estimation 20

2.7.1 *MMLE*s for the Normal 22

2.8 Minimum Hellinger Distance Estimation 29

2.8.1 Properties of the *MHDE* 30

2.8.2 Robustness of the *MHDE* 31

2.9 Hellinger Deviance Test and the Notion of Breakdown 33

Chapter Page

III. MINIMUM DISPARITY ESTIMATION 36
3.1 Discrete Case 36
3.1.1 Disparity Measures 37
3.1.2 Residual Adjustment Functions 39
3.1.3 Blended Families of Disparity Measures 42
3.1.4 Negative Exponential Disparity 43
3.1.5 Robustness of the *MDE*s 44
3.1.6 Second Order Efficiency 48
3.2 Continuous Case 49
3.2.1 The MLE^* and the Likelihood Disparity 53
3.2.2 Transparent Kernel 55
3.2.3 Efficiency of the *MDE*s 57
3.2.4 Negative Exponential Disparity Family 57

IV. WEIGHTED LIKELIHOOD ESTIMATION 62
4.1 The IRLE Method for Continuous Models 63
4.2 The Weighted Likelihood Equation 64
4.3 Weights based on the Pearson Residual 65
4.3.1 The Discrete Case 65
4.3.2 The Continuous Case 67
4.4 Influence Functions and Standard Errors 69
4.5 Robustness of the *WLE* 71
4.6 The Selection of the Smoothing Parameter 75

V. TESTS FOR EQUALITY OF TWO POPULATION MEANS 77
5.1 Introduction 77

Chapter Page

5.2 Alternatives to Student's t Test 79

5.2.1 Behrens-Fisher Problem and Welch's Tests 80

5.2.2 Yuen's Test 83

5.2.3 One-Step M-Estimator and the Bootstrap Confidence Intervals 86

5.3 Tests When Population Variances Are Equal 88

5.3.1 Simulation and Results 92

5.4 Tests When Population Variances Are Unequal 99

5.4.1 Simulation and Results 103

5.5 Concluding Remarks 105

VI. TESTS FOR EQUALITY OF VARIANCES 106

6.1 Introduction 106

6.2 Alternatives to Likelihood Ratio Tests 107

6.3 Simulation and Results 113

6.4 Concluding Remarks 114

VII. TESTS OF EQUALITY OF MORE THAN TWO POPULATION MEANS 115

7.1 Introduction 115

7.2 Alternatives to *ANOVA F* Test 116

7.3 Simulation and Results 126

7.4 Concluding Remarks 128

BIBLIOGRAPHY 129

APPENDIX A: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR THE EQUALITY OF TWO POPULATION MEANS WHEN THEIR VARIANCES ARE EQUAL 149

Chapter Page

APPENDIX B: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF TWO POPULATION MEANS WHEN THEIR VARIANCES ARE UNEQUAL 220

APPENDIX C: MEANS, VARIANCES, AND MEAN SQUARED ERRORS OF THE *NED(a)* ESTIMATES FOR THE POPULATION MEANS 256

APPENDIX D: AVERAGES OF THE *NED(a)* BASED POPULATION VARIANCES ESTIMATES 266

APPENDIX E: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF TWO POPULATION VARIANCES 270

APPENDIX F: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF THREE POPULATION VARIANCES . . . 283

APPENDIX G: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF THREE POPULATION MEANS 300

APPENDIX H: EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF FOUR POPULATION MEANS 313

APPENDIX I: C PROGRAM 1: EQUALITY OF TWO MEANS WITH EQUAL VARIANCES 336

APPENDIX J: C PROGRAM 2: EQUALITY OF TWO MEANS WITH UNEQUAL VARIANCES 361

Chapter Page

APPENDIX K: C PROGRAM 3: EQUALITY OF TWO VARIANCES . . . 387

APPENDIX L: C PROGRAM 4: EQUALITY OF THREE VARIANCES . . 401

APPENDIX M: C PROGRAM 5: EQUALITY OF THREE MEANS 416

APPENDIX N: C PROGRAM 6: EQUALITY OF FOUR MEANS 433

APPENDIX O: C PROGRAM 7: BANDWIDTH SELECTION 450

APPENDIX P: C PROGRAM 8: MSE CALCULATION 459

LIST OF TABLES

Table Page

5.1 Empirical levels ($\mu = 0$) and powers ($\mu = 1$) for the MHDE with various k, where $\hat{h} = k\hat{\sigma}$ and (n_1, n_2)=(10,10), (10, 40), (30, 30). Model is $N(\mu, 1)$. 96

A.1 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. 150

A.2 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. 160

A.3 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. 170

A.4 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. 180

A.5 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. 190

A.6 Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. 200

A.7 Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 8$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. 210

A.8 Empirical levels and powers: Tests for equality of two population means for $n_1 = 20, n_2 = 20$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. 212

Table Page

A.9 Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 40$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. 214

A.10 Empirical levels and powers: Tests for equality of two population means for $n_1 = 8, n_2 = 20$, with square root weight and k=0.9 if $n_i < 30$ for $i = 1, 2$. 216

A.11 Empirical levels and powers: Tests for equality of two population means for $n_i = 40, n_2 = 10$, with square root weights and k=0.9 if $n_1 < 30$ for $i = 1, 2$. 218

B.1 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$. 221

B.2 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$. 226

B.3 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$. 231

B.4 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$. 236

B.5 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$. 241

B.6 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$. 246

B.7 Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$. 251

C.1 Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $N(\mu_i, 1)$, $i = 1, 2$. 257

Table Page

C.2 Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1t(1)$, $i = 1, 2$. 260

C.3 Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1[N(0, 1)/U(0, 1)]$, $i = 1, 2$. 263

D.1 Averages of the $NED(a)$ based population variances estimates. Sample sizes are (n_1, n_2)=(10,10), (30,30) and (50,50). Number of replications is 5000. 267

E.1 Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. 271

E.2 Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. 277

F.1 Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.05$. 284

F.2 Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. 292

G.1 Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.05$. 301

G.2 Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.01$. 304

G.3 Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.05$. 307

G.4 Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.01$. 310

H.1 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.05$. 314

Table Page

H.2 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$. 317

H.3 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.05$. 322

H.4 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.01$. 325

H.5 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$. 328

H.6 Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.01$. 333

LIST OF FIGURES

Figure Page

3.1 The residual adjustment functions of the likelihood disparity (LD), the Hellinger distance (HD) and the negative exponential disparity (NED). 41

3.2 The residual adjustment functions of the family of negative exponential disparities ($NED(a)$) and the likelihood disparity (LD) for $-1 \leq \delta^* \leq 60$. 60

3.3 The residual adjustment functions of the family of negative exponential disparities ($NED(a)$), the likelihood disparity (LD) and the Hellinger distance (HD) for $-1 \leq \delta^* \leq 1$. 61

5.1 Empirical levels of the $MHDE$ by various values of k where $\hat{h} = k\hat{\sigma}$. Sample size: (n_1, n_2)=(10,10), (10,40), (30,30). Model is $N(\mu = 0, 1)$. 97

5.2 Empirical powers ($\mu = 1$) of the $MHDE$ by various values of k where $\hat{h} = k\hat{\sigma}$. Sample size: (n_1, n_2)=(10,10), (10,40), (30,30). Model is $N(\mu = 0, 1)$. 98

NOMENCLATURE

BWHD	**B**lended **W**eight **H**ellinger **D**istance
BWCS	**B**lended **W**eight **C**hi-**S**quared
CAN	**C**onsistent and **A**symptotically **N**ormal
cdf	**C**umulative **D**istribution **F**unction
CRLB	**C**ramér-**R**ao **L**ower **B**ound
CUAN	**C**onsistent and **U**niformly **A**symptotically **N**ormal
HD*	**H**ellinger **D**istance, i.e., $HD^*(f,g) = \|\|f^{1/2} - g^{1/2}\|\|$
HD	(Twice squared) **H**ellinger **D**istance, i.e., $HD(f,g) = 2\|\|f^{1/2} - g^{1/2}\|\|^2$
iid	**I**ndependently and **I**dentically **D**istributed
IRLE	**I**teratively **R**eweighted **L**ikelihood **E**stimation
KL	**K**ullback-**L**eibler divergence
LD	**L**ikelihood **D**isparity
MHD	**M**inimum **H**ellinger **D**istance
MHDE	**M**inimum **H**ellinger **D**istance **E**stimator (Estimate)
MLE	**M**aximum **L**ikelihood **E**stimator (Estimate)
MLE*	Minimum Disparity Version of **M**aximum **L**ikelihood **E**stimator (Estimate)
MML	**M**odified **M**aximum **L**ikelihood
MMLE	**M**odified **M**aximum **L**ikelihood **E**stimator (Estimate)

MML10	**M**odified **M**aximum **L**ikelihood **E**stimator (Estimate) based on **10%** Type II censored data
MNEDE	**M**inimum **N**egative **E**xponential **D**isparity **E**stimator (Estimate)
NCS	**N**eyman's **C**hi-**S**quared
NED	**N**egative **E**xponential **D**isparity
PCS	**P**earson's **C**hi-**S**quared
pdf	**P**robability **D**ensity **F**unction
PWD	**P**ower **W**eighted **D**ivergence
WLE	**W**eighted **L**ikelihood **E**stimator (Estimate)

LIST OF SYMBOLS

$\delta(\cdot)$	Pearson residual
$\delta_n(\cdot)$	Pearson residual in the discrete case
$\delta^*(\cdot)$	Pearson residual in the continuous case
θ	Parameter
Θ	Parameter space
μ	Population mean
ρ	Disparity measure
$\rho_G(\cdot)$	Disparity measure with respect to function G
σ^2	Population variance
Ω	Sample space
$\mathfrak{X}$	Discrete sample space
$A(\delta)$	Residual adjustment function
A_2	Curvature parameter; $A_2 = A''(0)$
C^T	Transpose of the vector or matrix C
$I(\theta)$	Fisher information
$\mathsf{IF}(\cdot)$	Influence function on θ
f_i^*	Nonparametric density estimator for the ith sample
$m_{\theta_i}^*$	Smoothed version of the model with parameter θ_i
$\mathbb{R}^p$	p-dimensional Euclidean space

$T(F)$	Statistical functional defined on distribution function F
$w(\cdot)$	Weight function related to the estimating equation
∇	(Partial) differential operator
$\|\|\cdot\|\|$	L_2-norm

CHAPTER I

INTRODUCTION

The purpose of this dissertation is to introduce some robust test procedures based on recent robust estimators known as weighted likelihood estimators. *Weighted likelihood estimation* (Lindsay, 1994; Basu, Markatou and Lindsay, 1995) is a simple modification of the maximum likelihood method that can be used to obtain efficient and robust estimators. The weighted likelihood approach utilizes the Pearson residuals of the *minimum disparity estimation* procedure (Lindsay, 1994) as weights.

Most of the statistical inferences are based on strong assumptions that the sample observations are *independently* and *identically* distributed, and that the underlying distribution is *normal.* While extremely well-designed experiments may make it possible to satisfy the first assumption, it is the second assumption that is rather unrealistic from a practical view. Geary (1947) wrote for the "new generation of students"

Normality is a myth; there never was, and never will be, a normal distribution.

Geary was not the only one who was aware of the problem of nonnormality at the early stage. Pearson (1931), Box (1953), and Tukey (1960) hinted that nonnormality might be a serious problem, especially in reducing power. The presence of outliers far away from the bulk of the data can easily destroy the assumed model.

For many years, eminent statisticians have been calling to attention the need for robust procedures when practical situation departs from the ideal described by such assumptions. Box (1953) first gave the word "robust" a statistical meaning. According to Stigler (1974), however, the subject of robust inference has a long and varied history. In Stigler's findings, Laplace in 1818, and Sheppard in 1899, are the first to present a large sample theory for one or two order statistics. Newcomb (1878)

provided the first sound, modern approach to robust estimation, including the first use of mixtures of normal densities as representing heavy-tailed distributions. Daniell (1920) gave the first mathematical treatment of the trimmed mean.

Recently, much attention has been given to a class of robust estimators which Huber (1964, 1972, 1977, 1981) calls *M–estimators*; *M* for maximum-likelihood type. Hampel (1968, 1971, 1974) also made major contribution to recent development of robust procedures.

Now, statisticians realize that the presence of the outliers far from the bulk of the data can absolutely ruin the statistical analysis. On the other hand, outliers may provide unique information about the model of interest that is not contained in the other observations. Statistical methodologies for dealing with the possible presence of outliers in data generally fall into two categories: identification and accommodation (Mason, Gunst and Hess, 1989). Identification refers to techniques used to determine whether any outliers exist in a data set and, if so, which observations are outliers. Accommodation of outliers refers to techniques used to mitigate their effects. These techniques include the deletion of outliers, trimming extreme observations to less extreme values, using outlier resistant estimators, respecifying the assumed model, or collecting additional data. Accommodation need not be preceded by the finding of outliers. Often outlier resistant estimators are used without regard to a prior determination that outliers exist.

Although the normality assumption seems like unrealistic, assuming nothing about the underlying distribution is denying the information that such techniques as plotting techniques and goodness-of-fit tests can provide about the shape of the underlying distribution. Thus, it may be rational to assume that the underlying distribution is one of the reasonably wide class of distributions including normal distribution (Tiku, Tan, and Balakrishnan, 1986). Tiku *et al.* defined that a robust procedure to depar-

ture from normality is the one that is almost as efficient as the classical procedure for a normal distribution but is considerably more efficient overall for nonnormal distribution, particularly distributions that constitute plausible alternatives to normality.

Box and Tiao (1964) distinguished two types of robustness: *criterion* robustness and *inference* robustness. In the Neyman-Pearson formulation, a criterion with optimal properties is selected, and the criterion robustness refers to the changes in its probability distribution when the parent distribution deviates from the form assumed. Inference robustness, which is perhaps more important, is concerned with changes in significance level when appropriate changes are made in the nature of the criterion to correspond with the changes in the parent distribution.

In the M-estimation, the robustness is achieved at some sacrifice of the efficiency at the model (Hampel, Rousseeuw and Stahe, 1986). In fact, this trade-off between efficiency and robustness must occur with robust procedures adhering to the influence function approach which was introduced by Hampel (1968). On the other hand, the minimum Hellinger distance estimator has the property of carrying some information about robustness which are not well explained by the influence function. The minimum Hellinger distance method attains first order efficiency, yet has certain robustness properties.

It is Wolfowitz (1953, 1954, 1957) who pioneered minimum distance estimation procedures. As a class of estimators, he established their strong consistency under general assumptions and considered the use of the minimized distance in testing goodness-of-fit.

Bhattacharyya (1943) used a function of the Hellinger distance as a measure of divergence between two populations. Matusita (1955) and Rao (1963) considered some application of the minimum Hellinger distance with multinomial models. Beran (1977a, 1977b) seems to be the first to use the Hellinger distance in continuous

models in the context of robust estimation. Stather (1981) extended Beran's work and considered applications in testing. Tamura and Boos (1986) applied minimum Hellinger distance estimation in the multivariate case. Simpson (1987) has looked at the minimum Hellinger distance for count data. Simpson (1989) also proposed a test based on the minimum Hellinger distance which is more robust than the likelihood ratio test. Donoho and Liu (1988) have looked at the robustness of the minimum distance functional, studying more general classes of distances.

Lindsay (1994) has investigated and identified the practical geometric characteristic of disparities between the data and the model which make the methods robust in the discrete case. Basu and Lindsay (1994) studied minimum disparity estimation in the continuous case. Basu and Sarkar (1994a) considered minimum disparity based goodness-of-fit tests for the multinomial models. Basu and Sarkar (1994b) applied the minimum disparity estimation method to determine robust parameter estimates in the measurement error models. Basu and Sarkar (1994c) extended the use of the minimum disparity tests to the continuous case. Harris and Basu (1994) have considered a general family of distances called the penalized Hellinger distance; this family contains the Hellinger distance and is a function of a parameter h which controls the weight of the empty cells in the distance. Basu, Harris and Basu (1996) proposed tests in a discrete model based on the penalized Hellinger distance. Basu, Sarkar and Basu (1997) applied minimum disparity estimation to test for equality of two population means under the normal model.

The weighted likelihood estimation procedure was initiated by Basu and Lindsay (1994). They proposed the *iteratively reweighted estimating equation* to solve the minimum distance estimating equation. The iteratively reweighted estimating method, which is similar to iteratively reweighted least squares, resolves the numerical difficulties in the Newton-Raphson method when there are a large number of

parameters. Later, Basu, Markatou and Lindsay (1995), and Markatou, Basu and Lindsay (1995) improved the iteratively reweighted estimation equation method and called it the weighted likelihood estimation. Basu and Sarkar (1997) considered an application of weighted likelihood estimating equation to get robust parameter estimates in errors-in-variable models.

In this dissertation, the weighted likelihood estimating equation will be used in conjunction with the minimum disparity estimation procedure. The existing tests, such as the Student's t test for equality of two population means with homogeneous variance, Welch's test for equality of two population means with heteroscedasticity, the Brown-Forsythe test for several means and the Brown-Forsythe test for equality of variances, will be applied using the weighted likelihood estimators. A brief outline of the dissertation is presented below.

In Chapter II we review statistical theories and definitions which we will frequently use in the dissertation, including the definitions and discussion of minimum Hellinger distance estimation (Beran, 1977b), and modified maximum likelihood estimation (Tiku, 1967). In Chapter III we review minimum disparity estimation in both the discrete case (Lindsay, 1994) and the continuous case (Basu and Lindsay, 1994). In Chapter IV weighted likelihood estimation procedures (Basu *et al.*, 1995; Markatou *et al.*, 1995) are reviewed.

Chapter V will discuss the applications of weighted likelihood estimators to the tests of two populations means. When population variances are equal, the Student's t test is the uniformly most powerful test. The Student's t test, however, is not robust against nonnormality and variance heterogeneity. We will propose the weighted likelihood estimator based tests which are robust against nonnormality and/or variance heterogeneity. Tiku's test (1980) is one of the known robust test against the violation of the normality assumption. We will propose a new test based on the weighted

likelihood estimator and compare it to the Tiku's test in terms of sizes and powers under various types of distributions through a Monte Carlo study. On the other hand, when population variances are unequal, Welch's test is the one of the most popular test. Numerous studies have shown that Welch's test is not robust when the underlying distribution is not normal. Recently, several alternative robust tests have been proposed that are more robust than Welch's test. We will construct a new alternative to Welch's test which is more robust than existing robust alternatives, like Tiku and Singh's (1981) test. A new test will be proposed using weighted likelihood estimation procedures. Monte Carlo results will be given to compare the performance of the weighted likelihood estimator based tests to the Tiku-Singh test. Chapter VI will be devoted to developing a test for equality of variances. Brown and Forsythe (1974a) proposed a robust test by modifying the Levene's (1960) test. The Brown-Forsythe test is one of the most recommended tests for equality of variances due to its robustness against nonnormality. We will modify the Levene's test using weighted likelihood estimators which will produce better results than the Brown-Forsythe test in terms of level and power. Finally, in Chapter VII, we will consider a new test based on weighted likelihood estimators, which is a modification of the known robust test, the Brown and Forsythe (1974b) test, for one-way analysis of variance. Comparative studies using simulations also will be presented.

CHAPTER II

LITERATURE REVIEW

In this chapter we will review some statistical theories which we will frequently use in this dissertation. Much of the notation and many of the definitions follow Rao (1973) and Johnson(1988).

2.1 Fisher Information and Large Samples Criteria

2.1.1 Fisher's Information Measure

Let $\mathbf{X}$ be a p–dimensional random vector in space $\Omega \subset \mathbb{R}^p$ and having probability density $P(\cdot, \theta)$ with respect to a σ-finite measure ν. Further assume $P(\cdot, \theta)$ is differentiable with respect to θ. For any measurable set in $\Omega \subset \mathbb{R}^p$ let

$$\frac{d}{d\theta}\int_C P(\mathbf{X}, \theta)d\nu = \int_C \frac{dP(\mathbf{X}, \theta)}{d\theta}d\nu. \tag{2.1}$$

Fisher's information measure on θ contained in the random vector $\mathbf{X}$ is defined by

$$I(\theta) = E_\theta\left[\frac{d\log P}{d\theta}\right]^2 = Var_\theta\left(\frac{d\log P}{d\theta}\right). \tag{2.2}$$

Here are some known properties of the information measure (2.2). For more details see Rao (1973; pp. 329–331).

(i) Let $I_1(\theta)$ and $I_2(\theta)$ be the Fisher information contained in two independent random variables X_1 and X_2 respectively, and $I(\theta)$ be information contained in (X_1, X_2). Then $I(\theta) = I_1(\theta) + I_2(\theta)$. (*Additivity*)

(ii) Let T be a measurable function of X with a density function f_θ with respect

to a σ-finite measure ν, satisfying a condition similar to (2.1). Let $I_T(\theta)$ be the information in T, i.e.,

$$I_T(\theta) = E_\theta\left[\frac{f'_\theta(T,\theta)}{f_\theta(T,\theta)}\right]^2.$$

Then

(a) $E_\theta\left[\dfrac{P'(X,\theta)}{P(X,\theta)}\middle| T=t\right] = \dfrac{f'(t,\theta)}{f(t,\theta)}$,

(b) $I \geq I_T$.

(iii) For a k–dimensional θ, let

$$I_{ij}(\theta) = E_\theta\left[\frac{\partial \log P}{\partial \theta_i}\cdot\frac{\partial \log P}{\partial \theta_j}\right], \qquad i,j = 1,2,\ldots,k.$$

Then the matrix $\mathbf{I}(\theta) = ((I_{ij}))$ is defined to be the information matrix.

We define the *score function* to be the first derivative of the log likelihood with respect to θ and denote it as $u(X,\theta)$=$d\log P(X,\theta)/d\theta$. Hence

$$I(\theta) = E_\theta\big[U(X,\theta)\big]^2 = Var_\theta\big(u(X,\theta)\big) \tag{2.3}$$

and for the T defined in (ii)

$$I_T(\theta) = Var_\theta\big(u(T,\theta)\big). \tag{2.4}$$

2.1.2 Fisher Consistency

Suppose $\{x_n\}$, $n = 1,2,\ldots$ is a sequence of observations and let $\{T_n\}$ be a sequence of estimators for a parameter θ. The estimator T_n is said to be *weakly consistent* for θ if $T_n \to \theta$ in probability and said to be *strongly consistent* if $T_n \to \theta$ with probability

1 (or almost surely). Consistency refers only to a limiting property of the estimator and is usually considered a minimal requirement for an inference procedure. The criterion of consistency is not valid with reference to a particular sample size. It only tells that with increase in sample size the estimator tends to the true value of the parameter.

Fisher introduced another definition of consistency. The definition of Fisher is applicable to any sample size and hence may be more meaningful in practice.

Definition 1. Let $x_1, x_2, \ldots, x_n$ be iid observations on a random variable with distribution function $F_\theta(\cdot)$ and F_n be the empirical distribution based on the observations. Then an estimator T_n is said to be *Fisher consistent* if and only if:

1. T is a statistical functional on the space of distribution functions and $T(F_n)$ is an estimate of parametric function $g(\theta)$ based on the observations, and

2. $T\Big(F_\theta(\cdot)\Big) = g(\theta)$ for all $\theta \in \Theta$, where Θ is a parameter space.

Despite the fact that a functional $T(F)$ is defined for a large nonparametric class of distributions, we will usually be interested in its behavior for F in the neighborhood of an F_θ belonging to a parametric model $\{F_\theta; \theta \in \Theta\}$.

2.1.3 Efficiency

If T_n is any unbiased estimator of a parametric function $g(\theta)$ based on a random sample of size n, then

$$Var_\theta(T_n) \geq \frac{\Big[g(\theta)\Big]^2}{I_T(\theta)}. \tag{2.5}$$

The quantity on the right hand side of the inequality is called the *Cramér-Rao lower bound* (*CRLB*) for T_n. When the sample size is fixed, the *(Cramér-Rao) efficiency*

of an unbiased estimator T_n can be defined as

$$eff(T_n) = \frac{CRLB}{Var_\theta(T_n)}, \qquad 0 \leq eff(T_n) \leq 1. \tag{2.6}$$

This approach is not completely satisfactory because the lack of efficiency may be caused by the bound rather than the estimator.

2.1.4 *CAN* Estimators

Suppose that

$$\sqrt{n}\big(T_n - g(\theta)\big) \xrightarrow{d} N\big[0, v(\theta)\big] \qquad \text{as} \qquad n \to \infty \tag{2.7}$$

for some $0 < v(\theta) < \infty$. It is clear T_n is consistent. If (2.7) holds we say that T_n is *consistent and asymptotically normal* (*CAN*). Many other estimators share this same limit, for example, if $\{T_n^*\}$ satisfies $\sqrt{n}(T_n^* - T_n) \to 0$ in probability, by Slutsky's theorem T_n^* is also *CAN*.

The *CRLB* continues to play a role in the study of asymptotic efficiency. Relation (2.7) suggests that $Var_\theta(T_n)$ is approximately $v(\theta)/n$. In fact subject to some general regularity conditions, it can be shown that

$$v(\theta) \geq \frac{[g'(\theta)]^2}{I(\theta)}$$

where $I(\theta)$ is the information in a single observation. This bound is attained by the maximum likelihood estimators.

The *asymptotic efficiency (as.eff)* of any *CAN* estimator T_n of $g(\theta)$ is defined by

$$as.eff(T_n) = \frac{[g'(\theta)]^2/I(\theta)}{v(\theta)}$$

and *CAN* estimator T_n is said to be asymptotically efficient if

$$v(\theta) = \frac{[g'(\theta)]^2}{I(\theta)}.$$

CRLB was thought to produce a foolproof measure of asymptotic efficiency until a startling example was produced by Hodges; a *super efficient estimator.* When the asymptotic variance is smaller than the information bound, we say that the estimator is super efficient. The existence of a super efficient estimator is troublesome. Rao (1963) takes an alternative approach of ruling out super efficient estimator by requiring that the convergence to normality be uniform in θ.

Let P_θ be the density function of the random variable $X_1, X_2, \ldots, X_n$ and define the the random variable Z_n

$$Z_n = \frac{1}{n}\left(\frac{d \log P_\theta}{d\theta}\right).$$

Rao (1963) defines that a consistent estimator T_n of θ is *first order efficient* if

$$\sqrt{n}|T_n - \theta - \gamma(\theta)Z_n| \to 0 \tag{2.8}$$

in probability or with probability 1, where $\gamma(\theta)$ is a function of θ only.

A *CAN* estimator which converges to normal uniformly in θ is said to be a *consistent and uniformly asymptotically normal* (*CUAN*) estimator. Rao (1963) shows that the asymptotic variance $v(\theta)$ of *CUAN* estimator has the value $[I(\theta)]^{-1}$ where $I(\theta)$ is the information on θ in a single observation. Maximum likelihood estimator of θ is first order efficient under some regularity conditions.

2.1.5 Second Order Efficiency

For different estimation methods that provide reasonably good estimators in the sense that, under suitable regularity conditions, they are all consistent and first order efficient according to criterion (2.8), Rao (1963) introduced *second order efficiency* to distinguish between these different estimators. The concept of second order efficiency is not as easy as first order efficiency. In this theory, the second order efficiency E_2 of an estimator T_n is measured by finding the minimum asymptotic variance of

$$U(\theta) - a(\theta)[T_n - \theta] - \gamma(\theta)[T_n - \theta]^2$$

over a and γ, where U is the score function for an independently and identically distributed (iid) random sample and a and γ are constants possibly depending on θ. The smaller the values of E_2, the greater the second order efficiency. Under some regularity conditions, in the special case of a multinomial distribution, the maximum likelihood estimator has the maximum second order efficiency.

2.2 von Mises Differentiable Statistical Functions

Suppose $X_1, X_2, \ldots, X_n$ is a random sample from a population with distribution function F and let T_n be a statistic which is a function of the sample. Let F_n and f_n be the empirical distribution function and empirical density function respectively. Statistics which can be represented as functionals $T(F_n)$ of the sample distribution F_n are called *statistical functions or functionals.* For example, the mean functional is defined by

$$\mu = T(F) = \int t dF(t)$$

for all distributions with finite first moment. The estimator corresponding to this functional is then

$$\hat{\mu} = T(F_n) = \int t dF_n(t) = \sum_{i=1}^{n} \frac{X_i}{n} = \bar{X}.$$

Theoretical investigation of statistical functionals as a class was initiated by von Mises (1947), who developed an approach for deriving the asymptotic distribution theory of such statistics. The notion of the differentiability of T is a key role in the statistical functional approach.

Many important statistics can be represented as a function of F_n, i.e., $T(F_n)$. Since F_n is a reasonable estimate of F in the sense of various convergence properties, we are expecting $T(F_n)$ to converge to $T(F)$ in a similar fashion, provided the functional $T(\cdot)$ is sufficiently well-behaved in a neighborhood of F. In this way we can consider F as a point in a class $\mathcal{F}$ of distribution functions and thus consider the notions of continuity, differentiability, and other regularity properties for functional the $T(\cdot)$ defined on $\mathcal{F}$. In this context von Mises (1947) introduces a Taylor expansion for $T(\cdot)$ to investigate asymptotic distributions of statistical functionals.

For more detailed explanations and examples of statistical functionals, Serfling (1980), and Randles and Wolfe (1979) are good references.

2.3 Influence Function

Given two points F and G in the space $\mathcal{F}$ of all distribution functions, the set of distribution functions $\mathcal{F}_t = \{F_t : F_t = (1-t)F + tG, 0 \leq t \leq 1\}$ represents the line segments in $\mathcal{F}$ joining F and G. Consider a functional T defined on $\mathcal{F}_t$ for all sufficiently small t. The *Gâteaux differential of T at F in the direction of G* is the limit

$$\lim_{t \to 0^+} \frac{T\big(F + t(F-G)\big) - T(F)}{t} = \int a_1(x) dG(x) \tag{2.9}$$

if there exists a real valued function a_1. By putting $G = F$, hence $\int a_1(x) dF(x) = 0$, in (2.9), we may rewrite it as

$$\frac{\partial}{\partial t}\Big[T\big((1-t)F + tG\big)\Big]_{t=0} = \int a_1(x) d(G-F). \tag{2.10}$$

Now, let $G = \Delta_x$, where Δ_x is a degenerated distribution function which puts mass 1 at the point x, and $\Delta \in \mathcal{F}$. The *influence curve or function* (IF) of T at F is given by

$$\mathsf{IF}(x; T, F) = \lim_{t \downarrow 0} \frac{T\big((1-t)F + t\Delta_x\big) - T(F)}{t}. \tag{2.11}$$

The influence function (IF) was introduced by Hampel (1968, 1974, see also Hampel *et al.*, 1986) in order to investigate the infinitesimal behavior of real valued functional. The above quantity, which is a function of x, allows us to look at what happens if we add one more observation with value x to a very large sample. That is, if we replace F by $F_{n-1} \approx F$ and put $t = 1/n$, the IF measures approximately n times the change of T caused by an additional observation in x when T is applied to a large sample of

size $n-1$.

2.4 *M*-estimator

Consider a functional T defined on distribution function F and $T(F)$ being defined as a solution t_0 of the equation

$$\int \psi(x, t_0) dF(x) = 0 \tag{2.12}$$

for any function $\psi(x,t)$. Such a function $T(\cdot)$ is called an M-functional corresponding to ψ. Typically, the equation (2.12) corresponds to minimization of the quantity

$$\int \rho(x, t_0) dF(x)$$

where

$$\psi(x,t) = c\frac{\partial}{\partial t}\rho(x,t)$$

for some constant c.

The M-estimator corresponding to ψ is a statistical functional $T(F_n)$, that is, a solution T_n of the equation

$$\sum \psi(X_i, T_n) = 0. \tag{2.13}$$

By putting $\rho(x, T_n) = -\log f_{T_n}(x)$ we can get the ordinary *MLE*. The idea of M-estimator was proposed by Huber(1964).

If G_n is the empirical cdf from the sample, then the solution T_n of (2.13) can be

written as $T(G_n)$, where T is the functional given as in (2.12), i.e.,

$$\int \psi(x, T(G))dG(x) = 0 \tag{2.14}$$

for all distributions G. Replace G by $F_t = (1-t)F + t\Delta_x$ and differentiate with respect to t, then

$$\int \psi(y, T(F))d(\Delta_x - F) + \int \frac{\partial}{\partial\theta}\Big[\psi(y,\theta)\Big]_{T(F)} dF(x) \cdot \frac{\partial}{\partial t}\Big[T(F_t)\Big]_{t=0}. \tag{2.15}$$

By using (2.11) we can represent the IF of the M-estimator as

$$\mathsf{IF}(x;\psi,F) = \frac{\psi(x,T(F))}{-\int(\partial/\partial\theta)[\psi(y,\theta)]_{T(F)}\, dF(y)} \tag{2.16}$$

provided the denominator is nonzero.

Now, consider some distribution G in the neighborhood of F, then the first order von Mises expansion of T at F evaluated in G is given by

$$T(G) - T(F) = \int \mathsf{IF}(x, T, F)d(G-F)(x) + \textit{remainder}. \tag{2.17}$$

Let us now look at the relation between the IF and the asymptotic variance. For the iid random variables $X_1, X_2, \ldots, X_n$ from F, $F_n \to F$ uniformly by the Glivenko-Cantelli theorem (See Ash, 1972). Therefore, we may replace G with F_n in (2.17) for large enough n. Furthermore, assume that $T_n(X_1, X_2, \ldots, X_n) = T_n(F_n)$ may be reasonably approximated by $T(F_n)$. Since $\int \mathsf{IF}(x;T,F)dF(x) = 0$, we obtain

$$T_n(F_n) \simeq T(F) + \int \mathsf{IF}(x;T,F)dF_n(x) + \textit{remainder} \tag{2.18}$$

and can rewrite as

$$\sqrt{n}(T_n - T(F_n)) \simeq \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \mathsf{IF}(X_i, T, F) + \textit{remainder}. \tag{2.19}$$

The first term on the right hand side is asymptotically normal by the central limit theorem and the remainder, in most cases, becomes negligible as $n \to \infty$. Hence, $\sqrt{n}[T_n - T(F)]$ has an asymptotic normal distribution with mean 0 and variance $v(T, F)$, where the asymptotic variance equals

$$v(T, F) = \int \mathsf{IF}(x; T, F)^2 \, dF(x). \tag{2.20}$$

Hence, for the M-estimator, using (2.16),

$$v(T, F) = \frac{\int \psi^2(x, T(F)) dF(x)}{\left[\int (\partial/\partial\theta)\psi(x, T(F)) dF(y)\right]^2}. \tag{2.21}$$

The asymptotic variance of the M–estimator is usually higher than the Cramer-Rao lower bound, and thus fails to be first order efficient in that case.

2.5 Nonparametric Kernel Density Estimation

One of the fundamental problems in statistics is developing models based on sample data so that further analyses can be done with statistical techniques. A possible disadvantage of parametric models is that they may not be robust against slightly contaminated data and may lead to erroneous conclusions. Rosenblatt (1956) first explored the world of nonparametric density estimation. Since the appearance of Ronsenblatt's paper, several methods haven been developed. We will briefly discuss only the kernel density estimation method. For those who are interested in other methods, Silverman (1986), Prakasa Rao (1983), Devroye and Györfi (1985), and

Simonoff (1996) are good references.

The most widely used and oldest density estimator is the *histogram.* The discontinuity of histogram causes extreme difficulty if derivatives of the estimates are required. Next consider the *naive estimator.* The probability density function f can be represented as a limit

$$f(x) = \lim_{h \to 0} \frac{1}{2h} P(x - h < X < x + h).$$

Now, we can estimate $f(x)$ from a sample of size n by choosing very small h and setting

$$\hat{f}(x) = \frac{1}{2nh} P[\text{number of } X_1, X_2, \ldots, X_n \text{ falling in } (x - h, x + h)].$$

This estimator is called the naive estimator. We can rewrite this estimator with a weight function. Define the weight function w by

$$w(x) = \begin{cases} \frac{1}{2} & \text{if } |x| < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h} w\left(\frac{x - X_i}{h}\right).$$

The kernel density estimation is a generalization of the naive estimator by replacing the weight function w by a *kernel function* k which satisfies

$$\int_{-\infty}^{\infty} k(x) dx = 1.$$

Usually, k is a symmetric probability density function. By analogy with the definition of the naive estimator, the *kernel density estimator* is defined by

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x - X_i}{h}\right)$$

or

$$\hat{f}(x) = \int \frac{1}{h} k\left(\frac{x - y}{h}\right) dF_n(y)$$

where h is the *smoothing parameter*, also called the bandwidth or the window width.

The kernel estimator is a sum of "bumps" placed at the observations. The kernel function k determines the shape of the bumps and the bandwidth h determines their width. The advantages of the kernel density estimator are: (i) $\hat{f}$ will itself be a pdf, and (ii) $\hat{f}$ will inherit all the continuity and differentiability properties of the kernel k. In density estimation the crucial problem is to choose the smoothing parameter suitably (Hall, 1983; Hall *et al.*, 1995).

2.6 Maximum Likelihood Estimation of Censored Data

Studies on the maximum likelihood estimation method for truncated or censored samples from normal population have been made by several statisticians, notably Cohen (1950, 1957, 1959, 1961), Cohen and Woodward (1953), and Gupta (1952). Cohen (1950, 1957, 1959) derived maximum likelihood estimators (*MLE*s) of singly truncated, Type I singly censored, and Type II singly censored samples from normal distributions, and also derived *MLE*s of doubly truncated or doubly censored samples from normal population. Our concern in this dissertation is not with estimating methods for truncated or censored data. However, we need to introduce them to

prevent us from possible confusion with the same terminology but different meaning. At this point we need to clarify the meaning of truncated or censored sample in the usual sense because Tiku (1967) defined it in a somewhat different manner.

Let n be the total number of sample specimens and r the number whose measurements are known. A sample is called as a singly truncated sample for which a terminus x_0 is specified and observation is possible only if $x \geq x_0$ (or $x \leq x_0$). Hence $n = r$ in this case. In a Type I singly censored sample, as in the singly truncated sample, a terminus x_0 is specified, but in this case sample specimens whose measurements fall in the restricted interval ($x \geq x_0$ or $x \leq x_0$) of the random variable may be identified and thus counted, though not otherwise measured. For example, in a life test experiment n items may be placed on test, but a decision made to terminate the test after a time x_0 has elapsed. Samples of this type thus consist of n observations of which r are fully observed and $n - r$ are censored with n being fixed and r a random variable. A Type II censored sample is one for which only r smallest (or largest) observations of n specimens are observed. Of the remaining $n - r$ censored observations, it is known only that $x < x_0$ (or $x > x_0$), where x_0 is the smallest (or largest) fully measured observation. In samples of this type both n and r are fixed, but x_0 is random. For example, in a life test, a total of n items is placed on test, but instead of continuing until all n items have failed, the test is terminated at the time of the rth item failure. Samples are said to be doubly truncated or doubly censored when truncation or censoring happens on the both sides. For more details of censoring or truncation, refer to Cohen(1957,1961) or Lawless (1982).

2.7 Modified Maximum Likelihood Estimation

Tiku (1967) also derived an *MLE* from what he called a censored sample and named it the modified maximum likelihood estimator (*MMLE*). Most of this section

discusses the work of Tiku (1967) and Tiku, Tan and Balakrishnan (1986).

Let $X_1, X_2, \ldots, X_n$ be a random sample of size n from a distribution. Arrange the n observations in ascending order of magnitude and censor (trim) the r_1 smallest and r_2 largest observations. The remaining observations

$$X_{(r_1+1)}, X_{(r_1+2)}, \ldots, X_{(n-r_2)}$$

constitute a Type II censored sample of size $n - r_1 - r_2$. The meaning of Type II censoring defined by Tiku (1967) is somewhat different from that described in Section 2.6. The censoring in this case is intentionally made after all the observations are collected, according to the order of magnitude, and not according to the order of observation in the sampling procedure. This procedure is actually a trimming of the sample, and not censoring as defined in Section 2.6. However, we can understand this procedure as follows. The sample size n and the amount of trimming r is fixed and the termini are random. Thus, the conditions to constitue a Type II doubly censored sample are satisfied as described in Section 2.6.

Suppose that the population distribution function is of the type $(1/\sigma)f((x-\mu)/\sigma)$, μ and σ being the location and scale parameters, respectively. Let $F(z) = \int_{-\infty}^{z} f(t)dt$, where $z = (x - \mu)/\sigma$. Then the likelihood function L based on a Type II censored sample $X_{(i)}, i = r_1 + 1, r_1 + 2, \ldots, n - r_2$ is given by

$$L = \frac{n!}{r_1! r_2!} \sigma^{-(n-r_1-r_2)} \left[\prod_{i=r_1+1}^{n-r_2} f(z_{(i)}) \right] \left[F(z_{(r_1+1)}) \right]^{r_1} \left[1 - F(z_{(n-r_2)}) \right]^{r_2}$$

where $z_{(i)} = (x_{(i)} - \mu)/\sigma$. The *MLE*s are solutions of the equations

$$\frac{\partial \log L}{\partial \mu} = 0 \quad \text{and} \quad \frac{\partial \log L}{\partial \sigma} = 0. \tag{2.22}$$

In many situations, however, the equation (2.22) is rather complex nonlinear and has to be solved by iterative methods (see Cohen, 1957, 1961; Harter and Moore, 1966). The solutions of the estimating equations for truncated and censored samples were proposed by Cohen (1950, 1957, 1961), using two-way interpolation. Due to the implicit nature of these iterations, however, it is difficult to make any analytical study of the resulting *MLE*, especially for small samples. Tiku (1967) introduced modified maximum likelihood estimators (*MMLE*s) as explicit functions of the sample observations.

Let $f(x;\theta)$ be the probability function of x, θ being an unknown parameter. To obtain the *MMLE*, Tiku sought a function L^* such that $d\log L^*/d\theta = 0$ has an explicit solution and is such that for all θ and $\mathbf{x}' = (x_1, x_2, \ldots, x_n)$,

$$\lim_{n\to\infty} \frac{d\log L^*(\mathbf{x})}{d\theta} = \frac{d\log L(\mathbf{x})}{d\theta}.$$

Then the *MMLE* is asymptotically identical with the *MLE*.

2.7.1 *MMLE*s for the Normal

For a sample of size n from the distribution $f(x) = \{\sqrt{2\pi}\sigma\}^{-1}\exp\{-(x-\mu)^2/(2\sigma^2)\}$, suppose that the smallest $r_1 = nq_1$ and the largest $r_2 = nq_2$ observations are censored, where q_1 and q_2 are fixed. It can be verified empirically that the points $g(z) = f(z)/[1 - F(z)]$ over an interval $a \leq z \leq b$ of finite length lie very close to the line $g(z) = \alpha + \beta z$ where

$$\beta = \frac{g(b) - g(a)}{b - a} \quad \text{and} \quad \alpha = g(z) - a\beta. \tag{2.23}$$

The maximum likelihood equations for estimating μ and σ are

$$\frac{\partial L}{\partial \mu} = \frac{n}{\sigma}\left\{\frac{1}{n}\sum_{i=r_1+1}^{n-r_2} z_{(i)} - q_1 g_1(z_{(r_1+1)}) + q_2 g_2(z_{(n-r_2)})\right\} = 0 \tag{2.24}$$

and

$$\frac{\partial L}{\partial \sigma} = \frac{n}{\sigma}\left\{-(1-q_1-q_2) + \frac{1}{n}\sum_{i=r_1+1}^{n-r_2} z_{(i)}^2 - q_1 z_{(r_1+1)} g_1(z_{(r_1+1)})\right.$$

$$\left. + q_2 z_{(n-r_2)} g_2(z_{(n-r_2)})\right\} = 0 \tag{2.25}$$

where $q_1 = r_1/n_1$, $q_2 = r_2/n_2$, $g_1(z) = f(z)/F(z)$ and $g_2(z) = f(z)/[1-F(z)]$.

To modify (2.24) and (2.25) so that they have explicit solutions, Tiku considered the linear approximations

$$g_1(z_{(r_1+1)}) \simeq \alpha_1 - \beta_1 z_{(r_1+1)} \quad \text{and} \quad g_2(z_{(n-r_2)}) \simeq \alpha_2 + \beta_2 z_{(n-r_2)}. \tag{2.26}$$

Note $g_1(z) = g_2(-z)$. In these equations,

$$\beta_1 = -\frac{g_1(k_1) - g_1(h_1)}{k_1 - h_1} \quad \text{and} \quad \alpha_1 = g_1(h_1) + h_1\beta_1 \tag{2.27}$$

$$\beta_2 = -\frac{g_2(k_2) - g_2(h_2)}{k_2 - h_2} \quad \text{and} \quad \alpha_2 = g_2(h_2) - h_2\beta_2 \tag{2.28}$$

where the intervals (h_1, k_1) and (h_2, k_2) are wide enough to cover both t_1 and $z_{(r_1+1)}$ and t_2 and $z_{(n-r_2)}$, respectively, and where t_1 and t_2 are determined by the equations

$$F(t_1) = q_1 \quad \text{and} \quad 1 - F(t_2) = q_2 \qquad \left(q_1 = \frac{r_1}{n},\ q_2 = \frac{r_2}{n}\right). \tag{2.29}$$

It is hard to know the exact locations of $z_{(r_1+1)}$ and $z_{(n-r_2)}$, but $\mathrm{E}(Z_{(r_1+1)}) = t_1$ and $\mathrm{E}(Z_{(n-r_2)}) = t_2$ for large n according to David (1981, p. 80). Hence, for reasonably large n, the interval (h_2, k_2) is likely to cover both t_2 and $z_{(n-r_2)}$ if h_2 and k_2 are determined by the equations

$$C_1 = 1 - F(h_2) = q_2 + \left[\frac{q_2(1-q_2)}{n}\right]^{\frac{1}{2}}, \quad \text{and} \quad C_2 = 1 - F(k_2) = q_2 + \left[\frac{q_2(1-q_2)}{n}\right]^{\frac{1}{2}},$$

and similarly for(h_1, k_1). To simplify the computations, Tiku (1967) suggested that one could take C_1 equal to the tabulated value of Pearson and Hartley (1972) just greater than $q_2 + \left[\frac{q_2(1-q_2)}{n}\right]^{\frac{1}{2}}$ and C_2 equal to the tabulated value just smaller than $q_2 + \left[\frac{q_2(1-q_2)}{n}\right]^{\frac{1}{2}}$. A comprehensive table for the values of α and β is given in Table B.1 (Tiku, Tan and Balakrishnan, 1986, p.75) when the sample size is less than or equal to 30.

As n tends to infinity, h_1 and k_1 both tend to t_1, and h_2 and k_2 both tend to t_2, in which case

$$\beta_1 = -\left[\frac{d}{dt}g_1(t)\right]_{t=t_1} = \frac{f(t_1)}{q_1}\left[t_1 + \frac{f(t_1)}{q_1}\right] \quad \text{and} \quad \alpha_1 = \frac{f(t_1)}{q_1} + \beta_1 t_1 \tag{2.30}$$

and

$$\beta_2 = -\left[\frac{d}{dt}g_2(t)\right]_{t=t_2} = \frac{f(t_2)}{q_2}\left[t_2 + \frac{f(t_2)}{q_2}\right] \quad \text{and} \quad \alpha_2 = \frac{f(t_2)}{q_2} - \beta_2 t_2. \tag{2.31}$$

The values of α and β from the equations (2.30) and (2.31) are also given in Table 2.7.1 under $n = \infty$ (Tiku, Tan and Balakrishnan, 1986, p.36; see also Tiku, 1967, p.157). Tiku suggested that the values of (α, β) may be obtained by using the asymptotic equations (2.30) and (2.31) for all $n \geq 10$ because the tabled values for $n \geq 10$ are not much different from each other.

Using the linear approximations (2.26), Tiku obtained the following *MML* equations from the ML equations (2.24) and (2.25):

$$\frac{\partial \log L}{\partial \mu} \simeq \frac{\partial \log L^*}{\partial \mu} = \frac{1}{\sigma}\left[\sum_{i=r_1+1}^{n-r_2} z_{(i)} - r_1(\alpha_1 - \beta_1 z_{(r_1+1)}) + r_2(\alpha_2 + \beta_2 z_{(n-r_2)})\right]$$
$$= \frac{m}{\sigma^2}(K + D\sigma - \mu) = 0 \tag{2.32}$$

and

$$\frac{\partial \log L}{\partial \sigma} \simeq \frac{\partial \log L^*}{\partial \sigma} = \frac{1}{\sigma}\left[-(n - r_1 - r_2) + \sum_{i=r_1+1}^{n-r_2} z_{(i)}^2 - r_1 z_{(r_1+1)}(\alpha_1 - \beta_1 z_{(r_1+1)})\right.$$
$$\left. + r_2 z_{(n-r_2)}(\alpha_2 + \beta_2 z_{(n-r_2)})\right]$$
$$= -\frac{A}{\sigma^3}\left(\sigma - \frac{B(\mu) + \left\{[B(\mu)]^2 + 4AC(\mu)\right\}^{\frac{1}{2}}}{2A}\right)$$
$$\times \left(\sigma - \frac{B(\mu) - \left\{[B(\mu)]^2 + 4AC(\mu)\right\}^{\frac{1}{2}}}{2A}\right)$$
$$= -\frac{1}{\sigma^3}\left[(A\sigma^3 - B\sigma - C) - m(K - \mu)(K + D\sigma - \mu)\right]$$
$$= 0 \tag{2.33}$$

where

$$A = n - r_1 - r_2, \qquad m = n - r_1 - r_2 + r_1\beta_1 + r_2\beta_2, \qquad D = \frac{r_2\alpha_2 - r_1\alpha_1}{m},$$

$$K = \frac{\sum_{i=r_1+1}^{n-r_2} x_{(i)} + r_1\beta_1 x_{(r_1+1)} + r_2\beta_2 x_{(n-r_2)}}{m},$$

$$B = r_2\alpha_2 \left(x_{(n-r_2)} - K\right) - r_1\alpha \left(x_{(r_1+1)} - K\right),$$

$$C = \sum_{i=r_1+1}^{n-r_2} \left(x_{(i)} - K\right)^2 + r_1\beta_1 \left(x_{(r_1+1)} - K\right)^2 + r_2\beta_2 \left(x_{(n-r_2)} - K\right)^2$$

$$= \sum_{i=r_1+1}^{n-r_2} x_{(i)}^2 + r_1\beta_1 x_{(r_1+1)}^2 + r_2\beta_2 x_{(n-r_2)}^2 - mK^2,$$

and where

$$B(\mu) = r_2\alpha_2 \left(x_{(n-r_2)} - \mu\right) - r_1\alpha \left(x_{(r_1+1)} - \mu\right),$$

$$C(\mu) = \sum_{i=r_1+1}^{n-r_2} \left(x_{(i)} - \mu\right)^2 + r_1\beta_1 \left(x_{(r_1+1)} - \mu\right)^2 + r_2\beta_2 \left(x_{(n-r_2)} - \mu\right)^2$$

$$= \sum_{i=r_1+1}^{n-r_2} x_{(i)}^2 + r_1\beta_1 x_{(r_1+1)}^2 + r_2\beta_2 x_{(n-r_2)}^2 - m\mu^2.$$

Note that both (2.32) and (2.33) satisfy the condition (2.29) since asymptotically $z_{(r_1+1)} = t_1$ and $z_{(n-r_2)} = t_2$, in which case

$$g_1\left(z_{(r_1+1)}\right) = \alpha_1 - \beta_1 z_{(r_1+1)} \qquad \text{and} \qquad g_2\left(z_{(n-r_2)}\right) = \alpha_2 + \beta_2 z_{(n-r_2)}. \tag{2.34}$$

The solutions to equations (2.32) and (2.33), with correction for bias in the estimators

of σ, yield the *MML* estimators $\hat{\mu}_c$ and $\hat{\sigma}_c$ as

$$\hat{\mu}_c = K + D\hat{\sigma}_c \tag{2.35}$$

and

$$\hat{\sigma}_c = \frac{B + (B^2 + 4AC)^{\frac{1}{2}}}{2\left[A\left(A-1\right)\right]^{\frac{1}{2}}}. \tag{2.36}$$

For $r_1 = r_2 = 0$, i.e., in the case of no censoring on either side of the sample, $\hat{\mu}_c$ and $\hat{\sigma}_c^2$ reduce to the sample mean and sample variance, respectively. Now, the equation

$$\frac{\partial \log L^*}{\partial \mu} = \frac{m}{\sigma^2}\left(K + D\sigma - \mu\right)$$

can be easily be solved to give

$$L^* = \exp\left[-\frac{m}{2\sigma^2}\left(K + D\sigma - \mu\right)^2 + g(\mathbf{x}, \sigma)\right], \tag{2.37}$$

where $g(\mathbf{x}, \sigma)$ is a function of $\mathbf{x}' = (x_{(r_1+1)}, x_{(r_1+2)}, \ldots, x_{(n-r_2)})$ and σ but is free of μ. From (2.33) and (2.37), we obtain

$$\left(\frac{\partial \log L^*}{\partial \sigma}\right)_{\mu=K+D\sigma} = \frac{\partial}{\partial \sigma} g(\mathbf{x}, \sigma) = \frac{1}{\sigma^3}\left(C + B\sigma - A\sigma^2\right)$$

where $g(\mathbf{x}, \sigma)$ is a function of $\mathbf{x}' = (x_{(r_1+1)}, x_{(r_1+2)}, \ldots, x_{(n-r_2)})$ and σ but is free of μ. From (2.33) and (2.37), we obtain

$$\left(\frac{\partial \log L^*}{\partial \sigma}\right)_{\mu=K+D\sigma} = \frac{\partial}{\partial \sigma} g(\mathbf{x}, \sigma) = \frac{1}{\sigma^3}\left(C + B\sigma - A\sigma^2\right)$$

which gives $g(\mathbf{x},\sigma) = (-B/\sigma - C/2\sigma^2 - A\log\sigma)\,H(\mathbf{x})$, where $H(\mathbf{x})$ is a function of $\mathbf{x}$ and is free of μ and σ. It follows that

$$L^* = \sigma^{-A}\exp\left[-\frac{B}{\sigma} - \frac{C}{2\sigma^2} - \frac{m}{2\sigma^2}\left(K + D\sigma - \mu\right)^2\right] H(\mathbf{x}). \tag{2.38}$$

If $r_1 = r_2 = r$, i.e., for symmetric censoring, then $\alpha_1 = \alpha_2 = \alpha$, $\beta_1 = \beta_2 = \beta$ and $D = 0$, in which case m, K, A, B, and C reduce to

$$m = n - 2r + 2r\beta, \tag{2.39}$$

$$\hat{\mu} = K = \frac{\sum_{i=r+1}^{n-r} x_{(i)} + r\beta\left(x_{(r+1)} + x_{(n-r)}\right)}{m}, \tag{2.40}$$

$$A = n - 2r, \tag{2.41}$$

$$B = r\alpha\left(x_{(n-r)} - x_{(r+1)}\right) \tag{2.42}$$

and

$$C = \sum_{i=r+1}^{n-r}\left(x_{(i)} - \hat{\mu}\right)^2 + r\beta\left[\left(x_{(n-r)} - \hat{\mu}\right)^2 + \left(x_{(r+1)} - \hat{\mu}\right)^2\right]$$

$$= \sum_{i=r+1}^{n-r} x_{(i)}^2 + r\beta\left(x_{(r+1)}^2 + x_{(n-r)}^2\right) - m\hat{\mu}^2. \tag{2.43}$$

The *MMLE*s, with correction for bias, for μ and σ are given by $\hat{\mu}$ and

$$\hat{\sigma} = \frac{B + (B^2 + 4AC)^{\frac{1}{2}}}{2\left[A\left(A-1\right)\right]^{\frac{1}{2}}}. \tag{2.44}$$

Tiku noted that for small samples, the estimators $\hat{\mu}_c$ and $\hat{\sigma}_c$ based on asymmetric censored samples (i.e., $r_1 \neq r_2$) are biased. To study this bias, Tiku(1970) conducted a Monte Carlo investigation and concluded that unless r_1 and r_2 are very different from each other, the bias is negligible.

Many statistical tests have been proposed by Tiku and others, which use the *MMLE*. Some of these test statistics will be discussed in Chapter 5 and Chapter 7.

2.8 Minimum Hellinger Distance Estimation

The Hellinger distance (HD^*) between two probability densities f and g is defined as

$$HD^*(f,g) = \left\{ \int (f^{1/2} - g^{1/2})^2 \right\}^{1/2} = ||f^{1/2} - g^{1/2}|| \tag{2.45}$$

where $||\cdot||$ represents L_2-norm. One can easily show that $HD^{*2}(f,g) = 2 - 2\int (fg)^{1/2}$. Moreover, $0 \leq \int (fg)^{1/2} \leq 1$ with equality on the left if and only if f and g are mutually singular and equality from right if and only if $f(y) = g(y)$ for all y.

Bhattacharyya (1943) used a function of the Hellinger distance as a measure of divergence between two populations. He showed that the Hellinger distance between empirical and theoretical multinomial distributions is asymptotically proportional to the usual chi-squared statistic. Matusita (1955) considered some applications of the Hellinger distance with multinomial distributions and obtained some limited results of the minimum Hellinger distance estimation and goodness-of-fit testing. Rao (1963)

also considered minimum Hellinger distance estimation for the multinomial distribution in comparison with several other methods of estimation. Papaioannou and Kempthorne (1971) proposed using a function related to the Hellinger distance as an alternative to the Kullback-Leibler discriminator information measure. Beran (1977a, 1977b) appears to be the first author to use the Hellinger distance in the context of robust estimation. In the first paper, he defines robustness of a location estimator as Hellinger differentiability of the corresponding functional. He then considers certain optimality properties for such an estimator. In his second paper, Beran (1977b) considers minimum Hellinger distance estimation for general parametric families. Pitman (1979) used the Hellinger distance in a rather elegant treatment of several fundamental results in statistical inference such as the Cramér-Rao inequality and the asymptotic behavior of maximum likelihood estimators. Stather (1981) extended Beran's results to include both discrete distributions and continuous distributions with infinite support. Stather also considered a modified minimum Hellinger distance approach to the problem of regression. Simpson (1987) considered minimum Hellinger distance estimation in the case of count data. Simpson confirmed Beran's results of robustness and efficiency of the minimum Hellinger distance estimator (*MHDE*). We will discuss minimum Hellinger distance estimation based mostly on Beran's (1977b) work.

2.8.1 Properties of the *MHDE*

Let $X_1, X_2, \ldots, X_n$ be iid random variables with a density belonging to a specified parametric family $\mathcal{G} = \{g_\theta : \theta \in \Theta\}$, where $\Theta \subset \mathbb{R}^p$. Let g_θ be the model density and let F_n and f_n be the empirical cdf and the empirical pdf respectively. The minimum Hellinger distance estimator of θ is that value (or values) θ^* in the parameter space Θ which minimizes the HD^* between f^* and g_θ, where f^* is a nonparametric density estimator of the true density estimated from the data. Beran noted that the *MHDE*

is related heuristically to the *MLE* of θ under the model, but not otherwise.

Let $\mathcal{F}$ denote the set of all densities with respect to Lebesgue measure. The *minimum Hellinger distance functional* $T(f)$ defined on $\mathcal{F}$ is

$$||f^{1/2} - g_{T(f)}^{1/2}|| = \min_{\theta \in \Theta} ||f^{1/2} - g_{\theta}^{1/2}||$$

for every $f \in \mathcal{F}$. If $T(f)$ is not uniquely defined, choose any of the possible values arbitrarily. For a given density estimator f^*, $T(f^*)$ is the *MHDE*. Beran (1977b) gave some results for the functional $T(f^*)$ which imply the existence, consistency, and asymptotic uniqueness under certain conditions. Stather (1981) has obtained similar results. Beran (1977b) and Stather (1981) also showed the asymptotic nomality of $T(f^*)$. In particular, if $f = g_\theta$, the *MHDE*, θ^*, and the *MLE*, $\hat{\theta}$, are asymptotically equivalent, i.e., $\theta^* = \hat{\theta} + o_p(n^{-1/4})$

2.8.2 Robustness of the *MHDE*

Beran (1977b) studied the robustness of the estimator $T(f^*)$ by considering what happens to the distribution of $T(f^*)$ as the distribution of data is varied. To do this Beran studied the Hellinger continuity of the approximated distribution for $T(f^*)$ as f varies within a small Hellinger neighborhood of g_θ. As a result, he proved that the normal approximation to the distribution of $T(f^*)$ is itself Hellinger continuous at g_θ and hence encourages the belief that the *MHDE* is a robust estimator under data contamination which corresponds to a small Hellinger metric perturbation of g_θ. By the nature of the continuity of $T(\cdot)$, small Hellinger metric changes in f^* will typically induce a correspondingly small change in the value of $T(f^*)$.

Beran also showed that the minimum Hellinger distance functional T is insensitive to perturbations of its argument in an infinitesimal neighborhood of g_θ. To make

this precise, Beran considered the set of all functions U defined on $\mathcal{F}$ that have the following properties: for every $\theta \in \Theta^\circ$, where Θ° is the interior of Θ,

$$U(g_\theta) = \theta$$

$$U(f) - U(g_\theta) = \int \varrho(x)\left[f^{1/2}(x) - g_\theta^{1/2}(x)\right]dx - o(\|f^{1/2} - g_\theta^{1/2}\|), \tag{2.46}$$

where ϱ is a p–dimensional vector whose component belongs to L_2. Without loss of generality, orthogonality of $g^{1/2}$ and every component of ϱ can be assumed. Beran examined the behavior of $c^T[U(f) - \theta]$ for every constant vector $c \in \mathbb{R}^p$ to find the functional U which is least affected by infinitesimal perturbation of g_θ, assuming that f is near g_θ in the Hellinger metric. Under some technical assumptions he showed that the functional T is (typically) locally minimax robust at g_θ, and $T(f^*)$ is asymptotically efficient under g_θ.

Furthermore, Beran investigated the behavior of T under a mixture model for gross errors; his results confirmed that T is robust and revealed the limitations of Hampel's (1974) influence function in assessing robustness. Let δ_x denote the uniform density on the interval $(z-\eta, z-\eta)$, where $\eta > 0$ is very small, and let $g_{\theta,\eta,z} = (1-\alpha)g_\theta + \alpha\delta_x$ for $\theta \in \Theta$, $\alpha \in [0, 1)$, and real z. The density models an experiment where independent observation distributed according to g_θ are mixed with approximately $100\alpha\%$ gross errors located near z. Beran proved that for every $\alpha \in (0, 1)$, the difference quotient (or α–influence curve) $\alpha^{-1}[T(g_{\theta,\eta,z}) - \theta]$ is a bounded continuous function of z such that $\lim_{z\to\infty} \alpha^{-1}[T(g_{\theta,\eta,z}) - \theta] = 0$. Hence the functional T is robust at g_θ against $100\alpha\%$ contamination by gross error at arbitrary real z; whether or not the influence curve of T is bounded is irrelevant.

In other words, the α–influence curves of T do not need to converge to the in-

fluence curve uniformly in z, so that the influence curve can differ dramatically in shape from each of the α–influence curves. This means, as Beran concluded, first, it is necessary to examine the α–influence curves rather than the influence curve to assess the robustness of a functional with respect to the gross error model, unless the convergence is uniform in z. Second, since a functional with well-behaved α–influence curves can have an unbounded influence curve, an estimator can be both robust and asymptotically efficient at the same time.

2.9 Hellinger Deviance Test and the Notion of Breakdown

The *likelihood ratio test* (LRT) is one of the most popular tests used in parametric inference. Generally, however, the LRT is not robust to outliers in the sample data. Simpson (1989) proposed a test based on the minimum Hellinger distance and called it the *Hellinger deviance test.* He showed that the Hellinger deviance test is asymptotically equivalent to the likelihood ratio test for local parametric alternatives.

Suppose $X_1, X_2, \ldots, X_n$ are independent observations in $\mathbb{R}^d$ with a density f_θ in $\mathcal{F} = \{f_\theta; \theta \in \Theta\}$, where $\Theta \in \mathbb{R}^p$. The Hellinger distance (HD^*) between two densities f and g is given as in (2.45).

Suppose we want to test the hypothesis

$$\mathrm{H}_0 : \theta \in \Theta_0 \qquad \text{against} \qquad \mathrm{H}_1 : \theta \in \Theta - \Theta_0,$$

where $\Theta_0 \subset \Theta$. The log-likelihood ratio test statistic has the form

$$\lambda_n = 2[\log\{L_n(\tilde{\theta})\} - \log\{L_n(\tilde{\theta}_0\}]$$

where $L_n(\theta) = \prod f_\theta(X_i)$ and $\tilde{\theta}$ and $\tilde{\theta}_0$ are suprema of L_n over Θ and Θ_0, respectively.

Simpson (1989) proposed the Hellinger deviance test of the form

$$d_n = 4n\{HD^{*2}(f_{\hat{\theta}_0}, \hat{f}) - HD^{*2}(f_{\hat{\theta}}, \hat{f})\} \tag{2.47}$$

where $\hat{\theta}$ and $\hat{\theta}_0$ correspond to the infima of $HD^{*2}(f_\theta, \hat{f})$ over Θ and Θ_0, respectively, and where $\hat{f}$ is the kernel density estimator of f_θ with a suitable kernel function. Similar statistics have been disscussed by Matusida (1955), Beran (1977b), Stather (1981), Cressie and Read (1984), and Sarkar and Basu (1995).

Simpson (1989, theorem 1) showed that under suitable regularity conditions

$$d_n - \lambda_n \xrightarrow{p} 0 \qquad \text{as } n \to \infty$$

which implies that the Hellinger deviance test has the asymptotic power for local parametric alternatives as the likelihood ratio test with the same critical values. Moreover, if the dimension of Θ is p and that of Θ_0 is $q(q < p)$, then d_n is approximately χ^2 distributed with $p-q$ degrees of freedom. Simpson also illustrated that the Hellinger distance tests have generally high breakdown point whereas, in many instances, the likelihood ratio tests have breakdown point of 0.

Breakdown point, roughly speaking, is the smallest fraction of data contamination needed to cause an arbitrarily large change in the estimate. An appealing feature of the breakdown point is its ease of interpretation as compared to more subtle indicators of stability like asymptotic minimax results. On the other hand, the breakdown point is concerned with extreme breakdown and extreme contamination that might be deemed unrealistic. Donoho and Huber (1983) strongly advocated the breakdown points as a measure of the insensitivity of an estimator to multiple outliers in the data. They noted the breakdown point is most useful in a small sample situation. He, Simpson and Portnoy (1990) introduced power and level breakdown functions

of a test statistic as a unified analysis of local stability and global stability. The power breakdown function gives the amount of contamination of each alternative distribution that can carry the test to a null value. That is, the power breakdown function of a test is roughly the least amount of contamination such that the test fails to be consistent. A minimum requirement for a testing procedure to be robust is that for any situation where the null hypothesis is false, the power breakdown function should be grater than zero. The level breakdown function gives the amount of contamination of a null distribution that can carry the test statistic to each value in the alternative space. For example, the classical t test has the power breakdown point identical to 0 and the level breakdown point $(\mu/\sigma)^2\{1 + (\mu/\sigma)^2\}^{-1}$. Hence, the t test is not robust and also lacks power. He *et al.* found that, in terms of the breakdown function, the sign test performs better than the Wilcoxon test or the t test, but Brown (1982) found the sign test to be sensitive to unequal variances.

CHAPTER III

MINIMUM DISPARITY ESTIMATION

In this chapter we will briefly discuss the use of the minimum distance methods which generate first order efficient robust estimators in parametric models. Lindsay (1994) has investigated and identified the particular geometric characteristics of the disparities between the data and the model which make the methods robust. Lindsay has defined a function called the residual adjustment function (RAF) which is instrumental in explaining the theoretical properties of minimum disparity estimators (MDEs). Lindsay's research was limited to the discrete case. Basu and Lindsay (1994) extended Lindsay's work to the continuous case. The discussion below will follow Lindsay (1994), and Basu and Lindsay (1994).

Note that many results of Basu (1991) will be also used in this chapter. There will be some confusion regarding the chronological order. The original unpublished manuscript of Lindsay (1994) was prepared in 1990, and Basu (1991) established his work based on Lindsay's 1990 manuscript.

3.1 Discrete Case

Lindsay (1994) investigated a large class of minimum distance type methods, including both maximum likelihood and minimum Hellinger distance, in the context of the multinomial model, because the multinomial model offers a simple setting with minimal mathematical nuisance.

Suppose that the sample space $\mathfrak{X}$ is a countable set, without loss of generality $\mathfrak{X} = \{0, 1, \ldots, K\}$, with K possibly infinite, and that $m_\theta(x)$ is a family of probability densities on $\mathfrak{X}$, indexed by $\theta \in \Theta$. Further, suppose that n independently and iden-

tically distributed observations $X_1, X_2, \ldots, X_n$ are made from $m_\theta(x)$. Let $d_n(x)$ be the proportion of the n observations which have value x. The function $d_n(x)$ will be called the empirical density function.

We will consider estimators that are constructed by minimizing the distance between $d_n(x)$ and $m_\theta(x)$ over $\theta \in \Theta$. In particular, Lindsay looked at a special subclass of distances generating first order efficient estimators, which he called *disparities*. Here, we will discuss the most important findings of Lindsay's work in developing the idea of the minimum disparity estimator.

3.1.1 Disparity Measures

Let G be a real valued, thrice differentiable function on $[-1, \infty)$ with $G(0) = 0$. Define the *Pearson residual function* $\delta_n(x)$ to be

$$\delta_n(x) = \frac{d_n(x) - m_\theta(x)}{m_\theta(x)}.$$

Then for any pair of densities $m_\theta(x)$ and $d_n(x)$ the *disparity measure* ρ_G, determined by G, is defined by

$$\rho_G(d_n, m_\theta) = \sum m_\theta(x) G(d_n(x)). \tag{3.1}$$

If G is a strictly convex function, then it follows from the Jensen's inequality that the above disparity is positive. The function ρ_G is equal to zero only when the functions d_n and m_θ are identical. Note too that the Pearson residuals have ranges $[-1, \infty)$.

An important class of such measure is the Cressie-Read (Cressie and Read, 1984;

Read and Cressie, 1988) family of *power divergences measures,* defined by

$$PWD_\lambda(d_n, m_\theta) = \sum d_n(x)\frac{[d_n(x)/m_\theta(x)]^\lambda - 1}{\lambda(\lambda+1)} \tag{3.2}$$

$$= \sum m_\theta(x)(1+\delta_n(x))\frac{(1+\delta_n(x))^\lambda - 1}{\lambda(\lambda+1)}. \tag{3.3}$$

For λ = -2, -1, -$\frac{1}{2}$, 0 and 1, the following well-known disparity measures are obtained: Neyman's chi-squared (NCS) divided by 2,

$$NCS(d_n, m_\theta) = \sum \frac{[d_n(x) - m_\theta(x)]^2}{2d_n(x)}; \tag{3.4}$$

Kullback-Leibler (KL) divergence,

$$KL(d_n, m_\theta) = \sum m_\theta(x)[\log m_\theta(x) - \log d_n(x)]; \tag{3.5}$$

Pearson's chi-squared (PCS) divided by 2,

$$PCS(d_n, m_\theta) = \sum \frac{[d_n(x) - m_\theta(x)]^2}{2m_\theta(x)}; \tag{3.6}$$

likelihood disparity (LD),

$$LD(d_n, m_\theta) = \sum d_n(x)[\log d_n(x) - \log m_\theta(x)]; \tag{3.7}$$

twice-squared Hellinger distance (HD)

$$HD(d_n, m_\theta) = 2\sum \left[d_n(x)^{1/2} - {m_\theta}^{1/2}\right]^2. \tag{3.8}$$

Note that the (twice-squared) Hellinger distance (HD) we are using in this dissertation differs from the Hellinger distance (HD^*) given in (2.45), i.e., $HD = 2HD^{*2}$. Using the

squared quantity of the HD^* in (3.8) helps to simplify the estimation and the factor of 2 has a standardizing purpose. This does not change the estimation properties of the *MHDE*. Note that the *PWD* can be made continuous at $\lambda = -1$ and $\lambda = 0$ by letting $PWD_{\lambda=-1} = \lim_{\lambda\to-1} PWD_\lambda$ and $PWD_{\lambda=0} = \lim_{\lambda\to 0} PWD_\lambda$.

3.1.2 Residual Adjustment Functions

Let ∇ denote differentiation with respect to θ. Under differentiability of the model with respect to θ, minimization of the disparity measure $\rho(d_n, m_\theta)$ over θ corresponds to solving an estimating equation of the form

$$-\nabla\rho_G(d_n, m_\theta) = \sum \left[G'(\delta_n(x))\frac{d_n(x)}{m_\theta(x)} - G(d_n(x))\right]\nabla m_\theta(x) = 0. \tag{3.9}$$

Here we have the negative derivative of the disparity so that in the case of the likelihood disparity (3.7) the equation agrees with the usual likelihood equations. The focal point of Lindsay's paper is, in general, that the minimum disparity estimating equation corresponding to the disparity measure ρ is of the form

$$-\nabla\rho = \sum A_\rho(\delta_n(x))\nabla m_\theta(x)\,dx = 0 \tag{3.10}$$

where $A_\rho(\delta_n) = (1+\delta_n)G'(\delta_n) - G(\delta_n)$. From here on $A_\rho(\delta_n)$ will be denoted as $A(\delta_n)$ if there is no confusion. Since $A'(\delta_n) = (1+\delta_n)G''(\delta_n)$, the function $A(\delta_n)$ is strictly increasing on $[-1, \infty)$ if G is strictly convex. Also, since $\sum \nabla m_\theta(x) = 0$ we can properly center the function A so that $A(0) = 0$. Furthermore, as $A'(0) = G''(0) > 0$, we can rescale A to make $A'(0) = 1$. These standardizations necessitate dividing *NCS* in (3.4) and *PCS* in (3.6) by 2, and multiplying *HD* in (3.8) by 2.

This centered and scaled function A is called the *residual adjustment function* (*RAF*) corresponding to the disparity measure ρ determined by G. The role of *RAF*

is very important in deriving most of the theoretical results about the efficiency, distributions and robustness of the minimum disparity estimators (*MDE*s).

For the likelihood disparity we have $A(\delta_n) = \delta_n$, so that maximum likelihood estimator treats the Pearson residuals linearly. Other *MDE*s can then be compared with the *MLE* based on how their *RAF*s deviate from linearity. All of the *MDE*s are essentially the same in terms of the first order asymptotic theory. They have the same asymptotic distribution and the same first order bias approximation. Figure 3.1, which shows the *RAF*s for *HD*, *LD* and the negative exponential disparity (*NED*) which we will introduce in Section 3.1.4 and Section 3.2.4, shows how the residuals are treated differently by the different disparities. But the residuals go to zero with probability 1 as n goes to infinity and so up to the first order this difference in the treatment of the residuals dose not matter. Later we will see how the difference in the treatment of the residual affects the estimation at the higher order. Also, the second derivative of *RAF* at zero, $A_2 = A''(0)$, called the *estimation curvature* or *curvature parameter,* will be shown to carry important information about the robustness and efficiency of the minimum disparity estimator.

Lindsay (1994) also presented the robustness property and the asymptotic properties of *MDE*s. We will discuss these topics in Section 3.2 together with the continuous case.

Lindsay (1994) showed how various distance measures can be modified to create disparity measures with a wide range of estimation curvatures A_2 by introducing *blended families of measures.* Lindsay also introduced the *negative exponential disparity* (*NED*) measure which produces robust *MDE*s, not only against outliers but against inliers as well. We will discuss these disparity measures in the next two sections for the discrete case.

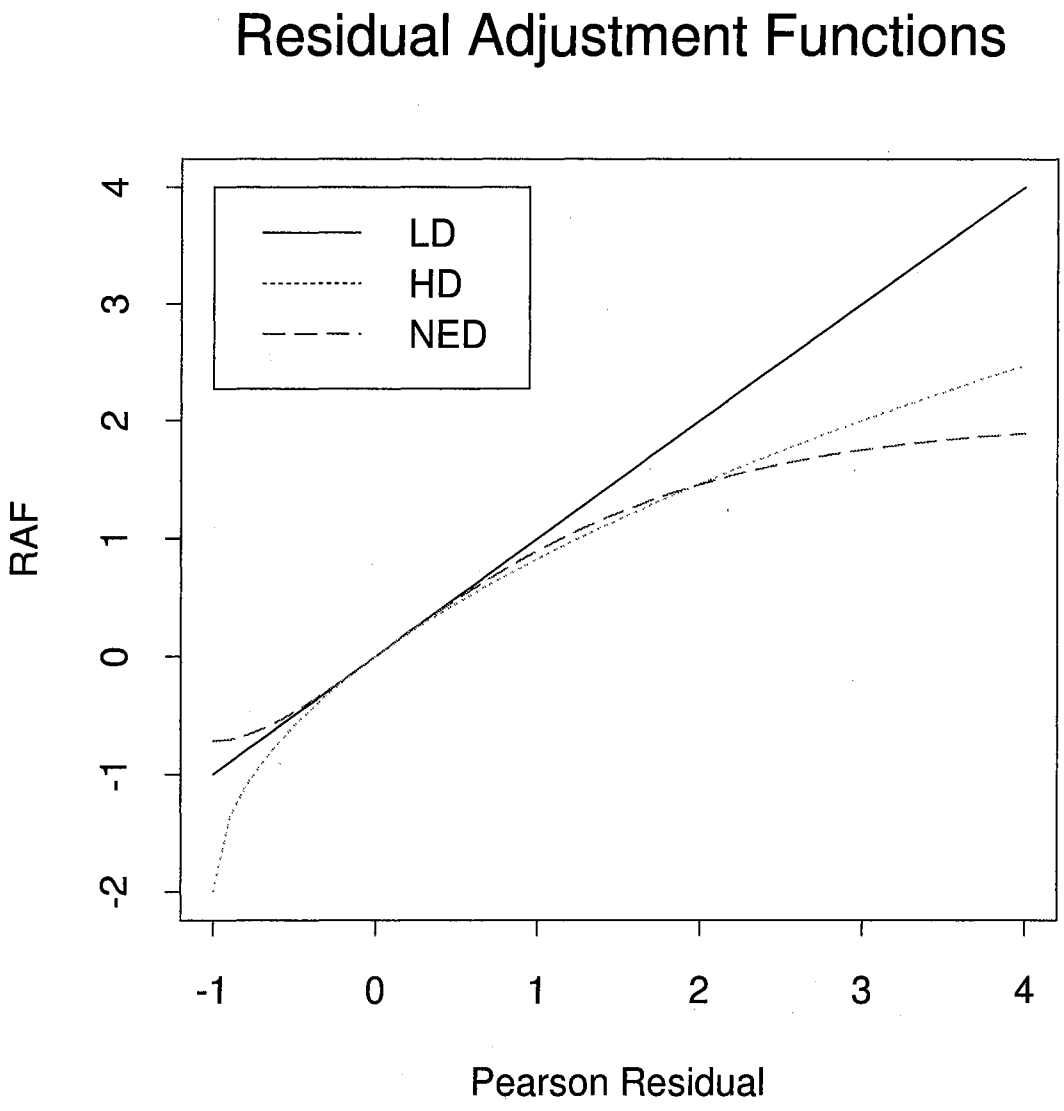

Figure 3.1: The residual adjustment functions of the likelihood disparity (LD), the Hellinger distance (HD) and the negative exponential disparity (NED).

3.1.3 Blended Families of Disparity Measures

Using a chi-squared distance, we can modify the "weights" given to the squared discrepancies. Let $\alpha \in [0,1]$ be a fixed number, and $\bar{\alpha} = 1 - \alpha$. Then define the *blended weight chi-squared* ($BWCS_\alpha$) disparity family to be

$$BWCS_\alpha = \sum \frac{\left[d_n(x) - m_\theta(x)\right]^2}{2\left[\alpha d_n(x) + \bar{\alpha} m_\theta(x)\right]}. \tag{3.11}$$

$BWCS_0$ corresponds to the Pearson's chi-squared and $BWCS_1$ corresponds to Neyman's chi-squared. The corresponding residual adjustment function is

$$A_\alpha(\delta_n) = \frac{\delta_n}{1 + \alpha\delta_n} + \frac{\bar{\alpha}}{2}\left[\frac{\delta_n}{1 + \bar{\alpha}\delta_n}\right]^2.$$

Another weighting scheme which generalizes Hellinger distance is *blended weight Hellinger distance* ($BWHD$). The blended weight Hellinger distance family can be defined as follows

$$BWHD_\alpha = \sum \frac{\left[d_n(x) - m_\theta(x)\right]^2}{2\left[\alpha\sqrt{d_n(x)} + \bar{\alpha}\sqrt{m_\theta(x)}\right]^2}. \tag{3.12}$$

This family of the Hellinger distance generates NCS when $\alpha = 1$ and PCS when $\alpha = 0$ and HD when $\alpha = 0.5$. The RAF of this family is

$$A_\alpha(\delta_n) = \delta_n\left[\alpha\sqrt{\delta_n + 1} + \bar{\alpha}\right]^{-2} + \frac{\alpha}{2}\delta_n^2\left[\alpha\sqrt{\delta_n + 1} + \bar{\alpha}\right]^{-3}.$$

Note that the estimation curvature of both disparities is $A_2 = 1 - 3\alpha$. So that the choice of $\alpha = \frac{1}{3}$, we will obtain $A_2 = 0$.

Basu and Sarkar (1994a) proposed goodness-of-fit tests based on $BWCS_\alpha$ and $BWHD_\alpha$ for multinomial models. They showed that the asymptotic null distribution

of the tests is chi-squared and recommended tests based on $BWCS_{1/3}$ and $BWHD_{1/9}$ for testing goodness-of-fit. Shin, Basu and Sarkar (1995, 1996) introduced a class of goodness-of-fit tests based on $BWHD_\alpha$ and $BWCS_\alpha$ which is a rich subfamily of the family of disparity tests introduced by Basu and Sarkar (1994a). They derived approximations of the exact null distributions of $BWHD_\alpha$ and $BWCS_\alpha$ tests that depend on the index parameter α. These approximation results make $BWHD_\alpha$ and $BWCS_\alpha$ tests obtain more accurate significance levels even in small samples.

3.1.4 Negative Exponential Disparity

Lindsay (1994) introduced a new disparity which can be effective for *inliers* as well as outliers. Inliers are values with δ close to -1 and represent cells having missing or nearly missing data.

Suppose one desires robustness against inliers and wants to seek RAFs that down-weight both positive and negative residuals relative to the maximum likelihood estimator, i.e., $|A(\delta)| \leq |\delta|$. Given that $A(0) = 0$ and $A'(0) = 1$ we want $A(\delta)$ to cross $A_{LD} = \delta$, at $\delta = 0$, and since $A'(0) = 1$ we must have $A''(0) = A_2 = 0$, otherwise $A(\delta)$ would stay on one side of $A_{LD}(\delta)$ in some neighborhood of 0; that is, $A(\delta)$ must be second-order efficient. The third derivative $A'''(0)$, if not itself zero, must be negative so that the crossing happens in the right direction. Lindsay starts with the convex function $G(\delta) = e^{-\delta} - 1$ to investigate the implication of above.

The negative exponential disparity (NED) measure corresponding to the function $G(\delta) = e^{-\delta} - 1$ is defined as

$$NED = \sum G(\delta) m_\theta(x) \tag{3.13}$$

and its RAF is given by

$$A_{NED}(\delta) = 2 - (2+\delta)e^{-\delta}.$$

Further, since

$$A'(\delta) = (1+\delta)e^{-\delta}, \qquad A''(\delta) = -\delta e^{-\delta}, \qquad \text{and} \qquad A'''(\delta) = (\delta - 1)e^{-\delta},$$

the minimum negative exponential disparity estimator ($MNEDE$) generates a second-order efficient estimator which shrinks both positive and negative residuals.

Basu and Sarkar (1994c), and Basu, Sarkar and Vidyashankar (1997) intensively studied the properties of minimum negative exponential disparity estimation. We will discuss this in detail later in Section 3.2.4.

3.1.5 Robustness of the *MDE*s

Let T be a statistical functional defined on the class of distributions $\mathcal{F}$. Let $x, y \in \mathcal{X}$, sample space, and $t \in \mathcal{F}$. Define the ϵ contaminated version of $t(x)$ as $t_\epsilon(x) = (1-\epsilon)t(x) + \epsilon\chi_y(x)$ where $\chi_y(x)$ is the indicator function for y. Then the influence function is defined to be

$$T'(y) = \left.\frac{\partial T\big((1-\epsilon)t + \epsilon I(x)\big)}{\epsilon}\right|_{\epsilon=0} = \frac{\partial T(t_\epsilon)}{\partial \epsilon}.$$

The following lemma of Lindsay is the key to establish that all the MDEs have the same influence function under the model $m_\theta(x)$. Let the score function $u(y,\theta)$ be $\nabla \log(m_\theta(y))$.

Proposition 3.1 (Lindsay, 1994). *For an estimating function of the form* $\sum A(\delta(x))$

$\times \nabla m_\theta(x)$, *the influence function of the estimators* $T(\cdot)$ *has the form*

$$T'(y) = \frac{A'\big(\delta(y)\big)u(y,\theta^*) - E\big[A'\big(\delta(X)\big)u(X,\theta^*)\big]}{E\big[u(X,\theta^*)u(X,\theta^*)^T A'\big(\delta(X)\big)\big] + \sum A\big(\delta(X)\big)\nabla^2 m_{\theta^*}(x)}$$

where $\theta^* = T(t)$, $\delta(x) = \big[t(x) - m_{\theta^*}(x)\big]/m_{\theta^*}(x)$. *If the density* $t(x)$ *is a model point* $m_\theta(x)$, *then* $\theta^* = \theta$, $\delta(x) = 0$ *for all* x, *and the estimators defined by* $\sum A(\delta(x))\nabla m_\theta(x) = 0$ *have influence function* $I(\theta)^{-1}u(y,\theta)$ *where* $I(\cdot)$ *is the Fisher information.*

Proof. See Lindsay (1994), Proposition 1. □

Hence, from Proposition 3.1, if the functional is the maximum likelihood estimator or the minimum Hellinger distance estimator, for $t(x) = m_\theta(x)$

$$T'(y) = \left.\frac{\partial T\big((1-\epsilon)m_\theta + \epsilon I(x)\big)}{\epsilon}\right|_{\epsilon=0} = \frac{u(y;\theta)}{I(\theta)}. \tag{3.14}$$

As a function of ϵ, $\Delta T(\epsilon) = T(t_\epsilon) - T(t)$ represents how the functional changes with contamination. Consider the first order Taylor expansion

$$\Delta T(\epsilon) = T(t_\epsilon) - T(t) \approx \epsilon T'(y). \tag{3.15}$$

This approximation is critical to the dual role of the influence function as a measure of efficiency and robustness. First, as shown in the previous chapter, the asymptotic variance of the estimator comes from the approximation $T(d) - T(t) = \sum T'd(x)$. In our case, if $t = m_\theta$, any estimator with the same influence function as the *MLE* has the same efficiency as the *MLE*. Thus all *MDE*s are first order efficient, and so are optimal under the model.

The second important role of the approximation (3.15) is in robustness. Let $t(x) = m_\theta(x)$ and the true model is $t_\epsilon(x)$, i.e., the model is contaminated by observations

at y in proportion ϵ. If we want to estimate the value $T(t) = \theta$ and the functional $T(d)$ is used, then the $\Delta T(\epsilon)$ represents the asymptotic bias in estimating θ under contamination. This means all the first order efficient estimators including *MDE*s have the same sensitivity to the contamination as the *MLE*.

Lindsay demonstrated how the approximation (3.15), and hence the influence function could be very misleading. The plot in example 1 of Lindsay (1994, p. 1083–1084) clearly shows that the linear influence function approximation for Hellinger distance works very poorly, even for extremely small ϵ. This example also shows evidence related to the results of Simpson (1987) that the Hellinger distance functional has an asymptotic breakdown point of 0.5 in the Poisson model.

Because of the inadequacy of the influence function when we consider the efficiency and robustness of some estimators, Lindsay considered an alternative to the influence function by looking at the second order Taylor approximation of $\Delta T(\epsilon)$.

Consider the ratio of the second order Taylor approximation ΔT_Q of $\Delta T(\epsilon)$ to the first ΔT_L. It is given by

$$\frac{\Delta T_Q}{\Delta T_L} = 1 + \frac{\epsilon}{2}\left[\frac{T''(y)}{T'(y)}\right].$$

This indicates that if ϵ is larger than $|T'(y)/T''(y)|$, then the two approximations differ by 50% or more.

Proposition 3.2 (Lindsay, 1994). *Let $I(\theta)$ be the Fisher information about scalar parameter θ in the model m_θ. For an estimator defined by an estimating function of the form* (3.10),

$$\left|\frac{T'(y)}{T''(y)}\right| = \left|I(\theta)[f_1(y) + A_2 f_2(y)]\right|^{-1} \tag{3.16}$$

where

$$f_1(y) = 2\nabla u(y,\theta) - 2E[\nabla u(X,\theta)] + T'(y)E[\nabla^2 u(X,\theta)], \tag{3.17}$$

and

$$f_2(y) = \frac{I(\theta)}{m_\theta(y)} + E[u(X,\theta)^3]\frac{u(y,\theta)}{I(\theta)} - 2u(y,\theta)^3. \tag{3.18}$$

Proof. See Lindsay (1994), Proposition 4. □

Corollary 3.3 (Lindsay, 1994). *If the model is a one-parameter exponential family and θ is the mean value parameter, then $f_1(y) = 0$ and so*

$$\left|\frac{T'(y)}{T''(y)}\right| = |A_2 Q(y)|^{-1}$$

where

$$Q(y) = \frac{1}{m_\theta(y)} + \frac{(y-\theta)E(X-\theta)^3}{\left[E(X-\theta)^2\right]^2} - \frac{(y-\theta)^2}{E(X-\theta)^2}. \tag{3.19}$$

Proof. See Lindsay (1994), Corollary 5. □

Proposition 3.2 and Corollary 3.3 show the important role of A_2 in the calculation of the ratio. If we look at the form of Q, the leading term becomes large for a small $m_\theta(y)$. Therefore, if we approximate Q by its leading term only, then we obtain the result that the quadratic and linear approximation to the bias caused by point contamination will differ substantially for ϵ with the order of magnitude of $m_\theta(y)/|A_2|$ or larger. Further, if $A_2 < 0$ then the quadratic approximation will predict 50% less bias, and if $A_2 > 0$ it will predict 50% more bias. Hence, the estimators with negative estimation curvature A_2 are robust, and $A_2 = -0.5$ for the Hellinger distance.

As a result of Corollary 3.3, if the influence function predicts a large influence for an observation y in an exponential family, the observation will actually have small influence on the *MDE*s.

3.1.6 Second Order Efficiency

All of the *MDE*s have the first order efficiency. Therefore the comparison between them should be done through higher orders. Let E_2 represent the second order efficiency of an estimator. The following proposition of Lindsay (1994) connects the *MDE* and the *MLE* in terms of second order efficiency. We know that the *MLE* is second order efficient in the multinomial model.

Proposition 3.4 (Lindsay, 1994). *Suppose the sample space* $\mathfrak{X} = \{0, 1, \ldots, K\}$ *is finite* $(K < \infty)$. *The second order efficiency of a minimum disparity estimator with RAF* $A(\delta)$ *is*

$$E_2(MDE) = E_2(MLE) + A_2^2 D$$

where D *is some nonnegative quantity depending on the model but not* $A(\delta)$.

Proof. See Lindsay (1994), Proposition 3. □

Proposition 3.4 states that the *MDE*s are second order efficient if $A_2 = 0$. However, the significance of Rao's second order efficiency is subject to some controversy. Berkson (1980) presented a lively discussion about this matter. Read and Cressie (1988) showed that the *MLE* is not necessarily second order optimal in the *PWD* family when a different second order optimal criterion is considered instead.

Lindsay (1994) offered a geometric description of the multinomial problems which shows that A_2^2 is a local summary measure of the estimator's lack of sufficiency. The

curvature parameter, A_2, is not a perfect measure of second order information loss, but it is a natural single number summary. Second order efficiency is achieved with $A_2 = 0$.

3.2 Continuous Case

The work of Lindsay (1994), which is discussed in the Section 3.1 is limited to the discrete case. Basu and Lindsay (1994) extended this procedure to the continuous case. Many of the properties of *MDE*s in the discrete case are shared by the *MDE*s in the continuous case.

Suppose $X_1, X_2, \ldots, X_n$ represent a random sample of size n from a continuous distribution M_θ with corresponding density m_θ, with respect to some dominating measure such as Lebesgue. Assume that M_θ is determined by its unknown parameter $\theta \in \Theta \subset \mathbb{R}^p$, where Θ is the parameter space and $\mathbb{R}^p$ is p–dimensional Euclidean space. Let F_n and f_n represent the empirical cdf and empirical pdf respectively. Because of the discrete nature of f_n, it is not possible to construct disparities between f_n and m_θ^* directly as in the discrete case. One way to overcome this problem is by using a nonparametric density estimator obtained from the data and then minimize the disparity between this density estimator and the model. We will denote the nonparametric density estimator of the data as f^*. It was Beran (1977b) who minimized the Hellinger distance between a kernel density estimator f^* and the model density m_θ^*.

In Beran's method, the choice of the sequence of kernel bandwidths becomes critical. The consistency of the kernel density estimator is very important in this case, and strong conditions have to be imposed on the kernel to make things work. Basu and Lindsay (1994) proposed convoluting the model with the same kernel as well. Let m_θ^* be the kernel integrated version of the model. Then Basu and Lindsay

minimized the disparity between f^* and m_θ^*. In terms of distance measure ρ, they minimized $\rho(f^*, m_\theta^*)$ with respect to θ to get the corresponding minimum distance estimator.

For a suitable kernel function $k(x, y, h)$, define $f^*(x)$, the kernel disparity estimator of the data, and $m_\theta^*(x)$, the kernel smoothed model density as

$$f^*(x) = \int k(x, y, h) dF_n(y),$$

$$m_\theta^*(x) = \int k(x, y, h) dM_\theta(y).$$

The kernel function $k(\cdot)$ can be the density of $N(y, h)$. The standard deviation h is the *smoothing parameter* or *bandwidth.* The smoothed version of model density $m_\theta^*(x)$ is also a density by itself. It will be assumed throughout the rest of this dissertation, that

$$\frac{\partial}{\partial\theta} \int m_\theta^*(x) dx = \int \frac{\partial}{\partial\theta} m_\theta^*(x) dx = 0.$$

Basu and Lindsay defined the Pearson residual in the continuous case in the same manner as in the discrete case, i.e.,

$$\delta^*(x) = \frac{f^*(x) - m_\theta^*}{m_\theta^*}.$$

Then they constructed the general form of the disparity ρ determined by G as

$$\rho_G(f^*, m_\theta^*) = \int G\Big(\delta^*(x)\Big) m_\theta^*(x)\, dx, \tag{3.20}$$

where G is a thrice differentiable convex function on $[-1, \infty)$ with $G(0) = 0$. This

guarantees that the disparity is nonnegative.

The minimum disparity version of the MLE, which will be denoted by MLE^*, can be obtained by minimizing the likelihood disparity: $\int \log\left(\frac{f^*(x)}{m_\theta^*(x)}\right) f^*(x)\,dx$. The natural counterparts of the other disparities in the discrete case are the Kullback-Leibler divergence (KL), the Pearson chi-squared (PCS), the Neyman's chi-squared (NCS), and the (twice-squared) Hellinger distance (HD). They are in turn constructed as follows:

$$KL(f^*(x), m_\theta^*(x)) = \int \log\left(\frac{m_\theta^*(x)}{f^*(x)}\right) m_\theta^*(x)dx,$$

$$PCS(f^*(x), m_\theta^*(x)) = \frac{1}{2}\int \log\left(\frac{f^*(x) - m_\theta^*(x)}{m_\theta^*(x)}\right) dx,$$

$$NCS(f^*(x), m_\theta^*(x)) = \frac{1}{2}\int \frac{f^*(x) - m_\theta^*(x)}{f^*(x)} dx,$$

$$HD(f^*(x), m_\theta^*(x)) = 2\int \left(\sqrt{f^*(x)} - \sqrt{m_\theta^*(x)}\right)^2 dx.$$

Basu and Lindsay also extended Cressie and Read's (1984, 1988) power weight divergence to its continuous analogue by defining

$$PWD(f^*(x), m_\theta^*(x)) = \int f^*(x) \frac{\left[f^*(x)/m_\theta^*(x)\right]^\lambda - 1}{\lambda(\lambda+1)} dx$$

$$= \int (\delta^*(x) + 1) \frac{\left(\delta^*(x) + 1\right)^\lambda - 1}{\lambda(\lambda+1)} m_\theta^*(x)dx.$$

Note that above disparities have already been standardized to make the residual adjustment function $A(\delta^*)$ have the same first order Taylor expansion as the likelihood

disparity.

By the differentiability with respect to θ under the model, we can obtain the estimator which minimizes the disparity (3.20) by solving the estimating equation

$$-\nabla\rho = \int \left[G'\big(\delta^*(x)\big) \frac{f^*(x)}{m_\theta^*(x)} - G\big(\delta^*(x)\big) \right] \nabla m_\theta^*(x) dx = 0. \tag{3.21}$$

We are looking at the negative of the derivative of the disparity for two reasons. First, to make the equation for the likelihood disparity maximize the likelihood, and second, to make the other minimum disparity estimators have the same first order term as that of the likelihood disparity.

Basu and Lindsay's proposal of minimizing the disparity between $f^*(x)$ and $m_\theta^*(x)$ has several advantages. Every reasonable minimum distance estimator is Fisher consistent for fixed h. Convergence of the sequence of the smoothing parameter to zero is not necessary anymore. With Basu and Lindsay's approach we can choose a particular kernel with a suitable fixed value of h and perform the analysis.

Let

$$A\big(\delta^*(x)\big) = G'\big(\delta^*(x)\big) \left(\frac{f^*(x)}{m_\theta^*(x)} - G\big(\delta^*(x)\big) \right)$$

so that

$$\int A(\delta) \nabla m_\theta^*(x) dx \tag{3.22}$$

is the estimating function corresponding to the estimating equation in (3.21). The strict convexity of $G(\cdot)$ implies that $A(\delta^*)$ is strictly increasing on $[-1, \infty)$. As in the discrete case, we can redefine the function A without changing its estimating properties so that $A(0) = 0$ and $A'(0) = 1$. This newly defined function $A(\delta^*)$ is the

residual function (RAF) in the continuous case.

3.2.1 The MLE^* and the Likelihood Disparity

The estimating equation of the likelihood disparity is given by

$$-\nabla LD = \int \frac{f^*(x)}{m_\theta^*(x)} \nabla m_\theta^*(x) dx = 0.$$

Since $\int \nabla m_\theta^*(x) dx = 0$, the above equation can be written as

$$\int \delta^*(x) \nabla m_\theta^*(x) dx = 0,$$

so that $A(\delta^*) = \delta^*$ for the LD. Generally, Taylor series expansion of the RAF $A(\delta^*)$ has the form

$$A(\delta^*) = \delta^* + A_2 \frac{\delta^{*2}}{2} + \ldots,$$

where $A_2 = A''(0)$. The coefficient of the second order term A_2 and all higher terms are equal to zero.

In the continuous case the MLE^* is important because it can be used for comparison to the MDEs. Basu and Lindsay (1994) presented several interesting results about distributions and the efficiency of the MDEs. They showed that the asymptotic distributions of all MDEs are (multivariate) normal with the same variance. They actually showed that the MDEs are asymptotically equivalent to the MLE^*. This asymptotic variance, however, is not necessarily equal to the inverse of the Fisher information, which means that the MDEs are not automatically first order efficient. But, Basu and Lindsay showed that MDEs can be first order efficient if one can choose a proper kernel.

Another important property of the MLE^* comes from the following Lemma of Basu and Lindsay (1994, Lemma 3.1). We need to introduce some more notation at this point. Let

$$\tilde{u}(x,\theta) = \nabla \log m_\theta^*(x) = \frac{\nabla m_\theta^*(x)}{m_\theta^*(x)}$$

and

$$u^*(t,\theta) = \int \tilde{u}(x,\theta)k(x;t,h)dx.$$

Lemma 3.5 (Basu and Lindsay, 1994). *Let f^* and m_θ^* be respectively the kernel density estimators obtained from the data and the smoothed model density. Let E_θ represent the expectation with respect to m_θ. Then the estimating equation of the MLE^* can be written as*

$$-\nabla LD(f^*, m_\theta^*) = \frac{1}{n}\sum u^*(X_i,\theta) = 0. \tag{3.23}$$

Further, $E_\theta\left[u^*(X,\theta)\right] = 0$ for all θ.

Proof. See Basu (1991, page 54). □

Note here we are assuming that $\int \nabla m_\theta^*(x)dx = 0$. We will call u^* the MLE^* score function and $u = u(x,\theta) = \nabla \log m_\theta(x)$ the MLE score function. Lemma 3.5 shows that MLE^* is an M-estimator so its efficiency and robustness are easily studied. Further, Lemma 3.5 tells us that the estimating equation of the MLE^* is an unbiased estimating equation.

3.2.2 Transparent Kernel

Since we are using smoothed version of the model density, a natural question is: how much information is lost by smoothing the original density function with the kernel? Basu and Lindsay illustrated that it is possible not to lose any information if the kernel is chosen properly.

Let's suppose that $X_1, X_2, \ldots, X_n$ are iid multivariate normal (MVN) with mean μ and variance Σ. Also suppose that the chosen kernel is $MVN(0, h^2 I)$. For multivariate normal m_θ^* is $MVN(\mu, \Sigma + h^2 I)$. The score equations for LD are

$$\mu: \qquad \int (\Sigma + h^2 I)^{-1}(x - \mu) dF^*(x) = 0$$

and

$$\Sigma: \qquad \int \left\{(x - \mu)(x - \mu)^T - (\Sigma + h^2 I)\right\} dF^*(x) = 0$$

where F^* is the cdf corresponding to f^*. Note that the distribution of X under F^* is the convolution of F_n and $MVN(0, h^2 I)$, hence the solutions to these score equations are just the usual maximum likelihood estimators, i.e.,

$$\hat{\mu} = \bar{X} \qquad \text{and} \qquad \hat{\Sigma} = \frac{1}{n} \sum (X_i - \bar{X})(X_i - \bar{X})^T.$$

Remarkably, the solution does not depend on the bandwidth h at all. Moreover, there is no loss of information. Such a kernel is called a *transparent kernel* by Basu and Lindsay. Here is a formal definition of the transparent kernel.

Definition 2 (Basu and Lindsay, 1994). The kernel $k(x; t, h)$ is defined to a trans-

parent kernel for $m_\theta^*(x)$ if the relation

$$Cu(X,\theta) + D = u^*(X,\theta)$$

hold for all $\theta \in \Theta$. Here θ is a p–dimensional parameter, C is a $p \times p$ nonsingular matrix which may depend on θ, and D is a p–dimensional vector.

We saw, however, that both u and u^* have expectation zero so that D must be zero. Hence we may define transparency by

$$Cu(X,\theta) = u^*(X,\theta). \tag{3.24}$$

The following lemma of Basu and Lindsay is the result of equation (3.24)

Lemma 3.6 (Basu and Lindsay, 1994). *Suppose that k is a transparent kernel for the family of models m_θ. Then the estimating equations for the MLE* of θ are equivalent to the ordinary maximum likelihood score functions for θ.*

Proof. See Basu (1991), Lemma 3.2 on page 56. □

Above lemma shows that if we can find a transparent kernel for the model, we can perform the kernel smoothing of densities without any loss of information. Now, the next question that aries is for which models do transparent kernels exist? Basu and Lindsay argued that there is no general answer to this, but they offered some examples showing that those exponential families which have closure under convolution have transparent kernels.

Proposition 3.7 (Basu and Lindsay, 1994). (i) *If $m_\theta(x)$ is the gamma density, with parameter (θ, λ), λ known, the kernel $k(y;x,h) = m_h(y-x)$ is transparent.*

(ii) *If* $m_\theta(x)$ *is the Poisson density, mean* θ*, then* $k(y;x,h) = m_h(y-x)$ *is transparent.*

3.2.3 Efficiency of the *MDE*s

Basu and Lindsay proved that the *MDE*s are asymptotically equivalent to the *MLE**s under the model. They showed that all of the minimum disparity estimators have the same influence function, using a similar approach to that used in the discrete case. They also showed that the asymptotic distributions of the *MDE*s are the same under sufficient regularity conditions. Further, *MDE*s are fully efficient when a transparent kernel is used. The idea of showing the robustness and efficiency of the *MDE*s is very similar to what we will discuss in the next chapter for weighted likelihood estimators. Hence, no further discussion will be given in this section.

3.2.4 Negative Exponential Disparity Family

Negative exponential disparity (*NED*) for the discrete case was discussed in Section 3.1.4. Basu *et al.* (1997) extended to the continuous case and investigate the properties of the *NED* along the lines of Lindsay's and Basu and Lindsay's work.

It was shown that the minimum negative exponential disparity generates a second order efficient estimator which shrinks both positive (outliers) and negative (inliers) residuals. Basu *et al.* established the robustness properties and the asymptotic properties of the *NED*.

As in Section 3.1.4, function $G(\delta) = e^{-\delta} - 1$ generates the negative exponential disparity whose *RAF* is given by

$$A(\delta) = 2 - (2+\delta)e^{-\delta}. \tag{3.25}$$

Basu *et al.* (1997), however, use $G(\delta) = e^{-\delta} - 2$ instead, because it produces above RAF directly without having to center $A(\delta) = (1+\delta)G'(\delta) - G(\delta)$. The negative exponential disparity measure in the continuous case is defined as

$$\int \left\{ \exp\left[-\frac{f^*(x)}{m_\theta^*(x)} + 1 \right] - 1 \right\} m_\theta^*(x)dx, \tag{3.26}$$

or, as

$$\int \left\{ \exp\left[-\frac{f^*(x)}{m_\theta^*(x)} + 1 \right] - 2 \right\} m_\theta^*(x)dx, \tag{3.27}$$

if we use $G(\delta) = e^{-\delta} - 2$.

Note that (3.26) and (3.27) differ only by a constant and have the same estimating properties.

Let $\mathcal{G}$ represent the class of continuous densities topologized by the L_2–norm, and let $\mathcal{F}_\theta = \{m_\theta; \theta \in \Theta\}$ be a parametric subclass of $\mathcal{G}$. Under suitable regularity conditions, Basu *et al.* showed the existence and consistency of the minimum negative exponential disparity estimator (*MNEDE*) and also that established the *MNEDE* is asymptotically fully efficient like the *MLE* under $\mathcal{F}_\theta$. For robustness they looked at the α–influence function to assess the minimum negative exponential disparity functional with respect to gross error model, as Beran (1977b) suggested, instead of looking at the influence function. They found that the *MNEDE* is robust against $100\alpha\%$ contaminated gross errors.

Family of the negative exponential disparities can be constructed by defining the RAFs as

$$A_{NED(a)}\big(\delta^*(x)\big) = \frac{(a+1) - \big[(a+1) + a\delta^*(x)\big]e^{-a\delta^*(x)}}{a^2} \tag{3.28}$$

where $\delta^* = [f^*/m_\theta^*] - 1$ and $a \in [0, \infty)$. The value $a = 1$ gives RAF of the usual negative exponential disparity. Further

$$A'_{NED(a)}(\delta^*) = (1+\delta^*)e^{-a\delta^*} \quad \text{and} \quad A''_{NED(a)}(\delta^*) = (1-a(1+\delta^*))e^{-a\delta^*}. \tag{3.29}$$

Hence, $A_2 < 0$, if $a > 1$, $A_2 = 0$, if $a = 1$ and $A_2 > 0$, if $0 \leq a < 1$.

Figure 3.2 shows that if $a < 1$, the RAF overweights the positive Pearson residual and downweights it if $a \geq 1$. Hence, according to Lindsay (1994), negative exponential disparity measures with $a \geq 1$ are robust against outliers. In fact, if $0 \leq \alpha < 1$, then the estimation curvature A_2 is positive and hence the *MNEDE*s are supposed to be nonrobust against outliers. Figure 3.3 shows the *RAF*s of the *NED* family with Pearson residual near 0, together with the *LD* and *HD*, and reveals how the $NED(a)$ treats the inliers ($\delta < 0$). As a moves toward 0, the $NED(a)$ downweights inliers more but overweights outliers more. Contrarily, the $NED(a)$ starts to act like a *HD* as a gets larger than 1. That is, the $NED(a)$ with $a > 1$, begins to stop downweighting inliers like the *HD*, but continues downweighting outliers.

Basu and Sarkar (1994c) showed heuristically that the efficiency of the *MHDE* and the *MNEDE* are quite close. Moreover, they provided empirical evidence that smoothing the model dramatically affects the improvement in efficiency of the scale estimator at the normal model. They are the first to apply and demonstrate the properties of the *NED* test. The simulation results of Basu and Sarkar (1994c) show that the *MNED* seems less efficient at the model but more robust than the *MHDE*. They considered this to be a suggestion that the curvature parameter is not a perfect measure in controlling the trade-off between robustness and efficiency.

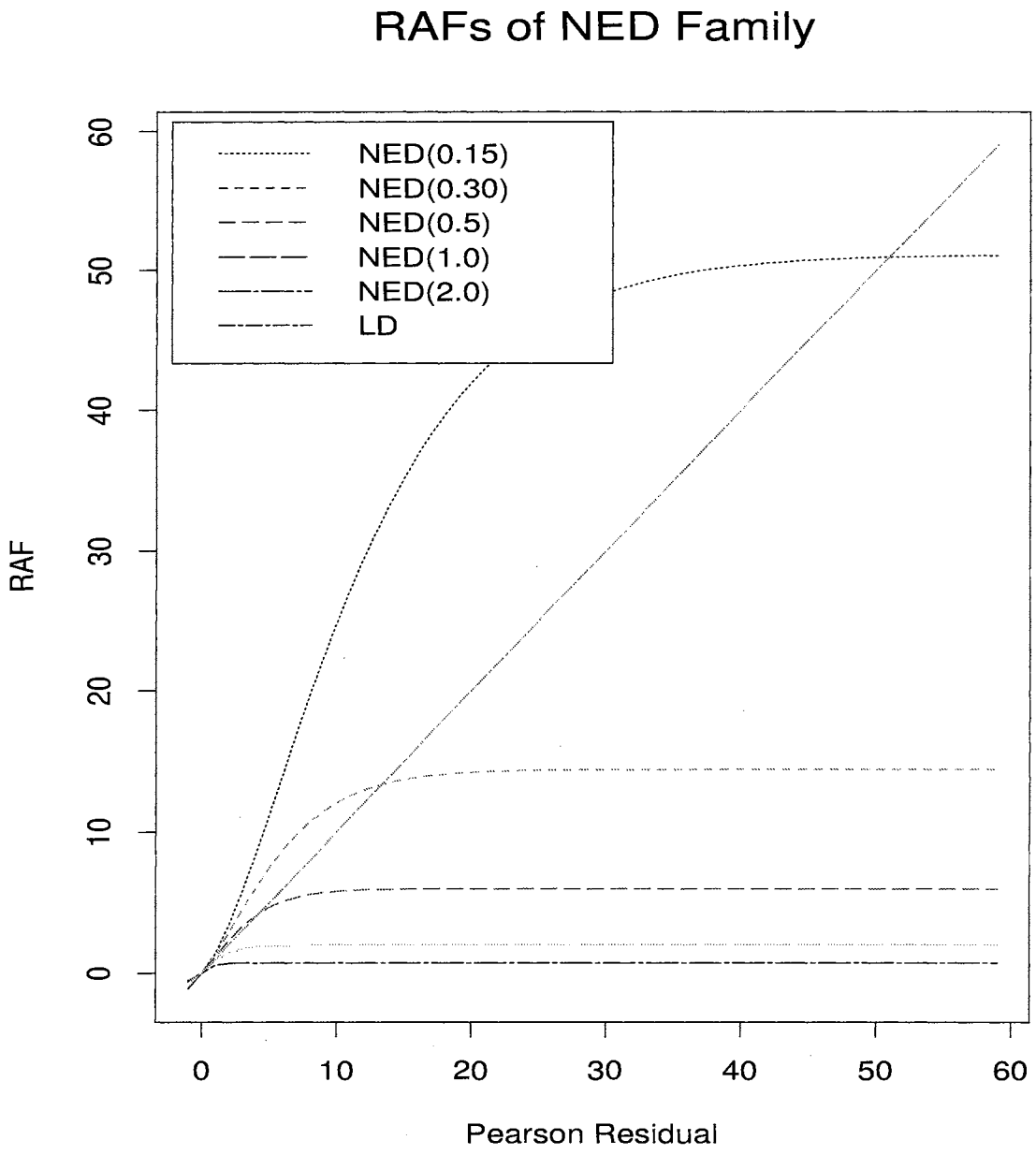

Figure 3.2: The residual adjustment functions of the family of negative exponential disparities ($NED(a)$) and the likelihood disparity (LD) for $-1 \leq \delta^* \leq 60$.

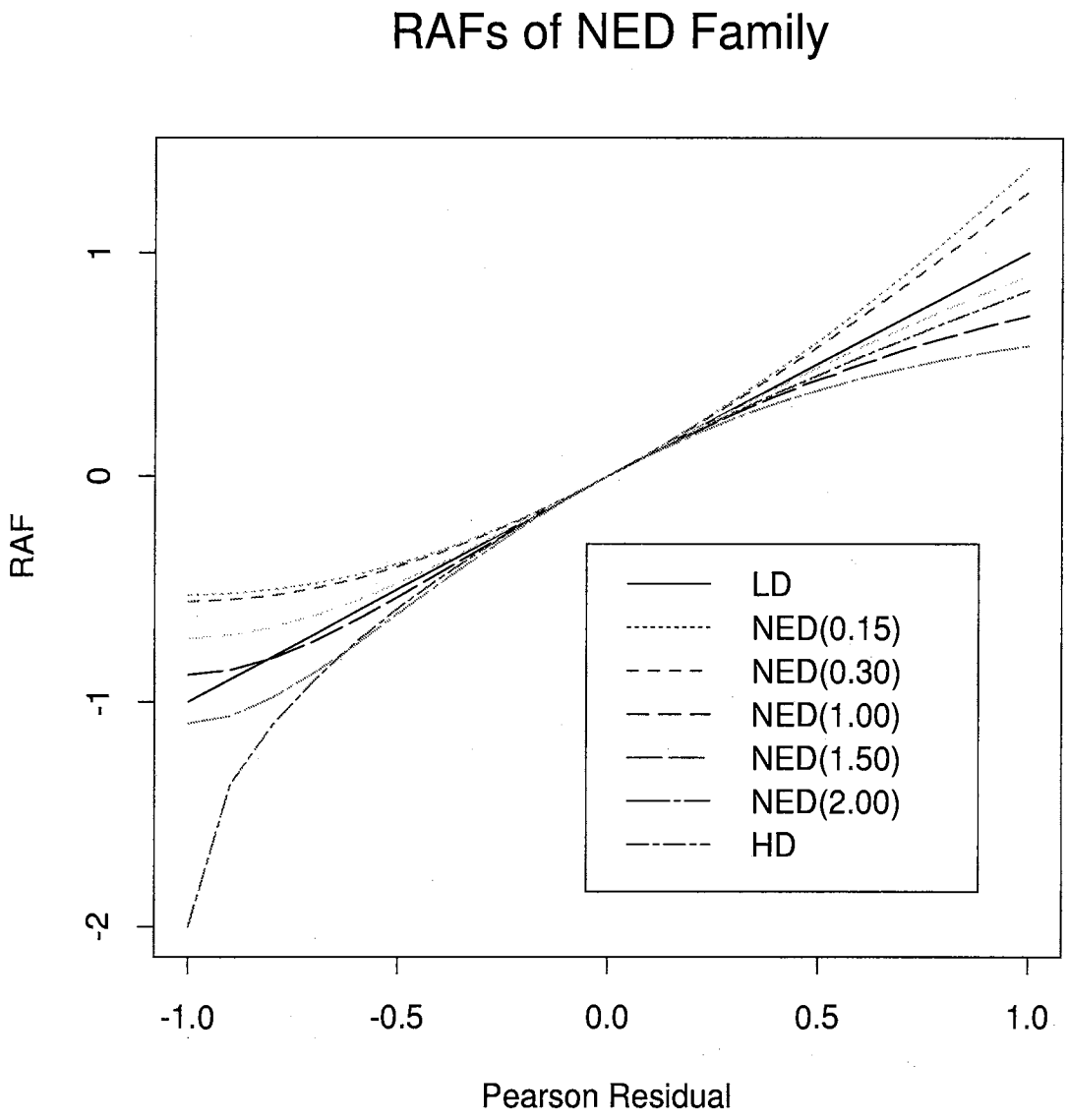

Figure 3.3: The residual adjustment functions of the family of negative exponential disparities ($NED(a)$), the likelihood disparity (LD) and the Hellinger distance (HD) for $-1 \leq \delta^* \leq 1$.

CHAPTER IV

WEIGHTED LIKELIHOOD ESTIMATION

The minimum disparity equations are usually nonlinear and have to be iteratively solved. In the case of continuous models this will involve handling several integrals which have to be evaluated numerically. Basu and Lindsay (1993, 1994) have developed an algorithm called the *iteratively reweighted likelihood equation* (*IRLE*) algorithm. The *IRLE* is similar in spirit to the iteratively reweighted least squares algorithm used in robust regression and makes the calculation of the minimum disparity estimators easier than usual methods like the Newton–Raphson algorithm.

The weighted likelihood estimation procedure consists of simply attaching weights, which depend on the Pearson residuals and the selected RAF, to the maximum likelihood score function, and solving the resulting set of estimating equations. The method avoids numerical integration completely by replacing integrals with sums over the data points. While this will not be minimizing a disparity, the results are remarkably similar, and the *IRLE* algorithm can be applied in the exact fashion.

The estimators obtained using weighted likelihood estimation are consistent, asymptotically normal and asymptotically efficient. Because of the asymptotic efficiency, the influence function of the estimators is generally unbounded. Despite this fact, the estimators are robust in terms of the probability model involved. Since the influence function is the product of the likelihood score function and the Fisher information matrix, it will be bounded if the score function is bounded.

4.1 The IRLE Method for Continuous Models

Since minimum disparity estimation requires iterative procedures, the numerical techniques should be applied and their rates of convergence are of great importance. The Newton–Raphson method becomes increasingly difficult to program as the number of parameters grows. This is because at each step we have to calculate and invert a Hessian matrix of the proper dimension. Basu and Lindsay (1993, 1994) proposed the method of iteratively reweighted likelihood equation to calculate the estimators. This is analogous to the method of iteratively reweighted least squares that is used in robust regression.

Assume that $\theta \in \Theta \subset \mathbb{R}^p$ and we are solving the estimating equation $\int A\big(\delta^*(x)\big)$ $\times \nabla m_\theta^*(x)\,dx = 0$. Since $\int \nabla m_\theta^*(x)\,dx = 0$ we can rewrite the estimating equations as

$$\int \frac{A\big(\delta^*(x)\big) - \lambda}{\delta^*(x)+1}\big(\delta^*(x)+1\big)\nabla m_\theta^*(x)\,dx = 0, \quad \text{for any constant } \lambda,$$

or

$$\int w(x)\frac{\nabla m_\theta^*(x)}{m_\theta^*} f^*(x)\,dx = 0, \tag{4.1}$$

where

$$w(x) = \big[A\big(\delta^*(x)\big) - \lambda\big]/\big(\delta^*(x)+1\big). \tag{4.2}$$

This is a weighted version of the defining equation of the MLE^* with weights $w(x)$. An initial estimate is chosen, and new weights $w(x)$ are created at each step to solve the estimating equations. When fast convergence is a consideration the optimal choice of λ is $\lambda = -1$. For this value, the rate of convergence of the *IRLE* at the

model is equivalent to that of the Newton–Raphson method, but the former is much easier to program.

4.2 The Weighted Likelihood Equation

Minimum disparity estimation via the *IRLE* algorithm has two problems when a random sample is available from a continuous parametric model. First, a transparent kernel may not be available. Second, the use of the *IRLE*, like the Newton–Raphson method, involves the use of numerical methods to evaluate the required integrals. Markatou, Basu and Lindsay (1995) and Basu, Markatou and Lindsay (1995) proposed new estimators which are similar to the minimum disparity estimators, but their first order efficiency does not depend on the choice of the kernel, and there are no integrals involved when solving their estimating equations. Markatou *et al.* (1995) discussed weighted likelihood estimation in the discrete case and Basu *et al.* (1995) in the continuous case. This section will follow mainly the paper of Basu *et al.* (1995).

Let $X_1, X_2, \ldots, X_n$ represent a random sample from a continuous distribution with density $m_\theta(x)$ having cdf M_θ. Let $u(x,\theta) = \nabla \log(m_\theta(x))$ be the score function. We assume that the *MLE* of θ is a solution of $\sum u(X_i,\theta) = 0$. Given any point y in the sample space, Basu *et al.* constructed a weight depending on y, a chosen model M_θ, and the sample empirical distribution F_n, say $w(y; M_\theta, F_n)$. Then considered solutions $\hat{\theta}_w$ to the weighted likelihood equations

$$\sum_{i=1}^{n} w_i u(X_i,\theta) = 0 \tag{4.3}$$

where $w_i = w(X_i; M_\theta, F_n)$. They selected weight function w such that $w(y; M_\theta, F_n)$ is 1, or close to 1, if the sample distribution F_n provides little or no evidence of the violation of the model M_θ in the neighborhood of y. Asymptotically, the weights

converge to 1 as F_n converges to M_θ. Therefore, the solution will be almost the same as the *MLE* if there are no unusual points in the data. If evidence from the data does indicate a lack of fit at or near y, then the weight will be less than one, with nearly zero weight given to wildly discrepant observations. Thus for a data set with a few extremely unlikely observations one expects a solution much like the *MLE* for the data without the extremely unlikely observations. In addition, the final fitted weights can be used to diagnose which data points were discrepant with the chosen model. Basu *et al.* employed the *IRLE* algorithm to solve the equation (4.3) and called the solution the weighted likelihood estimator (*WLE*).

4.3 Weights based on the Pearson Residual

4.3.1 The Discrete Case

A weighted likelihood approach to robust estimation has been developed by Markatou *et al.* (1995) in discrete models where the weights are the Pearson residual and are defined as

$$w(y; M_\theta, F_n) = w(\delta(y)) = \frac{A(\delta(y)) + 1}{\delta(y) + 1} \tag{4.4}$$

where A is a strictly increasing, twice differentiable function defined on $[-1, \infty)$, with the properties $A(0) = 0$ and $A'(0) = 1$. In particular, the function $A(\cdot)$ can be chosen as the *RAF* associated with a minimum disparity estimator. Since $\sum_y \nabla m_\theta(y) = 0$, and $u(y, \theta) = \nabla m_\theta(y)/m_\theta(y)$, one can rewrite the estimating equation (4.3) as

$$\sum \frac{A(\delta(y)) + 1}{\delta(y) + 1} u(y, \theta) d(y) = 0 \tag{4.5}$$

where $d(y)$ is the proportion of the sample observations with value $y \in \mathfrak{X}$. However, using $w(y; m_\theta, F_n) = w(\delta(y)) = [A(\delta(y)) + 1]/[\delta(y) + 1]$ and by rewriting the sum in y in terms of the sum in i, it can be seen that the estimating equation (4.5) is exactly equivalent to the estimating equation in (4.3). Thus defining the weights as in (4.4) guarantees that the weighted likelihood estimator is a root of the minimum disparity estimating equation in (3.10). This provides the rationale behind the choice of these weights, and helps establishing the connection between minimum disparity estimation and the weighted likelihood method. The solution of the weighted likelihood equation can be obtained iteratively by calculating new weights at each stage and solving the estimating equation, treating the weights as fixed constants.

The *RAF* of a disparity like the Hellinger distance, for which $A(\delta) \ll \delta$ for large δ, the weight function can severely downweight large Pearson residuals. For the sake of simplicity and interpretation, Basu *et al.* suggested that one can truncate the weights to constrain them in a closed interval $[0, 1]$. This can be done by using $w(x; M_\theta, F_n) = \min\{[A(\delta(x)) + 1]^+/[\delta(x) + 1], 1\}$. Asymptotically this makes no difference in the estimation procedure under the model. From the form of the weights it is clear that one can recover the *RAF* given the weights as

$$A(\delta) = -1 + (\delta + 1)w(\delta). \tag{4.6}$$

Lindsay (1994, equation 15) has shown that as long as the function $A(\delta)$ is increasing, the associated estimating equation (3.10) corresponds to a minimum disparity problem. That is, one can reconstruct a corresponding $G(\delta)$ function with the right properties. Since the truncation $w(x; M_\theta, F_n) = \min\{[A(\delta(x)) + 1]^+/[\delta(x) + 1], 1\}$ preserves the increasing nature of $A(\delta)$ in (4.6), the corresponding procedure still produces fully efficient estimates. More generally, one can consider high powers of

the weights and construct the estimating equation

$$\sum_i \{w(X_i, M_\theta, F_n)\}^p u(X_i, \theta) = 0. \tag{4.7}$$

The weights still converge to 1 asymptotically for any finite power p, but the amount of downweighting of discrepant cells increases with p. In this case, using Lindsay (1994, equation 15) one can generate a corresponding criterion function of the form (3.10) which asymptotically has a local minimum at the true parameter.

4.3.2 The Continuous Case

Basu *et al.* (1995) extended their previous work of the weighted likelihood estimation approach to the continuous case. The results of their work will be reviewed in this section.

Assuming the setup of Section 3.2, let the smoothed model family $m_\theta^*(x)$ satisfy $\int \nabla m_\theta^*(x)dx = 0$, i.e., the derivative with respect to θ can be taken inside the integral sign. Again using $w(x; M_\theta, F_n) = \min\{[A(\delta(x))+1]^+/[\delta(x)+1],\ 1\}$, and by doing the same manipulation as in Section 4.3.1, equation (3.22) can be rewritten as

$$\int w(x; M_\theta, F_n) \frac{\nabla m_\theta^*(x)}{m_\theta^*} dF^*(x) = 0, \tag{4.8}$$

where $F^*(x)$ represents the distribution function associated with the kernel density estimator $f^*(x)$. Although the above is a weighted version of the estimating equation of the likelihood disparity, it does not have a "weighted likelihood" interpretation, as both the model and the data have been smoothed with the kernel. In addition the estimating equations still involves an integral over the sample space rather than a sum over the observed data points. However, if we replace the smoothed empirical F^* with the unsmoothed empirical F_n, and $\nabla \log\left(m_\theta^*(x)\right)$ by the corresponding unsmoothed

version $\nabla \log\big(m_\theta(x)\big) = u(x,\theta)$, we get the estimating equation

$$\int w(x; M_\theta, F_n)\nabla \log\big(m_\theta(x)\big)dF_n = \frac{1}{n}\sum w(X_i, M_\theta, F_n)u(X_i,\theta) = 0. \tag{4.9}$$

The above equation still requires an iterative procedure to solve, however, but we have now avoided numerical integration over the range of the data. Now the kernel smoothing is involved *only* in the weight part, and not in the score part.

Basu *et al.* also showed how the curvature parameter and the smoothing parameter interact by examining the limiting distribution of the sum of weights when model is correct. Let $W_2 = -A_2$ and let $\tilde{w}^* = n^{-1}\sum w(X_i, M_\theta, F_n)$. Under suitable regularity conditions, the asymptotic distributions will satisfy:

$$2n(1-\tilde{w}^*) \approx W_2 \sum \delta^{*2}(X_i) \approx nW_2 \int \left[\frac{\int k(x;y,h)dF_n(x)}{m_\theta^*(y)} - 1\right]^2 dM_\theta(y).$$

Although this does not yield a simple limiting distribution, a simple calculation shows that the asymptotic mean of $2n(1-\tilde{w}^*)$ is

$$W_2\left[\int \frac{\int k^2(x;y,h)dM_\theta(x)}{m_\theta^*(y)}dM_\theta(y) - 1\right]. \tag{4.10}$$

The term in the brackets is a function of the smoothing parameter, but not the weight function. Using the illustration of Basu *et al.*, suppose that the model is normal, mean θ and variance σ^2, and the smoothing kernel is normal with variance h^2. The term in the brackets in the expression (4.10) is:

$$\frac{(\sigma^2+h^2)^{3/2}}{(3\sigma^2+h^2)^{1/2}h^2} - 1. \tag{4.11}$$

As h increases, $2n(1-\tilde{w}^*)$ moves toward zero, which corresponds to becoming more

like the *MLE* case. Now if h can be chosen as $h^2 = k\sigma^2$, where k is a constant, then relationship (4.11) becomes $[(1+k)^{3/2}][k(3+k)^{1/2}]$, indicating invariance of the method with respect to scale transformation. The discussion of Basu *et al.* for the selection of the smoothing parameter h will be reviewed in detail in Section 4.6.

Further investigation of Basu *et al.* showed how this calculation is related to the second order efficiency loss of the weight estimator. From the Cauchy-Schwarz inequality

$$\left[n^{-1}\sum(1-w_i)u_i\right]^2 \leq \left[n^{-1}(1-w_i)^2\right]\left[n^{-1}\sum u_i^2\right].$$

The second term on the right side converges to the Fisher information, while the calculation under the model assumption of the first term on the right side yields an asymptotic mean calculation similar to (4.10). Since the expectation of the left hand side is a measure of the information lost due to weighting, we see that this can be bounded in a simple fashion by restricting the variation in the weights.

4.4 Influence Functions and Standard Errors

Basu *et al.* investigated the influence function and standard errors of *WLE* under some further regularity conditions and showed that the *WLE* has full asymptotic efficiency under the model.

Suppose the weight function satisfies, for all x and F

$$w(x; M_\theta, F) \leq 1 \tag{4.12}$$

with equality for all x when $F = M_\theta$. Given a distribution F, the weighted likelihood

functional $\theta_w(F)$ will be chosen as a solution to the equation

$$\int w(x; M_\theta, F)u(x, \theta)dF(x) = 0. \tag{4.13}$$

Note that if $F = M_{\theta_0}$, then θ_0 is among the solutions to this equation, and so the method is Fisher consistent for θ if the root is chosen appropriately.

For a fixed distribution F, let $\theta_0 = \theta_w(F)$. Let $F_\epsilon(x)$ be the ϵ–contaminated distribution $(1-\epsilon)F(x) + \epsilon\Delta_t(x)$, $0 < \epsilon < 1$ and $\Delta_t(x)$ be the distribution that assigns mass 1 to the point t. If we let $F = M_{\theta_0}$, then for a smooth weight function we can expect

$$\frac{d}{d\epsilon} w(y; M_{\theta_w(F_\epsilon)}, F_\epsilon)\Big|_{\epsilon=0} = 0 \tag{4.14}$$

since $w(y; M_\theta, F)$ takes on its maximum value when $F = M_{\theta_0}$ (i.e., $\epsilon = 0$) if (4.12) holds. In this case, the influence function (IF) of the weighted likelihood functional is given by

$$\theta'_w(t) = \frac{d}{d\epsilon}\theta_w(F_\epsilon)\Big|_{\epsilon=0} = \left[\int -\nabla u(x; \theta_0)dM_{\theta_0}(x)\right]^{-1} u(y; \theta_0), \tag{4.15}$$

which is exactly the same as that of the maximum likelihood functional. It can be checked easily by taking a derivative on both sides of the equation

$$\int w(x; M_{\theta_w(F_\epsilon)}, F_\epsilon)u\Big(x, \theta_w(F_\epsilon)\Big)dF_\epsilon(x) = 0$$

with respect to ϵ and evaluating it at $\epsilon = 0$, in which case one gets $\theta'(t) = \mathcal{A}(F)\mathcal{B}(t)$,

where

$$\mathcal{A}(F) = \Big\{ \int w'\big(\delta(x)\big)\big(\delta(x)+1\big)u^*(x;\theta_0)u^T(x;\theta)dF(x)$$

$$- \int w\big(\delta(x)\big)\nabla u(x;\theta_0)dF(x)\Big\}^{-1}$$

and

$$\mathcal{B}(t) = w\big(\delta(x)\big)u(t;\theta_0) + \int w'\big(\delta(x)\big)\frac{k(x;t,h)}{m^*_{\theta_0}(x)}u(x;\theta_0)dF(x)$$

$$- \int w'\big(\delta(x)\big)\big(\delta(x)+1\big)u(x;\theta_0)dF(x).$$

Here $u^*(x;\theta) = \int \nabla \log\big(m^*_\theta(x)\big)k(y;x,h)dy$. If the model is correct, $w\big(\delta(x)\big) = 1$ and $w'\big(\delta(x)\big) = 0$ and hence the IF is the same as that of the *MLE*. Thus weighted likelihood method should provide, under some further regularity conditions, efficient estimates of the model parameters when the model is true. In addition, the covariance matrix Σ of $\sqrt{n}\theta_w(F_n)$ can be estimated by

$$\hat{\Sigma} = \mathcal{A}(F_n)\left[\frac{1}{n}\Big\{\mathcal{B}(X_i,F_n)\mathcal{B}^T(X_i,F_n)\Big\}\right]\mathcal{A}^T(F_n).$$

4.5 Robustness of the *WLE*

The robustness properties of the weighted likelihood methods are not found directly from the influence function at the model because the influence function of the weighted likelihood estimator will be always equal to that of the maximum likelihood estimator. Basu *et al.*, however, made a distinction between the procedures by looking at the second order approximation for the functional as in minimum disparity

estimation.

Let $\Delta\theta(\epsilon)$ be the change in the functional $\theta(\cdot)$ due to ϵ–contamination. i.e., $\Delta\theta(\epsilon) = \theta(F_\epsilon) - \theta(F)$ represents the bias of the method. A simple calculation shows that for point mass contaminations the ratio of the quadratic to the linear approximation of $\Delta\theta(\epsilon)$ equals $1+\big(\theta''(y)/\theta'(y)\big)(\epsilon/2)$. Hence, if the amount of contamination ϵ is greater than $\epsilon_{crit} = |\theta'(y)/\theta''(y)|$, the two approximations differ by 50% or more. If $\theta'(y)$ and $\theta''(y)$ have opposite signs then the quadratic approximation will predict with less bias. The estimation curvature A_2 plays a pivotal role in the computation of this ratio.

In particular, let $I(\theta)$ be the Fisher information about a scalar parameter θ in model m_θ. For an estimator defined by the estimating equation of the form (4.3),

$$\epsilon_{crit} = \Big|I(\theta)\big(f_1(t) + A_2 f_2(t)\big)\Big|^{-1}$$

where

$$f_1(t) = 2\nabla u(t;\theta) - 2E_\theta\big[\nabla u(X;\theta)\big] + \theta'(y)E_\theta\big[\nabla^2 u(X;\theta)\big],$$

$$f_2(t) = \frac{I(\theta)}{u(t,\theta)}\int \frac{k^2(x;t,\theta)u(x,\theta)}{m_\theta(x)}dx$$

$$- 2\int u^2(x,\theta)k(x;t,h)dx + \theta'(t)\int u^3(x,\theta)m_\theta(x)dx$$

and

$$I(\theta) = \int u^2(x,\theta)m_\theta(x)dx.$$

For the mean parameter of the exponential family model $f_1(t)$ is zero, and $\theta''(t) = \theta' A_2 f_2(t)$. Thus $\theta'(t)$ and $\theta''(t)$ have opposite signs if $A_2 f_2(t) < 0$.

Basu *et al.* presented some numerical values (Basu *et al.*, 1995; Table 1) of $f_2(t)$ for the various values of h ($0.05 \leq h \leq 0.95$) and various values of t ($2.0 \leq t \leq 4.5$) at the normal model when the true distribution is $N(0,1)$ with a normal kernel function. They showed that $f_2(t)$ is positive for all the cases considered, guaranteeing a less bias for the second order prediction for disparities with $A_2 < 0$ (such as *HD*). The value of f_2 decreases as h increases, indicating that care should be taken in selecting the value of h to achieve robustness. For more details see Lindsay (1994, p. 1091) and Basu and Lindsay (1994, pp. 697-698). Basu *et al.* also noted that the larger the outlying t values, the smaller the value of h needed to achieve robustness.

Basu *et al.* also commented that there are technical difficulties in defining breakdown points when the estimator is defined as a solution to an estimating equation and established the outlier stability of the weighted likelihood estimating equations under appropriate conditions based on Lindsay's arguments about the outlier stability of the minimum disparity estimating equations.

Consider a fixed model $m_\theta(x)$ and contamination level ϵ. Let $\{\xi_j : j = 1, 2, \ldots\}$ be a sequence of elements of the sample space $\mathfrak{X}$. Let F_n represent the empirical distribution function. Let the ϵ-contaminated data $F_{n,j}(x) = (1-\epsilon)F_n(x) + \epsilon\Delta_{\xi_j}(x)$, and $F_{n,\epsilon}(x) = (1-\epsilon)F(x)$. Also, let $f_j^*(x) = \int k(x;t,h)dF_{n,j}(x)$ and $f_\epsilon^*(x) = (1-\epsilon)f^*(x)$. Furthermore, let the corresponding Pearson residuals be denoted as $\delta_j(x) = f_j^*(x)/m_\theta^*(x) - 1$, and $\delta_\epsilon^*(x) = f_\epsilon^*(x)/m_\theta^*(x) - 1$.

Definition 3 (Lindsay, 1994). We will say that $\{\xi_j\}$ constitues an outlier sequence for the model $m_\theta(x)$ and data $F_n(x)$ if $\delta_j(\xi_j) \to \infty$, and $f^*(\xi_j) \to \infty$ as $j \to \infty$.

Lemma 4.1 (Lindsay, 1994). *The sequence $\{\xi_i\}$ constitues an outlier sequence if and only if $f^*(\xi_j) \to 0$ and $m_\theta(\xi_j) \to 0$ as $j \to \infty$.*

Proof. See Lindsay (1994), Lemma 9. □

Proposition 4.2 (Lindsay, 1994). *If for some* $k > 1$, $E_\theta\left[|u(X;\theta)|^k\right] < \infty$ *for all* θ, $A(-1)$ *is finite and* $A(\delta) = O(\delta^{(k-1)/k})$ *as* $\delta \to \infty$, *and* $A(\cdot)$ *is outlier stable for all model* m_θ.

Proof. See Lindsay (1994), Proposition 14 and Corollary 15. □

Let $\tilde{u}(x;\theta) = \nabla \log\left(m_\theta^*(x)\right)$. The following theorem of Basu *et al.* provides the outlier stability of the weighted likelihood estimating equation. This theorem is a simple generalization of Proposition 4.2 of Lindsay (1994).

Theorem 4.3 (Basu, Markatou, and Lindsay, 1995). *Assume that*

(i) $E_{m_\theta^*}\left[|\tilde{u}(X;\theta)|\right]$ *is finite for all* θ,

(ii) *for some* $k > 1$, $E_{m_\theta^*}\left[|\tilde{u}(X;\theta)|^k\right] < \infty$ *for all* θ; $A(\delta) = O(\delta^{(k-1)/k})$ *as* $\delta \to \infty$,

(iii) $\left|\dfrac{u(\xi_j;\theta)}{\tilde{u}(\xi_j;\theta)}\right|$ *remains bounded as* $j \to \infty$,

(iv) *the kernel function* $k(\cdot)$ *is bounded,*

(v) $A(-1)$ *is finite.*

Then the estimating function displays the following convergence:

$$\int w(x; F_{n,j}, M_\theta)u(x;\theta)dF_{n,j} \to \int w(x; F_{n,\epsilon}, M_\theta)u(x;\theta)dF_{n,\epsilon}.$$

Above theorem states that if a datum from original data set is moved toward infinity, then the estimator of the mean shifts at first with the datum, but then returns to the value it would have had if the point had been deleted from the data. Basu *et al.* noted that condition (iii) is generally satisfied in the exponential family

under bounded kernels and condition (iv) is satisfied, for example, by the normal family of kernels.

4.6 The Selection of the Smoothing Parameter

Basu *et al.* recommended choosing the smoothing parameter h^2 as $k\sigma^2$, where k is some positive constant, based on the following reasons.

It is a desirable property that the weighted likelihood procedure should vary appropriately under transformation of the data and the model in the location scale model. One way to maintain location scale equivalence is to select h based on a robust equivalent estimator of scale. However, the Pearson residuals are unduly affected by the magnitude of the outlier in the sense that the asymptotic residual at an outlying point t depends on the variance of the underlying assumed model. To illustrate, assume the true model is $F_\epsilon = (1-\epsilon)N(0,\sigma^2) + \epsilon\Delta_t(x)$ where $N(0,\sigma^2)$ is the nominal model. Let the kernel function be $N(0,h^2)$. Then the asymptotic Pearson residual at t is

$$\delta(t) = \epsilon\left\{\left(\frac{\sigma^2+h^2}{h^2}\right)^{1/2}\exp\left[\frac{t^2}{2(\sigma^2+h^2)}\right]-1\right\}.$$

Here $\delta(t)$ is a function of σ^2 so the Pearson residual is obviously affected by the magnitude of the outlier. To overcome this drawback of the Pearson residual, select h as a fraction of σ, i.e., $h^2 = k\sigma^2$, where k is some positive constant. Then for $y = c\sigma$, where c is some constant, the asymptotic Pearson residual at y is

$$\delta(y) = \epsilon\left\{\left(\frac{1+k^2}{k^2}\right)^{1/2}\exp\left[\frac{c^2}{2(1+k^2)}\right]-1\right\}.$$

The Pearson residual, $\delta(y)$, in this case is independent of σ^2, so that the choice of k

can be made independently of the scale parameter of the model.

To calculate the value of k, Basu *et al.* solved the weight function for k. That is, they fixed the amount of contamination ϵ at some reasonable value ($\epsilon = 0.25$). Then they chose a weight function w and solved for k in the equation $w(\delta(y)) = w_0$, where w_0 is a number significantly smaller than 1. This guarantees that the weight assigned to an observation that is at least $c\sigma$ units away from the mean of the model with a contaminating proportion ϵ or greater is at most w_0.

CHAPTER V

TESTS FOR EQUALITY OF TWO POPULATION MEANS

5.1 Introduction

Traditionally, a statistical test has been called robust if both its Type I error rate (significance level) and its power are not affected drastically by departure from normality. Many tests have been developed and proposed to achieve robustness against not only nonnormality, but also departure from equal variances. In this chapter we will discuss robust tests which are based on *WLE*s.

Suppose $X_{i1}, X_{i2}, \ldots, X_{in_i}$ is an independent random samples from populations with μ_i and variance σ_i^2, $i = 1, 2$. A common statistical procedure is to compare population means by testing

$$\mathrm{H}_0 : \mu_1 = \mu_2.$$

For this problem, the independent two sample Student's t test is the usual tool under the assumption of normality and equal but unknown variances. The behavior of the t statistic against the departure from the normality and/or the homogeneity assumptions has been studied by numerous authors. In the early stages (Daniels, 1938), Student's t test was considered nearly immune to violations of assumptions other than independence. Studies by Bartlett (1935), Box and Anderson (1955), Cochran (1947), Gayen (1950) and others presented evidence that suggest departure from normality does not cause the actual Type I error rate to differ greatly from the nominal level α. In addition, studies of Box (1953, 1954a), Horsnell (1953), Welch (1938) and others showed the departure from equality of variances does not affect the

Type I error rate of Student's t tests either, unless the departure is so severe it can be identified by a simple inspection of the data.

Boneau (1960) and Havlicek and Peterson (1974) found that the relationship between actual Type I error rates and the nominal Type I error rates is influenced by the sample size when the population variances are unequal. Scheffé (1959) had already shown that the t statistic is asymptotically normal even under nonnormal populations or unequal variances. On the other hand, if the sample sizes are unequal, Cressie and Whitford (1986) argued that the t statistic, under heteroscedasticity, is not even asymptotically correct. Whenever sample sizes are large and approximately the same, or moderately large and equal, the actual Type I error rate and the nominal Type I error rate (α) are near one another. However, the degree to which the sample sizes must be of similar size is unclear. Wilcox (1990, 1994) showed that there are situations where nonnormality can seriously affect Type I error error rates. In addition, nonnormality can have a substantial effect on the power of methods for comparing means (Hampel, Ronchetti, Rousseeuw, and Stahel, 1986). Further, Ramsey (1980) presented the boundary conditions where the t test fails to be robust to violations of the equal population variance assumption even when equal sized samples come from normal populations. Results of Boneau (1960), Havlicek and Peterson (1974), Pratt (1964) and Scheffé (1959) showed that Student's t produces actual Type I error rates less than α (i.e., conservative) when the larger sample size is associated with the larger population variance, sometimes known as *positive condition* . Conversely, the t test produces actual Type I error rates larger than α (i.e., liberal) if the larger sample size is coupled with the smaller population variance, sometimes known as *negative condition* (Coombs, Algina and Oltman; 1996). Wilcox, Charlin and Thompson (1986) verified these patterns using a Monte Carlo study.

These patterns are expected if we look at the t test statistic. Suppose $\sigma_1^2 > \sigma_2^2$

and $n_1 > n_2$. Then the pooled sample variance

$$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_2 + n_2 - 2} \tag{5.1}$$

is more heavily weighted by s_1^2. Consequently, a relatively larger value of s_p^2 than when $n_1 > n_2$ and $\sigma_1^2 < \sigma_2^2$ makes the calculated t values smaller, causing a decrease in the power of t. In the following sections, we will discuss some alternatives to the Student's t test when the nomality and/or equal variance assumptions are violated.

5.2 Alternatives to Student's t Test

Over the years, many robust tests as alternatives to the Student's t test have been proposed. Behrens, in 1929, was the first to consider a solution to the problem of testing the equality of two population means without assuming equality of population variances. Fisher (1935, 1939) derived Behrens' solution in terms of his fiducial distribution. Thus this problem is now known as the Behrens-Fisher problem. Behrens-Fisher's solution to this problem of comparing means when the variances are different is not acceptable to many statisticians, however, because the actual Type I error rate of the associated test is often less than the nominal level (Best and Rayner, 1987). Numerous solutions to the Behrens-Fisher problem have been developed by statisticians like Welch (1938, 1947), Aspin(1948, 1949), Banerjee (1960, 1961), Cochran (1964), Yuen (1974), Tiku and Singh (1981) and Wilcox (1992) among others. Some of these tests will be discussed in the following two sections.

5.2.1 Behrens-Fisher Problem and Welch's Tests

Welch (1938) discussed three tests: u, v and z. u is the usual t statistic with pooled sample variance (s_p^2) and v is

$$v = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{5.2}$$

which became known as the univariate Behrens-Fisher statistic (Nel, van der Merwe, and Moser, 1990). When sample sizes are equal ($n_1 = n_2$), u and v are equivalent. Welch suggested two types of critical values: approximate degrees of freedom (APDF) and series solutions. The APDF procedures approximates the degrees of freedom which specify the sampling distribution of the test statistic. Series solutions are derived by utilizing a series expansion to determine the critical value for the rejection region. The test using the estimator for degrees of freedom f is known as *Welch's test,* where f is determined by

$$\frac{1}{f} = \frac{c^2}{n_1 - 1} + \frac{(1-c)^2}{n_2 - 1}, \qquad c = \frac{\sigma_1^2/n_1}{\sigma_1^2/n_1 + \sigma_2^2/n_2}. \tag{5.3}$$

Fairfield-Smith (1936) gave an approximation similar to Welch's.

Note that Scheffé (1943), Banerjee (1960), James (1965), and Ostle and Mensing (1975) have also derived approximations to the distribution of the test statistic. Nel, van der Merwe and Moser (1990) derived exact distribution of the Behrens-Fisher statistic by replacing σ_i^2 with its corresponding unbiased estimator s_i^2, $i = 1, 2$, which is similar to the approach of Welch's (1947) series expansion. The significance of the difference between the two means will tend to be underestimated when $\sigma_1^2 < \sigma_2^2$ and overestimated when $\sigma_1^2 > \sigma_2^2$, whenever the u (Student's t) test is used. Welch mentioned that when the smaller sample comes from the more variable population

(the negative condition) the effective number of degrees of freedom f is liable to be much smaller than $n_1 + n_2 - 2$. Even when $\sigma_1^2 = \sigma_2^2$ the effective degrees of freedom is still much smaller than the degrees of freedom of the u (Student's t) test.

Welch (1947) then expressed the series critical values for v as a function of s_1^2, s_2^2 and the nominal level α and developed a series critical value in powers of $(n_i - 1)^{-1}$. Welch's second order series critical value for two samples case is given by

$$h_\alpha(s^2) = z_\alpha + z_\alpha \frac{(1+z_\alpha^2)}{4} \sum_{i=1}^{2} \frac{c_i^2}{f_i} - z_\alpha \left[\frac{(1+z_\alpha^2)}{2} \sum_{i=1}^{2} \frac{c_i^2}{f_i^2} + \frac{(3+5z_\alpha^2+z_\alpha^4)}{3} \sum_{i=1}^{2} \frac{c_i^3}{f_i^2} - \frac{(15+23z_\alpha^2+9z_\alpha^4)}{32} \left\{ \sum_{i=1}^{2} \frac{c_i^2}{f_i^2} \right\}^2 \right] \tag{5.4}$$

where

$$P[v < h_\alpha(s^2)] = 1 - \alpha, \quad f_i = n_i - 1, \quad \text{and} \quad \Phi(z_\alpha) = 1 - \alpha.$$

This critical value can be extended for testing equality of more than two means in a straightforward manner. Welch also gave the APDF critical values for the $k > 2$ samples case. Aspin (1948, 1949) extended Welch's (1947) series solution up to the fourth order term and investigated, for equal sample sizes, variation in the critical values of each term. This series solution has become known as the Aspin-Welch test.

Welch (1938) also proposed, when $n_1 \neq n_2$ and both are greater than 3, that

$$z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{(n_1-1)s_1^2}{n_1(n_1-3)} + \frac{(n_2-1)s_2^2}{n_2(n_2-3)}}}. \tag{5.5}$$

The null hypothesis is rejected if $|z| > \nu_0$ where ν_0 is the larger of $\sqrt{(n_1-3)/(n_1-1)}\, t_{\alpha/2}$ and $\sqrt{(n_2-3)/(n_2-1)}\, t_{\alpha/2}$, and $t_{\alpha/2}$ is $1 - \alpha/2$ percentile of t distribution with ap-

proximate degrees of freedom f in (5.3).

Welch explained that this z test is less dependent on σ_1^2/σ_2^2 than v and produces actual Type I error rates closer to the nominal level than either u or v. Note that the z and the v tests are equivalent when $n_1 = n_2$. Fenstad (1983) compared the z test to the v test in the case of $\sigma_1^2/\sigma_2^2 = 1/4$ and $\sigma_1^2/\sigma_2^2 = 4$ and highly recommended Welch's z test for the Behrens-Fisher problem. Fenstad also argued that the chosen values of σ_1^2/σ_2^2 are not extreme at all. Best and Rayner (1987), however, argued that their simulation results show no evidence to suggest the actual z Type I errors are better than the actual v type I error sizes.

Wang (1971) investigated the Type I error rates associated with use of the Welch test and the Aspin-Welch test and showed satisfactory agreement of the Welch test to the desired level. Best and Rayner (1987) conducted a comparative study on Welch's test with the Wald, likelihood ratio and score statistics. They noted that the Welch test compared well with the usual Student's t test when variances are equal and when $f \geq 5$. Further, on the basis of size and power studies, they recommended routine use of the Welch test whenever data are such that $f \geq 5$. Ryan, Joiner, and Ryan (1982) also agreed with Best and Rayner's suggestion. Davenport and Webster (1975) used numerical integration and a method proposed by Cochran (1951) to compute the Type I error rate for v^2. They summarized that, where $\vartheta = \sigma_1/\sigma_2$,

- The size of the Welch test v approaches the nominated level when $(n_2\vartheta^2/n_1) \to \infty$ or $(n_2\vartheta^2/n_1) \to 0$.

- The size of the test v for given $(n_2\vartheta^2/n_1)$, n_1, n_2 is the same as the test for $n_1/(n_2\vartheta^2)$, n_1, n_2.

- When $(n_2\vartheta^2/n_1) > 1$ and $n_1 > n_2$, the size of the v test is approximately equal to the nominal level.

- When $(n_2\vartheta^2/n_1) > 1$ and $n_1 < n_2$, then size of the test can be somewhat different from the nominal level, the difference depending on $(n_2\vartheta^2/n_1)$ and increasing as $|n_2 - n_1|$ increases.

Best and Rayner (1987) also have obtained Monte Carlo Type I error rates similar to the results of Davenport and Webster.

James (1951, 1954) generalized the Welch (1947) series solutions to the k samples case and multivariate case, respectively. In response, Welch (1951) also gave another approach as an alternative to James's (1951). These tests will be discussed in Chapter 7.

5.2.2 Yuen's Test

Tukey and McLaughlin (1963) proposed one-sample trimmed t statistic which uses the ratio of the trimmed mean and the Winsorized sum of squared deviations of the sample as a treatment for long-tailed distributions like the mixture of two normal distributions. The g-times trimmed and Winsorized means are defined, respectively, as

$$\bar{X}_{tg} = \frac{1}{n-2g}\left(X_{g+1} + \cdots + X_{n-g}\right) \tag{5.6}$$

$$\bar{X}_{wg} = \frac{1}{n}\left\{(g+1)X_{g+1} + X_{g+2} + \cdots + X_{n-g-1} + (g+1)X_{n-g}\right\}. \tag{5.7}$$

The g-times Winsorized sum of squared deviations is

$$\begin{aligned} SSD_{wg} = {} & (g+1)(X_{g+1} - \bar{X}_{wg})^2 + (X_{g+2} - \bar{X}_{wg})^2 + \ldots \\ & + (X_{n-g-1} - \bar{X}_{wg})^2 + (g+1)(X_{n-g} - \bar{X}_{wg})^2. \end{aligned} \tag{5.8}$$

It was shown that the family of trimmed means performs fairly well over a very wide spectrum of underlying distributions (Andrews *et al.* 1972). Yuen and Dixon (1973) extended Tukey and McLaughlin's work to the two-sample cases under the assumptions of normality, equal variance, and independence; the test statistic is given by

$$t_{t/w} = \frac{\bar{X}_{1tg} - \bar{X}_{2tg}}{\left\{\frac{SSD_{1wg} + SSD_{2wg}}{h(h-1)}\right\}^{\frac{1}{2}}} \tag{5.9}$$

where $h = n - 2g$, $\bar{X}_{itg}$ and SSD_{iwg} are defined in (5.6) and (5.8) for i =1,2. Yuen and Dixon (1973) compared the power and power efficiency functions of the trimmed t relative to the Student's t. Numerical results were presented for sample sizes of 5, 7, 10 and for $\delta = 0$, 1, 3 and 5, where $\delta = |\mu_1 - \mu_2|/\{\sigma(2/n)^{\frac{1}{2}}\}$. The noncentrality parameter, δ, is often called the *effect size parameter*, especially by researchers in the behavioral sciences (Coombs, *et al.*, 1996; Wilcox, 1995a). Yuen and Dixon noted that the two-sample trimmed t can be satisfactorily approximated by the Student's t with degrees of freedom corresponding to the the reduced sample. They argued that the gain due to the trimmed t can be appreciable for long-tailed distributions while the loss of power efficiency is small under the normal. Yuen and Dixon only considered symmetric distributions in the Monte Carlo study. Yuen and Dixon (1973) also suggested an alternative for unequal samples sizes, which is given by

$$t_{t/w} = \frac{\bar{X}_{1tg} - \bar{X}_{2tg}}{\left\{\left(\frac{SSD_{1wg} + SSD_{2wg}}{h_1 + h_2 - 2}\right)\left(\frac{1}{h_1} + \frac{1}{h_2}\right)\right\}^{\frac{1}{2}}} \tag{5.10}$$

where h_1=$n_1 - 2g$, h_2=$n_2 - 2g$. $t_{t/w}$ is approximately t distributed with $h_1 - h_2 - 2$ degrees of freedom under the null. Yuen and Dixon (1973) reported that for small

samples and mild contamination, the trimmed t produces lower power than Student's t. However, for sample size at least 10, the trimmed t does turn out to be generally superior.

Further, Yuen (1974) proposed a trimmed t test for unequal population variances. The test is defined by

$$t^*_{t/w} = \frac{\bar{X}_{1tg} - \bar{X}_{2tg}}{\left\{ \frac{s^2_{1wg}}{h_1} + \frac{s^2_{2wg}}{h_2} \right\}^{\frac{1}{2}}} \tag{5.11}$$

where $s^2_{wg} = SSD_{wg}/(h-1)$, and has approximately a t distribution with degrees of freedom f^* calculated from

$$\frac{1}{f^*} = \frac{c^{*2}}{h_1 - 1} + \frac{(1-c^*)^2}{h_2 - 1}, \qquad c^* = \frac{s^2_{1wg}/h_1}{s^2_{1wg}/h_1 + s^2_{2wg}/h_2}.$$

Yuen conducted Monte Carlo studies to compare her test to Welch's (1938) under various long-tailed, short-tailed and mixed distributions. She chose the set of variance ratios $\vartheta = \sigma_1/\sigma_2$ as 0.25, 0.5, 2 and made comparison under $\delta = 0$, 1, 2, where $\delta = |\mu_1 - \mu_2|/(\sigma_1^2/n_1 + \sigma_2^2/n_2)^{1/2}$. Yuen noted that the power of trimmed t never exceeded the corresponding power of Welch's v under exact nomality. For long-tailed or contaminated with outliers, she strongly recommended the trimmed t.

Wilcox (1994) gave some results for trimmed mean when distributions are skewed in terms of Cornish-Fisher expansion and simulation. He claimed that confidence intervals based on the trimmed mean are more accurate than confidence intervals based on the arithmetic mean whether the underlying distribution is symmetric or not. Wilcox (1995b, 1997) also presented a bootstrap approach in calculating the significance level for the hypothesis test and the confidence intervals for pairwise comparisons of trimmed means. The bootstrap method of Efron (1987) will be dis-

cussed briefly in the next section. Hall (1986, 1988) presented detailed discussion of bootstrap confidence intervals.

5.2.3 One-Step M-Estimator and the Bootstrap Confidence Intervals

Wilcox (1992) proposed one-step M-estimators of locations with the bootstrap confidence interval approach of Efron (1982, 1987). Let $X_1, X_2, \ldots, X_n$ be a random sample, and let ψ be an odd and nonincreasing function. An M–estimate of location is any quantity $\hat{\mu}_m$, satisfying

$$\psi\left(\frac{x_i - \hat{\mu}_m}{\hat{\zeta}}\right) = 0,$$

where $\hat{\zeta}$ is some measure of scale. Let $\hat{\zeta} = 1.48 \cdot \text{MAD}$, where MAD=median$\{|X_i - M|\}$ and M=median(X_i). By choosing Huber's (1964) ψ function, i.e., $\psi(x) = \max\{-k, \min(k, x)\}$, and $k = 1.28$, the one-step M-estimator — one iteration of the Newton method — $\hat{\mu}_m$ is obtained as follows (for more summary, see Staudte and Sheather, 1990):

$$\hat{\mu}_m = M + \hat{\zeta}\frac{\sum\psi\left(\frac{x_i - M}{\hat{\zeta}}\right)}{\sum\psi'\left(\frac{x_i - M}{\hat{\zeta}}\right)}$$

$$= \frac{k\hat{\zeta}(i_2 - i_1) + \sum_{i=i_1+1}^{n-i_2} X_{(i)}}{n - i_1 - i_2}$$

where $X_{(1)} \leq X_{(2)}, \leq \cdots \leq X_{(n)}$, i_1={number of X_i: $(X_i - M)/\hat{\zeta} < -k$}, and i_2 = {number of X_i: $(X_i - M)/\hat{\zeta} > k$}. Note that $\psi'(x)$ is the first derivative of ψ, which is 1 if $|x| \leq k$, and zero otherwise.

The estimated variance $\hat{\eta}$ of $\hat{\mu}_m$ is, from 2.21 in Section 2.4,

$$\hat{\eta} = \frac{\sum \hat{\zeta}\psi^2(u_i)}{\{\sum \psi'(u_i)\}^2}$$

where $u_i = (x_i - M)/\hat{\zeta}$. Let the one-step M-estimate of population means be $\hat{\mu}_{1m}$ and $\hat{\mu}_{2m}$. Then Wilcox's (1992) H test statistic is given by

$$H = \frac{\hat{\mu}_{1m} - \hat{\mu}_{2m}}{(\hat{\eta}_1 - \hat{\eta}_2)^{1/2}}$$

where $\hat{\eta}_1$ and $\hat{\eta}_2$ are corresponding standard errors of the one-step M-estimates.

To compute a bootstrap confidence interval, let $X_{i1}, X_{i2}, \ldots, X_{in_i}$, $i = 1, 2$ be the two random samples. For each i, draw a random bootstrap sample, i.e., for fixed i, select n_i observations with replacement. Here the probability of each observation being selected is $1/n_i$ within the ith group. Next, compute the one-step M–estimate based on the resampled observation, say $\hat{\mu}^*_{im1}$ and corresponding standard errors $\hat{\eta}^*_{i1}$, $i = 1, 2$. Then calculate H, say H^*_1, using $\hat{\mu}^*_{im1}$ and $\hat{\eta}^*_{i1}$, $i = 1, 2$. Repeat this procedure B times yielding $\hat{\mu}^*_{imb}$ and $\hat{\eta}^*_{ib}$, and H^*_b, $b = 1, 2, \ldots, B$. Let $H^*_{(1)} \leq H^*_{(2)} \leq \cdots \leq H^*_{(B)}$ be the H^*_b values in ascending order, and $m^* = [(1 - \alpha)]B$, where $[t]$ is the greatest integer less than or equal to t. Then $\hat{H}_{1-\alpha} = H^*_{(m^*)}$ is an estimate of the $(1 - \alpha)$th quantile of the distribution of $|\hat{\mu}^*_{1m} - \hat{\mu}^*_{2m} - \mu_1 + \mu_2|/(\hat{\eta}_1 + \hat{\eta})^{1/2}$, even when the distribution is asymmetric. Thus, reject H_0 if $|H| > \hat{H}_{1-\alpha}$. The choice of B, following Hall's (1986) suggestion, is to choose B so that $(B+1)^{-1}$ is a multiple of $1-\alpha$. Wilcox choose B=399 for $\alpha = 0.5$. For some problems, however, the bootstrap-percentile method requires B to be 1,000 or larger (Efron, 1987).

Wilcox (1992) compared this H test to Yuen's (1974) 10% trimmed mean test, Welch's test and his own median test based on the Harrell and Davis (1982) estimator. He summarized that the H test is better in terms of both Type I errors and power.

The problem of using Wilcox's approach is that this method can cause "divided by zero" errors within the bootstrap algorithm when sample sizes are less than 21. In addition, the Wilcox procedure requires much longer simulation execution time than other procedures, including our proposed procedure.

Tiku (1980, 1982) and Tiku and Singh (1981) proposed tests based on Tiku's *MMLE*s for equality of two population means. These tests will be introduced in the following sections together with the proposed test of ours.

5.3 Tests When Population Variances Are Equal

In this section we will propose a test based on *WLE*s with both Hellinger distance and negative exponential disparities based weights for testing equality of two population means under the equal variance assumption. When population variances are equal and the underlying distribution is normal the Student's t test is the uniformly most powerful test. The Student's t test statistic is of the form

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{s_p^2\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \tag{5.12}$$

where

$$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_2 + n_2 - 2}$$

and has a t distribution with $n_1 + n_2 - 2$ degrees of freedom. Here, $\bar{x}_i$ and s_i^2, $i = 1, 2$ are the sample means and sample variances respectively.

Tiku (1980) proposed a test statistic based on the *MML*10s which are the *MMLE*s based on 10% censored samples discussed in Section 2.7. Let $\tilde{\mu}_i$ and $\tilde{\sigma}_i^2$, $i = 1, 2$ be *MMLE*s (i.e., *MML*10s) of population means and variances from $100r_i\%$ censored

data, where $r_i = [0.5 + 0.1n_i]$. Tiku's test statistic is defined as

$$t_{MML} = \frac{\tilde{\mu}_1 - \tilde{\mu}_2}{\sqrt{\tilde{\sigma}^2\left(\frac{1}{m_1} + \frac{1}{m_2}\right)}} \tag{5.13}$$

where

$$\tilde{\sigma}^2 = \frac{(A_1 - 1)\tilde{\sigma}_1^2 + (A_2 - 1)\tilde{\sigma}_2^2}{A_1 + A_2 - 2}$$

$$A_i = n_i - 2r_i, \qquad m_i = n_i - 2r_i\beta_i, \qquad i = 1, 2.$$

Here the A_i, $i = 1, 2$, are calculated from (2.41) and the values of the coefficients α and β are taken from Table B.1 in Tiku, Tan and Balakrishnan (1986, pp.74–75). Tiku's test has an asymptotic distribution of $t(A_1 + A_2 - 2)$. Tiku (1980) conducted a simulation which compares his test to some nonparametric tests such as the Wilcoxon test. The simulation results show that Tiku's test is in general more powerful than the nonparametric tests he considered, except for some very short tailed distributions (uniform, for example) and some extremely skewed distributions (log-normal, for example).

Basu, Sarkar and Basu (1997) proposed a test for equality of two population means under the normal model using weighted likelihood estimators with the Hellinger distance based weights. They compared the empirical powers and empirical levels to those of Tiku's through a Monte Carlo study for various contaminated models. To establish the test statistic based on *WLE*s, Basu *et al.* (1997) used the extension of the minimum disparity estimation equation to the two samples case in a straightforward manner. Let

$$\rho(f_i^*, m_{\theta_i}^*) = \int A(\delta^*)\nabla m_{\theta_i}^* dx$$

be the minimum disparity measure for the ith sample. Then the overall measure of two disparities can be defined by

$$\rho_o = \frac{n_1\rho(f_1^*, m_{\theta_1}^*) + n_2\rho(f_2^*, m_{\theta_2}^*)}{n_1 + n_2}. \tag{5.14}$$

This form of the disparity for the Hellinger distance was considered by Simpson (1989), shown in (2.47). Sarkar and Basu (1995) also used this form of the overall disparity to construct a minimum disparity test for multiple discrete populations.

Let

$$w(\delta^*(x)) = \frac{A(\delta^*(x)) + 1}{\delta^*(x) + 1} \qquad \text{and} \qquad \delta^*(x) = \frac{f^*(x) - m_\theta^*(x)}{m_\theta^*(x)}$$

be the weight function and the Pearson residual, respectively. To find the *WLE*s we need to solve the weighted likelihood estimating equation

$$\int w(\delta^*(x))\nabla_\theta\{\log m_\theta(x)\}dF_n(x) = \frac{1}{n}\sum w(\delta^*(X_j))u(X_j,\theta) = 0 \tag{5.15}$$

where ∇_θ is the (partial) differential operator with respect to the parameter θ, $u(x,\theta)$ is the likelihood score function and F_n is the empirical distribution function. For the two samples case with parameter $\theta_i' = (\mu_i, \sigma^2)$ we have

$$\mu_i: \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij},\theta_i) = 0, \quad i = 1,2,$$

$$\sigma^2: \quad \sum_{i=1}^{2}\int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \sum_{i=1}^{2}\frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij},\theta_i) = 0.$$

These three estimating equations can be simplified as

$$\begin{aligned} \mu_i: &\quad \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))(X_{ij}-\mu_i)=0, \quad i=1,2, \\ \sigma^2: &\quad \sum_{i=1}^{2}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))\{(X_{ij}-\mu_i)^2-\sigma^2\}=0. \end{aligned} \tag{5.16}$$

The above equations in (5.16) are solved iteratively by creating new weights $w_i(\cdot)$ at each stage. The solutions of the equation (5.16) are of the form

$$\begin{aligned} \hat{\mu}_i &= \frac{\sum_{j=1}^{n_i} w_i\left(\delta_i^*(X_{ij})\right)X_{ij}}{\sum_{j=1}^{n_i} w_i\left(\delta_i^*(X_{ij})\right)}, \quad i=1,2, \\ \hat{\sigma}^2 &= \frac{\sum_{i=1}^{2}\sum_{j=1}^{n_i} w_i\left(\delta_i^*(X_{ij})\right)\left(X_{ij}-\hat{\mu}_i\right)^2}{\sum_{i=1}^{2}\sum_{j=1}^{n_i} w_i\left(\delta_i^*(X_{ij})\right)}. \end{aligned} \tag{5.17}$$

The simulation results of Basu *et al.* (1997) reveal that the test based on *WLE*s with *HD* weights generates higher (empirical) powers than Tiku's for all the sample sizes considered. However, the empirical levels in small samples are a little higher than the nominal levels. With sample sizes as large as 75, the empirical levels of Basu-Sarkar-Basu's test are very close to the nominal level.

In this dissertation we will extend Basu *et al.*'s (1997) work by including the *WLE*s with the negative exponential disparity based weight and will investigate the behavior of these tests not only under the contaminated distributions but also under various shapes of distributions like short-tailed, long-tailed and skewed.

The *RAF*s of the *HD* and the *NED* family are

$$A_{HD}(\delta^*(x)) = 2[\sqrt{(\delta^*+1)}-1]$$

and

$$A_{NED(a)}\big(\delta^*(x)\big) = \frac{(a+1) - \big[(a+1) + a\delta^*(x)\big]e^{-a\delta^*}}{a^2}.$$

Let $\left(\hat{\mu}_{HD1}, \hat{\mu}_{HD2}, \hat{\sigma}^2_{HD}\right)$ and $\left(\hat{\mu}_{NED(a)1}, \hat{\mu}_{NED(a)2}, \hat{\sigma}^2_{NED(a)}\right)$ be the *WLE*s based on the Hellinger distance weights and the negative exponential disparity weights with the values of a, respectively. Then the test statistics based on the *WLE*s have exactly the same form as the Student's t test in (5.12), replacing $\bar{x}_1, \bar{x}_2$ and s_p^2 with corresponding *WLE*s respectively.

5.3.1 Simulation and Results

Simulations were conducted to compare the performance of the Student's t test, Tiku's robust test, and Basu-Sarkar-Basu's *HD* test. The test based on $NED(a)$ are also included in the simulations. The values of a for $NED(a)$ are 0.15, 0.30, 0.50, 1.00, 2.00. The alternative hypothesis is

$$\mathrm{H}_1 : d = \mu_1 - \mu_2 > 0$$

as in Tiku's simulation (1980). We will follow Tiku's simulation setup for the verification of our simulation results. Various models and sample sizes are selected. Models considered in this simulation are very similar to those of Tiku's. They are (a) $N(0,1)$, (b) $0.9N(0,1) + 0.1t(1)$, (c) $0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$, (d) $t(2)$, (e) *Cauchy* with median 0 and the first quartile 1, (f) $U(0,1)$, (g) $\chi^2(4)$, (h) $Exp(1)$, (i) $\chi^2(1)$, and (j) $Lognormal(0,1)$. The sample sizes used are (n_1, n_2) = (8, 8), (20, 20), (40, 40), (50, 50), (75, 75), (100, 100).

The simulations are coded with Borland® C^{++} (1994), IMSL® C/Stat/Library (1995a) and IMSL® C/Stat/Library (1995b) and performed on Pentium 133Hz PC.

The program code is attached in Appendix I. Two sets of five thousand different samples are generated for each distribution using IMSL® routines with sizes n_i, i=1, 2 and the amount d is added to the first samples to produce mean differences.

The initial values of μ_1, μ_2 and σ^2 for estimating *WLE*s for *HD* and $NED(a)$ are defined as $\hat{\mu}_i^{(0)} = \text{median}\{X_{i1}, X_{i2}, \ldots, X_{in_i}\}$, i =1,2, $\sigma^{(0)}$ = 1.48·MAD where MAD= median$\{|X_{11}-\hat{\mu}_1^{(0)}|, \ldots, |X_{1n_1}-\hat{\mu}_1^{(0)}|, |X_{21}-\hat{\mu}_2^{(0)}|, \ldots, |X_{2n_2}-\hat{\mu}_2^{(0)}|\}$. The kernel function used is $N(0, h^2)$ with $\hat{h} = k \cdot \text{MAD}_i$ where k=0.5 and MAD$_i$=median$\{|X_{i1} - \hat{\mu}_i^{(0)}|, \ldots, |X_{in_i} - \hat{\mu}_i^{(0)}|\}$. The results of the simulations are given in Tables A.1, A.2, A.3, A.4, A.5 and A.6 in Appendix A. In the tables, the values of d represent the differences between two means so that the numbers are empirical levels when $d = 0$, and powers when $d \neq 0$. We consider the cases when d=0.0, 0.5, 1.0, 1.5, 2.0, 2.5, and 3.0, for all models except $U(0, 1)$ where d=0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6.

The results of the Monte Carlo study confirms those of Basu *et al.* (1997) for *HD* when sample sizes are (n_1, n_2)=(20, 20), (40, 40), (50,50), and (75, 75) under the models they considered in their simulations i.e., $N(0, 1)$, $0.9N(0, 1) + 0.1t(1)$ and $0.9N(0, 1)+0.1[N(0, 1)/U(0, 1)]$. For sample sizes (n_1, n_2)=(8,8), *HD* based empirical levels are relatively higher than the nominal levels. This is expected from the results of Basu *et al.* (1997). The remedy for small sample sizes will be discussed later in this section. The overall performance of the *HD* is better than Tiku's in terms of power under all distributions considered, especially, if sample sizes are large.

For the contaminated models and Cauchy distribution, Student's t test loses its control over Type I error rates and powers. The simulation also shows that the Student's t test is Type I robust in others cases, i.e., it controls only the Type I error rates but not powers. Student's t test preforms better in the short-tailed case, $U(0, 1)$, than Tiku's. Tiku *et al.* (1986) noted the performance of Tiku's test can be unsatisfactory with a short-tailed distribution like uniform or a "disaster" distribution

like Cauchy and can be improved by using *MMLE*s from no censored or 30% cesored data.

The test based on the ordinary negative exponential disparity estimators with $a = 1.0$ ($NED(1.00)$) shows higher empirical levels and powers than those of the *HD*. Basu and Sarkar (1994c) noted this phenomenon in conjunction with the trade-off between robustness and efficiency. They reported that, at the model, the *MNEDE* appears to be slightly less efficient and slightly more robust than the *MHDE*. They considered this as a suggestion of possible limitations of the curvature parameter in controlling the trade-off between robustness and efficiency. With large samples ($n \geq 40$), however, the $NED(1.00)$ performs as well as the *HD* at least with the contaminated models and short-tailed model, $U(0, 1)$. The performance of the $NED(1.00)$ for long tailed distributions and skewed distribution seems to fail to control the Type I error rates even with large sample sizes. The $NED(a)$ generally produces relatively higher Type I error rates and powers than those of the *HD*. For the performance of $NED(a)$, we see that the empirical levels of the tests decrease as a decreases at the cost of losing powers. One advantage of the $NED(a)$, however, is that one can choose the value of a to achieve control over level or power. The $NED(0.15)$ controls the Type I error rates better than *HD* with normal and contaminated models where sample sizes are (8, 8).

From Lindsay (1994), the estimators with $A'(\delta) < 0$ are supposed to be robust. The $A'_a(\delta)$ of the $NED(a)$ is positive if $a < 1$ and negative if $a > 1$, hence we are expecting that the larger the positive value of a, the more robust the test is. In this simulation, however, it didn't turn out that way, at least with the *NED* family, in the sense that the test fails to control the type I error rate. The means, variances and mean squared errors of the $NED(a)$ estimator of population means were computed. Also computed were the mean of the $NED(a)$ estimates of population variance. The

the results are in the Tables C.1, C.2 and C.3 in Appendix C, and Table D.1 in Appendix D. Under the model $N(0,1)$, (Table C.1), the best estimate happens near $a = 0.15$ in terms of MSE, and under the model $0.9N(0,1) + 0.1t(1)$, (Table C.2), it happens near $a = 2.50$ instead. Under the model $0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$, (Table C.3), the best estimate occurs near $a = 1.00$. We also note that the actual values of the calculated means, variances and MSEs are relatively close to each other. If we look at the averages of the estimated population variances of the $NED(a)$ in Table D.1, the closest value to the population variance value 1 happens at a near 0.15 for $N(0,1)$, at $a = 2.5$ for $0.9N(0,1) + 0.1t(1)$, and at a near 1 for $0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$. These results agree with the MSE of the estimates of population means. The largest estimated variances happen always at $a = 0.15$, the smallest at $a = 2.5$. As a consequence, the larger estimates of variance contribute to smaller calculated test statistic values which result in fewer rejections of the null hypotheses making the empirical level lower. Note also that as the sample size increases the performance of the $NED(a)$ is better with a near 1. It is not clear at this point what causes $NED(a)$ behave strangely. We need more research on tests based on the *WLE*s with the minimum negative exponential disparity weights when $a \neq 1$.

To improve the performance of controlling Type I error rates of the tests based on *WLE*s, especially for small sample cases, we use the square root of the $W_i(\cdot)$ in the estimating equations and k=0.9 if $n_i < 30$, i=1,2 in the smoothing parameter $h = k\sigma$. The empirical levels and powers of the *HD* are computed with various values of k $(0.1 < k < 0.9)$ under the model $N(0,1)$. Given in the Table 5.1, Figure 5.1 and Figure 5.2 are corresponding graphs of the level and power at the nominal level of α=0.5. From Figure 5.1, we can identify that the Type I error rates decrease as k increases. Particularly, the *HD* produces Type I error rates very close to the nominal levels for sample sizes of (10, 40) and (30, 30) if k is near 0.9.

Table 5.1: Empirical levels ($\mu = 0$) and powers ($\mu = 1$) for the MHDE with various k, where $\hat{h} = k\hat{\sigma}$ and (n_1, n_2)=(10,10), (10, 40), (30, 30). Model is $N(\mu, 1)$.

		Empirical Level			Empirical Power		
	k	(10,10)	(10, 40)	(30, 30)	(10,10)	(10, 40)	(30, 30)
$\alpha = 0.05$	0.1	0.143	0.080	0.083	0.786	0.884	0.987
	0.2	0.111	0.069	0.066	0.766	0.891	0.985
	0.3	0.096	0.065	0.068	0.754	0.889	0.986
	0.4	0.088	0.065	0.065	0.750	0.885	0.987
	0.5	0.079	0.063	0.062	0.742	0.885	0.987
	0.6	0.075	0.061	0.060	0.737	0.885	0.987
	0.7	0.071	0.058	0.055	0.734	0.877	0.985
	0.8	0.068	0.056	0.054	0.731	0.876	0.985
	0.9	0.066	0.055	0.053	0.728	0.876	0.984
$\alpha = 0.01$	0.1	0.063	0.027	0.024	0.588	0.736	0.952
	0.2	0.043	0.019	0.016	0.546	0.726	0.941
	0.3	0.032	0.016	0.015	0.522	0.716	0.936
	0.4	0.027	0.017	0.013	0.502	0.698	0.935
	0.5	0.022	0.016	0.012	0.489	0.694	0.933
	0.6	0.019	0.015	0.011	0.478	0.688	0.936
	0.7	0.018	0.016	0.010	0.470	0.688	0.933
	0.8	0.017	0.016	0.010	0.463	0.686	0.933
	0.9	0.017	0.015	0.010	0.458	0.686	0.932

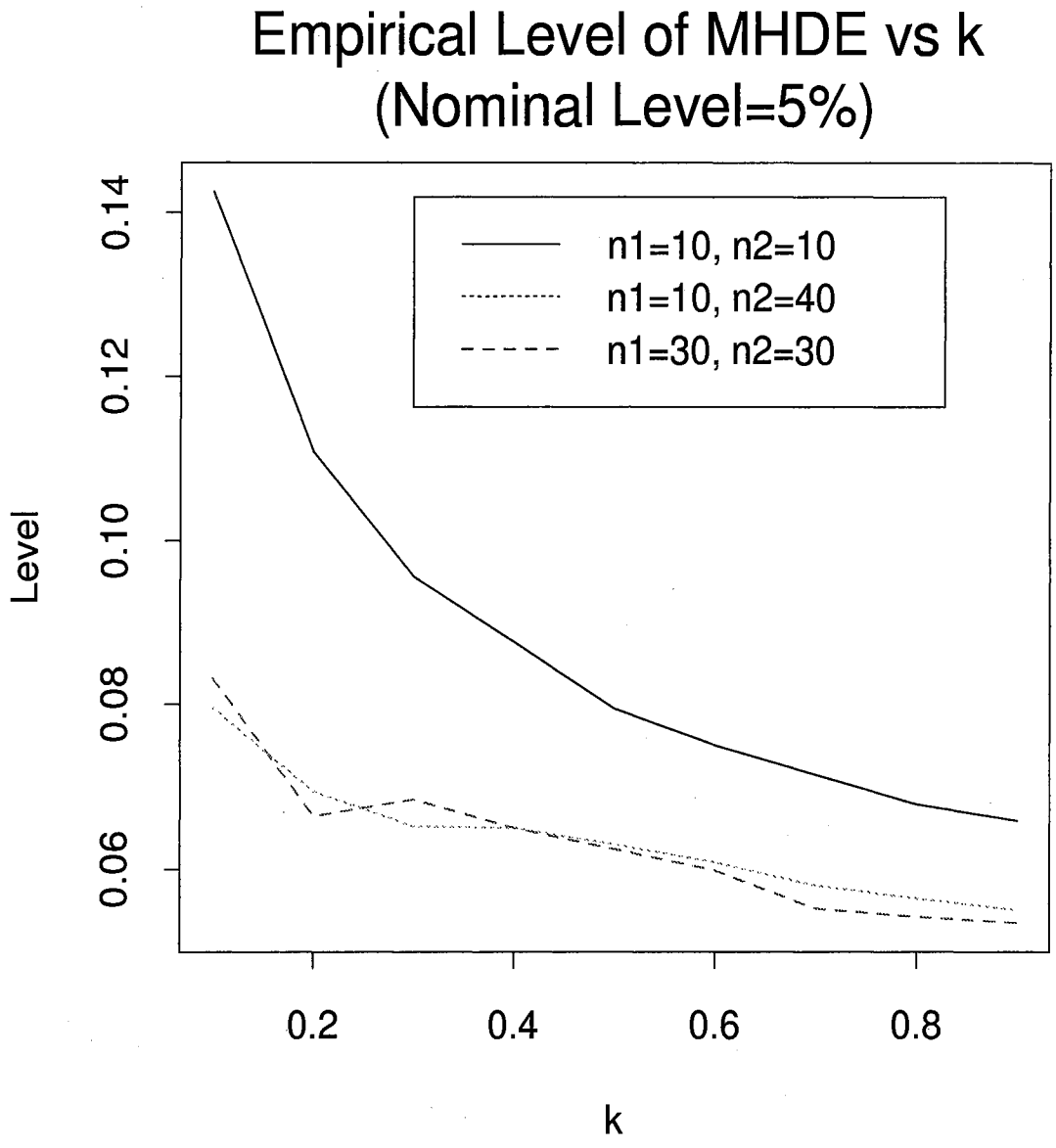

Figure 5.1: Empirical levels of the *MHDE* by various values of k where $\hat{h} = k\hat{\sigma}$. Sample size: (n_1, n_2)=(10,10), (10,40), (30,30). Model is $N(\mu = 0, 1)$.

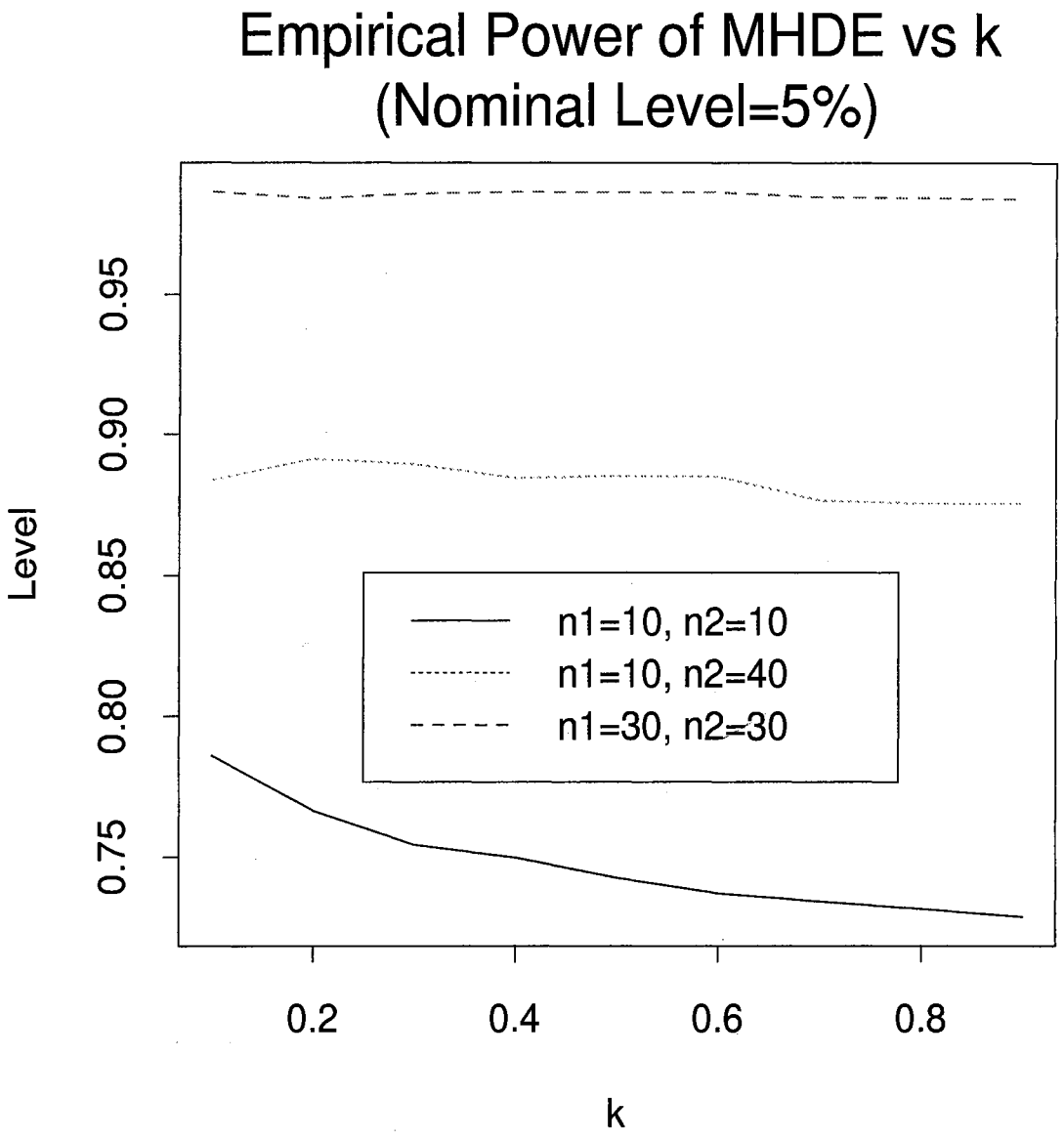

Figure 5.2: Empirical powers ($\mu = 1$) of the *MHDE* by various values of k where $\hat{h} = k\hat{\sigma}$. Sample size: (n_1, n_2)=(10,10), (10,40), (30,30). Model is $N(\mu = 0, 1)$.

Further, when sample sizes are equal, the curves resemble a strictly decreasing curve. Figure 5.2 shows similar patterns in the power.

As a result of this computation and adapting Basu *et al.*'s (1995) suggestion, we performed additional simulations for small sample size cases and unequal sample size cases. These simulations were conducted with selected models of $N(0,1)$, $t(2)$, $0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$, and $0.9N(0,1) + 0.1t(1)$. The sample sizes were (n_1, n_2)=(8, 8), (20, 20), (8, 20), (40, 40) and (40, 10). The results of the simulations with d =0.0, 1.0, 2.5, are given in Tables A.7, A.8, A.10, A.9 and A.11 in Appendix A. By comparing Tables A.7, A.8, and A.9 to Tables A.1, A.2, A.3, we see drastic improvement in the empirical levels or Type I error rates for the *HD* and the $NED(a)$ but relatively less loss in the powers. In all cases, the empirical levels of the *WLE*s are almost the same as the nominal levels. Further, both the empirical levels and powers are better than Tiku's in the sense of producing the empirical levels closer to the nominal levels and the powers higher than those of Tiku's.

The reason for what is happening in the later simulations is that (i) a smoother density estimator is being used than before by putting k=0.9 if $n_i < 30$ i=1,2, and (ii) the weights are made closer to 1 (because $0 \leq w \leq 1$) than before by taking square roots of the original weights. This approach seems to prevent the *WLE*s from downweighting observations too much when $A(\delta^*) < \delta^*$. We obtained similar results for $(n_1, n_2) = (8, 20)$ and $(40, 10)$ except when $(n_1, n_2) = (8, 20)$ and the model is $t(2)$.

5.4 Tests When Population Variances Are Unequal

It is well known that for the Behrens-Fisher problem there is no uniformly most powerful test for all sample sizes, so it is of interest to consider the performance of asymptotically optimal tests. Welch's v test is one of the most popular tests for testing

equality of two population means when variances are unequal. Welch (1938) proposed three tests described in Section 5.2.1, and the most well known v test statistic, which is often denoted t_v, is

$$t_v = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{5.18}$$

where $\bar{x}_i$ and s_i^2, $i = 1, 2$, are the sample means and sample variances. Welch (1938, 1947) proposed several tests based on t_v where the critical values of the tests are varied. One of the procedures is the approximate degrees of freedom ($APDF$) method which specifies the sampling distribution of Welch's test statistic t_v. The APDF critical value of t_v is a percentile of the Student's t distribution with f degrees of freedom, where

$$\frac{1}{f} = \frac{c^2}{n_1 - 1} + \frac{(1-c)^2}{n_2 - 1}, \qquad c = \frac{\sigma_1^2/n_1}{\sigma_1^2/n_1 + \sigma_2^2/n_2}. \tag{5.19}$$

In practice, we use the estimate of f from the data.

Tiku and Singh (1981) proposed a Welch-type statistic, t^*_{MML}, based on Tiku's (1967, 1980) modified maximum likelihood estimators, and showed that this statistic is robust to symmetric and moderately skewed distributions. They investigated the power of the t^*_{MML} and concluded that t^*_{MML} is more powerful than Yuen's (1974) test based on trimmed means and matched variances. Tiku and Singh, however, used a variance formula which differs from what Yuen (1974) used in her test. Yuen originally used winsorized means as in (5.7) in calculation of the variances but Tiku and Singh used trimmed means as defined in (5.6).

Let $\tilde{\mu}_i$ and $\tilde{\sigma}_i^2$, $i = 1, 2$, be the $MML10$s of population means and variances from the $100r_i\%$ censored data, where $r_i = [0.5 + 0.1n_i]$. Tiku and Singh's (1981) test

statistic is defined as

$$t^*_{MML} = \frac{\tilde{\mu}_1 - \tilde{\mu}_2}{\sqrt{\left(\frac{\tilde{\sigma}_1^2}{m_1} + \frac{\tilde{\sigma}_2^2}{m_2}\right)}} \tag{5.20}$$

and has an approximate t distribution with h degrees of freedom where h is determined by applying the $APDF$ in (5.19), i.e.,

$$\frac{1}{h} = \frac{d^2}{A_1 - 1} + \frac{(1-d)^2}{A_2 - 1}, \qquad d = \frac{\tilde{\sigma}_1^2/m_1}{\tilde{\sigma}_1^2/m_1 + \tilde{\sigma}_2^2/m_2}.$$

Here $A_i = n_i - 2r_i$, and $m_i = n_i - 2r_i + 2r_i\beta_i$, i=1,2. The values of β_i are given in Table B.1 in Tiku, Tan and Balakrishnan (1986, pp.74–75).

Let

$$w(\delta^*(x)) = \frac{A(\delta^*(x)) + 1}{\delta^*(x) + 1} \qquad \text{and} \qquad \delta^*(x) = \frac{f^*(x) - m^*_\theta(x)}{m^*_\theta(x)}$$

be the weight function and the Pearson residual respectively. To find the WLEs we need to solve the weighted likelihood estimating equation

$$\int w(\delta^*(x))\nabla_\theta\{\log m_\theta(x)\}dF_n(x) = \frac{1}{n}\sum w(\delta^*(X_j))u(X_j,\theta) = 0$$

where ∇_θ is the (partial) differential operator with respect to the parameter θ, $u(x,\theta)$ is the likelihood score function and F_n is the empirical distribution function. For the

two samples case with parameter $\theta_i' = (\mu_i, \sigma_i^2)$ we have

$$\mu_i : \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij},\theta_i) = 0, \quad i = 1,2,$$

$$\sigma_i^2 : \quad \int w_i(\delta_i^*(X_{ij}))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(x))u(X_{ij},\theta_i) = 0, \quad i = 1,2.$$

These four estimating equations can be simplified as

$$\begin{aligned} \mu_i : \quad & \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))(X_{ij} - \mu_i) = 0, \quad i = 1,2, \\ \sigma_i^2 : \quad & \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))\{(X_{ij} - \mu_i)^2 - \sigma_i^2\} = 0, \quad i = 1,2. \end{aligned} \tag{5.21}$$

The above equations in (5.21) are solved iteratively by creating new weights $w_i(\cdot)$ at each stage. The solutions of the equations (5.21) are of the form

$$\begin{aligned} \hat{\mu}_i &= \frac{\sum_{j=1}^{n_i} w_i^{1/2}\left(\delta_i^*(X_{ij})\right)X_{ij}}{\sum_{j=1}^{n_i} w_i^{1/2}\left(\delta_i^*(X_{ij})\right)}, \quad i = 1,2, \\ \hat{\sigma}_i^2 &= \frac{\sum_{j=1}^{n_i} w_i^{1/2}\left(\delta_i^*(X_{ij})\right)\left(X_{ij} - \hat{\mu}_i\right)^2}{\sum_{j=1}^{n_i} w_i^{1/2}\left(\delta_i^*(X_{ij})\right)}, \quad i = 1,2. \end{aligned} \tag{5.22}$$

Note we use the square roots of the weights instead of the original weights. Since all of the *WLE*s have the same distribution as the *MLE** under the model and the *MLE** is the same the *MLE* under the normal model, we can proceed exactly along the same lines as Welch (1938) to develop a Welch-type robust test based on the *WLE*s. Letting $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ denote the *WLE*s with the Hellinger distance weights. The

proposed statistic is then defined as

$$t^*_{HD} = \frac{\hat{\mu}_1 - \hat{\mu}_2}{\sqrt{\frac{\hat{\sigma}_1^2}{n_1} + \frac{\hat{\sigma}_2^2}{n_2}}}.$$

The asymptotic distribution of the test t^*_{HD} is a t distribution with g degrees of freedom determined by

$$\frac{1}{g} = \frac{b^2}{n_1 - 1} + \frac{(1-b)^2}{n_2 - 1}, \qquad b = \frac{\hat{\sigma}_1^2/n_1}{\hat{\sigma}_1^2/n_1 + \hat{\sigma}_2^2/n_2}.$$

A test statistics based on the ordinary *NED* can be constructed by the same analogy. The square root weights, $0 \leq W^{1/2}(\delta^*) \leq 1$ are used in the computation of *WLE*s for both *NED* and *HD* weights.

5.4.1 Simulation and Results

To verify the robustness of the above four tests, Welch (1938), Tiku and Singh (1981), t^*_{HD}, and t^*_{NED}, we simulate 5,000 Monte Carlo runs using IMSL® C/Stat/Library random number generators in Borland® C^{++} on a Pentium 133Hz PC. Five different types of distributions are considered in the simulations. They are two contaminated distributions, $0.9N(0,1)+0.1N(0,64)$ and $0.9N(0,1)+0.1[N(0,1)/U(0,1)]$; one long-tailed distribution $t(4)$; one short tailed distribution $U(-1,1)$; and the standard normal distribution. These distributions are properly scaled so that the population standard deviations are all equal to 1 and then multiplied by σ_i, where σ_i, $i = 1, 2$, are the actual values of population standard deviations we will consider. The values of the population standard deviations considered are (σ_1, σ_2) = (1,1), (1,2) and (1,4). For example, if y_1 is a random number from $t(5)$ for the first population, then we use the quantity $x_1 = y_1\sigma_1/\sqrt{2}$. The amount d = 0, 1.5 and 2.5 are added to the first

sample to produce the mean differences except for the uniform distribution. For the uniform distribution, $u(-1,1)$, $d = 0$, 0.5, and 1.0 are added instead to produce the mean differences not too large.

The initial values of μ_1, μ_2 and σ^2 for estimating the *WLE*s for the *HD* and $NED(a)$ are defined as $\hat{\mu}_i^{(0)} = \text{median}\{X_{i1}, X_{i2}, \ldots, X_{in_i}\}$, and $\sigma_i^{(0)} = 1.48 \cdot \text{MAD}_i$, i=1,2. The kernel function used is $N(0, h^2)$ with $\hat{h} = k \cdot \text{MAD}_i$ where k=0.9 if $n_i < 30$, k=0.5 otherwise, and MAD_i=median$\{|X_{i1} - \hat{\mu}_i^{(0)}|, \ldots, |X_{in_i} - \hat{\mu}_i^{(0)}|\}$, $i = 1, 2$.

Selected sample sizes are (n_1, n_2) = (10,10), (10,20), (20,10), (30,30), (10,40), (40,10) and (70,70). The results of the simulations are presented in Tables B.1, B.2, B.3, B.4, B.5, B.6, and B.7 in Appendix B. The program code is attached in Appendix J.

Under the model $N(0,1)$, Welch's test controls Type I error rate better than the other three tests. With the contaminated models, however, Welch's test shows relatively lower empirical levels in most cases, which shows that Welch's test is not even Type I robust.

Under the contaminated model, tests based on the *WLE*s produce higher powers than Tiku's. When larger sized samples come from more variable populations (positive condition; see Coomb *et al.*, 1996), the *HD* test seems to perform better on both Type I error rate and power than Tiku's. On the contrary, the *NED* test shows slightly higher empirical levels than the *HD* test under the same conditions.

For the long-tailed symmetric distribution $t(4)$, all tests show about the same performance except for the Welch test. Welch's test shows lower powers than the others. For the short-tailed distribution $U(-1,1)$, Tiku's test produces relatively lower power in most of the cases. Tests based on WLEs produce higher powers than other tests under the short-tailed model. Moreover, Welch's test seems to perform well also under the short-tailed distribution.

5.5 Concluding Remarks

It has been shown, empirically, that the tests based on the *WLE*s seem to be excellent robust alternatives to the existing tests for equality of two population means, whether the population variances are equal or not. These tests are quite robust with respect to departure from normality. The remedy for the performance of the *WLE*s with the small samples is resolved by choosing square root weights and slightly wider bandwidths. The computation of the *WLE*s are not that complicated, even though we need to solve the estimating equations iteratively, especially in the age of high performance computers. From a practical view point, tests based on *WLE*s are at least as good as Tiku's tests under contamination and about the same as the Student's t test or Welch's test under the normal model.

CHAPTER VI

TESTS FOR EQUALITY OF VARIANCES

6.1 Introduction

To test $H_0 : \sigma_1^2 = \sigma_2^2$ or equivalently $\sigma_1^2/\sigma_2^2 = 1$ under the assumption of normality, the uniformly most powerful test is based on the statistic F, which is the ratio of the two sample variances, i.e.,

$$F = \frac{s_1^2}{s_2^2}.$$

Bartlett (1937, 1939) generalized this likelihood ratio test to the case of k populations. Unfortunately, this F test (or Bartlett test) is not at all robust even asymptotically; this test can have actual Type I error rates several times larger than the nominal level, which inhibits its usefulness (Box, 1953; Gartside, 1972; Tan, 1982; Conover, 1980). Several robust tests have been proposed by Levene (1960), Layard (1973), Miller (1968), Flinger and Killeen (1976) and studied by Gartside (1972), Brown and Forsythe (1974a,b), Conover, Johnson and Johnson (1981), and Tiku and Balakrishnan (1984).

Conover *et al.* (1981) conducted a study of homogeneity of variance. Fifty-six tests for equality of variances were compared. They recommended three tests in terms of robustness and power, which are Levene's test, Fligner and Killeen's χ^2 and Fligner and Killeen's F using the sample median instead of the sample mean.

Milliken and Johnson (1984, p. 22) also made the following recommendations based on the results of Conover *et al.* from a practical point of view.

- *If the analyst is confident that the data are nearly normal, then use*

Bartlett's test or Hartley's test. If the sample sizes are very unequal, use Bartlett's test; otherwise, Hartley's test should be accurate enough to determine whether the usual F tests and t tests for comparing population means are appropriate.

- *For very large data sets, use Box's test. Box's test is very robust, but not very powerful for small sample sizes.*
- *In all other instances, use Levene's test. Levene's test was shown to be nearly as good as Bartlett's and Hartley's test for normally distributed data and superior to them for nonnormally distributed data. If the data tend to be very skewed, Levene's test can be improved by replacing the sample means by the sample medians.*

We will introduce a robust test for equality of variances based on the weighted likelihood estimators using the Hellinger distance. This test seems to perform quite well against nonnormality, including skewed, contaminated and long-tailed distributions. A comparative study is done for testing two and three population variances through simulations and the results are presented.

6.2 Alternatives to Likelihood Ratio Tests

Let $x_{i1}, x_{i2}, \ldots, x_{in_i}$, $i = 1, 2, \ldots, k$, be random observations from k population distributions, with sample means $\bar{x}_i$ and sample variances s_i^2 respectively.

Bartlett's (1937) test under normality is

$$B = -\frac{1}{c}\sum_i (n_i - 1)\log\left(\frac{s_i^2}{s^2}\right)$$

where

$$s_i^2 = \sum_j \frac{(x_{ij} - \bar{x}_i)^2}{(n_i - 1)}$$

$$s^2 = \sum_i \frac{(n_i - 1)s_i^2}{\sum_i (n_i - 1)}$$

and

$$c = 1 + \frac{1}{3(k-1)}\left[\sum_i \left(\frac{1}{n_i - 1}\right) - \frac{1}{\sum_i (n_i - 1)}\right].$$

Bartlett showed that under $H_0 : \sigma_1^2 = \sigma_2^2 = \cdots = \sigma_k^2$, B is distributed approximately as a chi-squared distribution with $k - 1$ degrees of freedom, and the approximation is good for quite small samples. Box (1953), however, showed if the underlying distribution is arbitrary, then

$$B \xrightarrow{d} \left[1 + \frac{\gamma}{2}\right]\chi^2_{k-1}, \qquad \gamma = \frac{E[X - \mu]^4}{\sigma^4} - 3.$$

where γ is the kurtosis of the underlying distribution.

Box(1953) developed a test known as a *log-ANOVA* test. Suppose $n_i = mJ_i$, the ith sample is divided randomly into J_i subsamples, each containing m observations. Let $y_{ij} = \log s_{ij}^2$, where s_{ij}^2 is the sample variance of the jth subsample of the ith sample. Then the y_{ij} are approximately distributed as $N[\log \sigma_i^2, 2/(m-1) + (\gamma/m)]$, and the Box procedure is used to perform a one-way analysis of variance on the y_{ij} and to test H_0 by the standard F test. Under H_0, the test statistic

$$G = \frac{\sum_i J_i(\hat{y}_{i.} - \bar{y}_{..})/(k-1)}{\sum_i \sum_j (y_{ij} - \hat{y}_{i.})^2 / \sum_i (J_i - 1)}$$

is asymptotically distributed as $\chi^2_{k-1}/(k-1)$.

Levene (1960) suggested the following one-way *ANOVA* type statistic

$$W_0 = \frac{\sum_i n_i(\bar{z}_i - \bar{z})^2)/(g-1)}{\sum_i \sum_j (z_{ij} - \bar{z}_i)^2 / \sum_i (n_i - 1)}$$

where

$$z_{ij} = |x_{ij} - \bar{x}_i|, \tag{6.1}$$

$$\bar{z}_i = \sum Z_{ij}/n_i \quad \text{and} \quad \bar{z} = \sum\sum z_{ij} / \sum n_i \,. \tag{6.2}$$

The critical values of W_0 are obtained from the F table with $k-1$ and $\sum(n_i - 1)$ degrees of freedom. Alternative fomulations considered by Levene were to replace z_{ij} by $\sqrt{z_{ij}}$ or by $\log z_{ij}$.

Brown and Forsythe (1974a) modified Levene's test and suggested W_{50} by replacing the sample mean $\bar{x}_i$ by the sample median $\tilde{x}_i$ in forming z_{ij} and W_{10} by the replacing the sample means by the 10% trimmed mean of the ith sample. The test statistic W_{50} is the test recommended by Conover *et al.* (1981) and Milliken and Johnson (1982) as Levene's test using the median. Hence the test W_{50} is defined by

$$W_{50} = \frac{\sum_i n_i(\bar{z}_i - \bar{z})^2)/(g-1)}{\sum_i \sum_j (z_{ij} - \bar{z}_i)^2 / \sum_i (n_i - 1)} \tag{6.3}$$

with $z_{ij} = |x_{ij} - \tilde{x}_i|$.

Layard (1973) suggested a chi-squared test statistic which is a function of the kurtosis. The kurtosis is estimated by pooling the numerators and denominators of

the individual estimates of kurtosis for each group. Using his notation, let

$$S' = \frac{1}{\hat{\tau}^2}\sum(n_i - 1)\left[\log s_i^2 - \frac{\sum(n_i - 1)\log s_i^2}{\sum(n_i - 1)}\right]^2$$

where

$$\hat{\tau}^2 = 2 + \left\{1 - \frac{k}{\sum n_i}\right\}\hat{\gamma},$$

and

$$\hat{\gamma} = \frac{(\sum n_i)\sum\sum(x_{ij} - \bar{x}_i)^4}{\{\sum\sum(x_{ij} - \bar{x}_i)^2\}^2} - 3$$

is the estimate of the kurtosis. Then S' is asymptotically distributed as χ^2 with $k-1$ degrees of freedom.

Miller (1968) pointed out that for very small samples the high correlation between deviations in the sample group destroy the robustness of the Levene's tests. To overcome this problem, he suggested use of the Layard test in a jackknife procedure and computed one-way *ANOVA* F. The null distribution of the Miller's jackknife statistic is exactly the same as that of Levene's W_0 test. Brown and Forsythe (1974a) argued that empirical sampling with ten or more observations per group does not indicate the problem Miller reported. Flinger and Killeen's (1976) χ^2 and F tests use the ranks of $|x_{ij} - \bar{x}_i|$ and the normal scores based on the ranks. Later, Tiku and Balakrishnan (1984) proposed tests based on the *MMLE*s for $k = 2$, however, Welsh and Morrison (1990) noted that Tiku and Balakrishnan's proof of the asymptotic normality for the test is not correct, and the variance formula is inappropriate in general.

The Brown-Forsythe test, i.e., the modified Levene's test with median, as shown

in (6.3), is the one of the most recommended test among other competitors. We will consider this test to compare our proposed test. Brown and Forsythe (1974a) compared their test to the F ratio test, Miller's jackknife test, Layard's χ^2 test and the original Levene's test for two populations. According to the results of Brown and Forsythe, Levene type procedures appear to be more robust. When variances are equal, however, the Brown-Forsythe test is conservative for small samples sizes under the normal model. Along the lines of the work of Brown and Forsythe (1970a), we will modify Levene's test by replacing the sample mean with the Hellinger distance weighted *WLE*s. Let $\hat{\mu}_i$, $i = 1, 2, \ldots, k$ be weighted likelihood estimates of the k population means, where the weights of *WLE*s are calculated using the *HD*. Then the modified Levene's test with *WLE*s is defined by

$$W_{HD} = \frac{\sum_i n_i(\bar{z}_i - \bar{z})^2)/(k-1)}{\sum_i \sum_j (z_{ij} - \bar{z}_i)^2 / \sum_i (n_i - 1)} \tag{6.4}$$

where $z_{ij} = |x_{ij} - \hat{\mu}_i|$ and $\bar{z}_i$ and $\bar{z}$ are defined as in (6.2). The asymptotic critical values of W_{HD} are obtained from the F table with $k-1$ and $\sum_i (n_i - 1)$ degrees of freedom.

To get the *WLE*s for the proposed test in (6.4), let

$$w(\delta^*(x)) = \frac{A(\delta^*(x)) + 1}{\delta^*(x) + 1} \qquad \text{and} \qquad \delta^*(x) = \frac{f^*(x) - m_\theta^*(x)}{m_\theta^*(x)}$$

be the weight function and the Pearson residual, respectively. To find the *WLE*s we need to solve the weighted likelihood estimating equation

$$\int w(\delta^*(x)) \nabla_\theta \{\log m_\theta(x)\} dF_n(x) = \frac{1}{n} \sum w(\delta^*(X_j)) u(X_j, \theta) = 0$$

where ∇_θ is the (partial) differential operator with respect to the parameter θ, $u(x, \theta)$

is the likelihood score function and F_n is the empirical distribution function. For the k samples case with parameter $\theta_i' = (\mu_i, \sigma_i^2)$, $i = 1, 2, \ldots, k$, we have

$$\mu_i : \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij}, \theta_i) = 0, \quad i = 1, 2, \ldots, k,$$

$$\sigma_i^2 : \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij}, \theta_i) = 0, \quad i = 1, 2, \ldots, k.$$

These $2k$ estimating equations can be simplified as

$$\begin{aligned} \mu_i : &\quad \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))(X_{ij} - \mu_i) = 0, \quad i = 1, 2, \ldots, k, \\ \sigma_i^2 : &\quad \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))\{(X_{ij} - \mu_i)^2 - \sigma_i^2\} = 0, \quad i = 1, 2 \ldots, k. \end{aligned} \tag{6.5}$$

The above equations in (6.5) are solved iteratively by creating new weights $w_i(\cdot)$ at each stage. The solutions of the equations (6.5) are of the form

$$\begin{aligned} \hat{\mu}_i &= \frac{\sum_{j=1}^{n_i} w_i\big(\delta_i^*(X_{ij})\big)X_{ij}}{\sum_{j=1}^{n_i} w_i\big(\delta_i^*(X_{ij})\big)}, \quad i = 1, 2, \\ \hat{\sigma}_i^2 &= \frac{\sum_{j=1}^{n_i} w_i\big(\delta_i^*(X_{ij})\big)\big(X_{ij} - \hat{\mu}_i\big)^2}{\sum_{j=1}^{n_i} w_i\big(\delta_i^*(X_{ij})\big)}, \quad i = 1, 2. \end{aligned} \tag{6.6}$$

Note that we do not use the square roots of the weights in this case. For the test of equality of variances, it seems the original weights produces better results than the square root weights. Also the original bandwidth $h = 0.5 \cdot \text{MAD}_i$ is used. Even though the *WLE*s for the variances are not used in the test, we need them to solve the equations (6.5). The program codes are attached in Appendix K and Appendix L.

6.3 Simulation and Results

Three tests are compared through Monte Carlo studies: the F ratio test (or Bartlett's in the case of three populations), the Brown-Forsythe test and the modified Levene's test with the *MHDE*s.

Four different distributions are included in the simulations. They are: (1) mixed distribution: $0.9N(0,1)+0.1N(0,8)$, (2) $N(0,1)$, (3) long-tailed distribution: $t(4)$ and (4) skewed distribution: $\chi^2(4)$. These distributions are those distributions that Brown and Forsythe considered in their simulation, except for the mixed one. Samples sizes are (n_1, n_2) = (40,40), (10,10), (10,20) and (20,40) for two populations,and (n_1, n_2, n_3) = (40,40,40), (10,10,10), (10,10,40), (20,20,40) and (40,40,40) for three populations. For two populations, two sets of (n_1, n_2) random numbers are generated from the IMSL® C/Stat/Library routines and used in the calculation of the test statistics. The observations in the second sample are rescaled to reflect the ratio of population variances of the two groups; i.e., $(\sigma_1^2 : \sigma_2^2)$ is (1:1), (1:2), (1:4) and (1:8) for the equal samples sizes case, and (1:1), (1:2), (1:4), (1:8), (8:1), (4:1) and (2:1) for the unequal sample sizes case. For the three population case, $(\sigma_1^2, \sigma_2^2, \sigma_3^2)$ =(1:1:1), (1:1:2), (1:1:4), (1:2:4), (4:2:1), (4:1:1) and (2:1:1). These are repeated 5,000 times for each distribution and the number of rejections at both the nominal level $\alpha = 0.5$ and 0.1. are recorded.

The results of the simulations are given in Tables E.1 and E.2 in Appendix E for the two population cases; in Tables F.1 and F.2 in Appendix F for the three population cases. As we expected, the F test and Bartlett's test reject far too often when the underlying distributions are not normal. For small sample sizes, the Brown-Forsythe tests is conservative when the underlying distributions are normal as noted by Brown and Forsythe. The modified Levene's test with *MHDE*s has shown an excellent

performance for mixed and long-tailed distributions. Both levels and powers of the *HD* test appear better than Brown-Forsythe test except for the skewed distribution. Under the skewed distribution, the *HD* test has slightly higher empirical levels than the nominal level, but has highest power.

6.4 Concluding Remarks

Tests based on the *WLE*s were considered. Simulation results indicated that the equality of variances in long-tailed and mixed distributions can best be tested by a modified Levene's test with the *HD* weighted *WLE*s among its competitors. For the skewed distribution, the *HD* test is a robust alternative to the Brown-Forsythe in terms of power.

CHAPTER VII

TESTS OF EQUALITY OF MORE THAN TWO POPULATION MEANS

7.1 Introduction

The testing of the equality of more than two population means from independent samples is a common statistical problem. Unfortunately, the famous one-way analysis of variance (*ANOVA*) F test is sensitive to a lack of homogeneity of the within group variances. Sheffé (1959) noted the actual Type I error rate of a test may be different from the nominal level when the groups have different underlying population variances. There are numerous studies which show that the *ANOVA* F test is not robust to failure of the equal population variances assumption (Brown and Forsythe, 1974b, 1974c; Rogan and Keselman, 1977; Clinch and Keselman, 1982; Wilcox,1988, 1993, 1995a; Bradley, 1978; Coombs *et al.*, 1996). In the early stages, Box (1954a, 1954b) and Horsnell (1953) considered the F test to be robust to the equal variance assumption whenever samples sizes were equal. The study of Box (1954a), however, is limited to the situation where the ratio of the largest standard deviation to the smallest (say, ϑ) less than or equal to $\sqrt{3}$. Wilcox (1987) and Fenstad (1983) argued and demonstrated that value of ϑ as large as 4 is not extreme. In addition, numerous investigations discuss the effect of nonnormality on the power of the F test (Boneau, 1963; Srivastava, 1969; Tan 1982; Tiku, 1971; Wilcox 1987). Tan (1982) summarized that for moderate departures from normality but equal sample sizes, the F test is quite robust with respect to power. For severe departures, like exponential or log-normal distributions, the effect is considerable and increase as sample size differences increase. Tiku (1982) noted that kurtosis has a more dominant role than skewness in

determining power.

In the following sections we will introduce some alternatives to the *ANOVA F* test which are known to be robust in some sense. We will also propose a test based on the *WLE*s with the Hellinger distance based weights and compare it to the some of the alternatives.

7.2 Alternatives to *ANOVA F* Test

Let x_{ij} represent the jth observation in the ith group, where $j = 1, 2, \ldots, n_i$ and $i = 1, 2, \ldots, k$. Assume that the x_{ij} are independent normal variates with mean μ_i and variance σ_i^2. The *ANOVA F* test statistic is

$$F = \frac{\sum_i n_i(\bar{x}_{i.} - \bar{x}_{..})^2/(k-1)}{\sum_i (n_i - 1)s_i^2/(N-k)} \tag{7.1}$$

where

$$N = \sum_i n_i,$$

$$\bar{x}_{i.} = \sum_i \frac{x_{ij}}{n_i},$$

$$\bar{x}_{..} = \sum_i \sum_j \frac{x_{ij}}{N} = \sum_i \frac{n_i \bar{x}_{i.}}{N},$$

$$s_i^2 = \sum_i \frac{(x_{ij} - \bar{x}_{i.})^2}{(n_i - 1)}.$$

The *ANOVA F* follows the F distribution with $k-1$ and $N-k$ degrees of freedom if all population means and variances are equal.

Welch (1951) extended his approximate degrees of freedom procedure (Welch,

1947) to the k groups situation and proposed a test when at least one pair of population variances may not be equal. The statistic is

$$W = \frac{\sum_i w_i(\bar{x}_{i.} - \tilde{x}_{..})^2/(k-1)}{\left[1 + \frac{2(k-2)}{g^2-1}\sum_i \frac{1}{n_i - 1}\left(1 - \frac{w_i}{w}\right)^2\right]}, \tag{7.2}$$

where

$$w_i = \frac{n_i}{s_i^2},$$

$$w = \sum_i w_i,$$

$$\tilde{x}_{..} = \sum_i \frac{w_i \bar{x}_{i.}}{w}.$$

The distribution of W is approximately F with $k-1$ and f degrees of freedom if all population means are equal, even if the variances are unequal, where

$$\frac{1}{f} = \frac{3}{g^2-1}\sum_i \frac{1}{n_i - 1}\left(1 - \frac{w_i}{w}\right)^2. \tag{7.3}$$

Satterthwaite (1941) generalized Welch's idea of the approximate degrees of freedom method to more than two samples, which is known as Satterthwaite's approximate F test.

James(1951) generalized the Welch (1947) series approximation solutions and proposed the test

$$J = \sum_i w_i(\bar{x}_{i.} - \tilde{x}_{..})^2,$$

where

$$w_i = \frac{n_i}{s_i^2},$$

$$w = \sum_i w_i,$$

$$\tilde{x}_{..} = \sum_i \frac{w_i \bar{x}_{i.}}{w}.$$

The critical value of the James test is an asymptotic percentile of χ^2 distribution with $k-1$ degrees of freedom. James noted that if sample sizes are not quite large, this procedure will not be very accurate. James, following Welch (1947), found a series development of a function $h(s_1^2, s_2^2, \ldots, s_k^2) \equiv h(s^2)$ of the variance estimates such that

$$P\Big[\sum w_i(\bar{x}_{i.} - \tilde{x}_{..})^2 \leq 2h(s^2)\Big] = 1 - \alpha. \tag{7.4}$$

James found an approximation of order -2 in the $v_i = n_i - 1$ which is known as James's second-order approximation. The expression is

$$2h(s^2) \simeq c + \frac{1}{2}(3\chi_4 + \chi_2)\sum \frac{1}{v_i}\Big(1 - \frac{w_i}{w}\Big)^2$$

$$+ \left\{ \frac{1}{16}(3\chi_4 + \chi_2)^2\Big(1 - \frac{k-3}{c}\Big)\Big(\sum \frac{1}{v_i}\Big(1 - \frac{w_i}{w}\Big)^2\Big)^2 \right.$$

$$+ \frac{1}{2}(3\chi_4 + \chi_2)\Big[(8R_{23} - 10R_{22} + 4R_{21} - 6R_{12}^2 + 8R_{12}R_{11} - 4R_{11}^2)$$

$$+ (2R_{23} - 4R_{22} + 2R_{12}^2 + 4R_{12}R_{11} - 2R_{11}^2)(c-1))$$

$$+ \frac{1}{4}(-R_{12}^2 + 4R_{12}R_{11} - 2R_{12}R_{10} - 4R_{11}R_{10} - R_{10}^2)(3\chi_2 - 2\chi_2 - 1)\Big]$$

$$+ (R_{23} - 3R_{22} + 3R_{21} - R_{20})(5\chi_6 + 2\chi_4 + \chi_2)$$

$$+ \frac{3}{16}(R_{12}^2 - 4R_{23} + 6R_{22} - 4R_{21} + R_{20})(35\chi_8 + 15\chi_6 + 9\chi_4 + 5\chi_2)$$

$$+ \frac{1}{16}(-2R_{22} + 4R_{21} - R_{20} + 2R_{12}R_{10} - 4R_{11}R_{10} + R_{10}^2)(9\chi_8 - 3\chi_6 - 5\chi_4$$

$$- \chi_2) + \frac{1}{4}(-R_{22} + R_{11}^2)(27\chi_8 + 3\chi_6 + \chi_4 + \chi_2)$$

$$\left. + \frac{1}{4}(R_{23} - R_{12}R_{11})(45\chi_8 + 9\chi_6 + 7\chi_4 + 3\chi_2) \right\}, \tag{7.5}$$

where

$$c = \chi^2_{k-1,1-\alpha}$$

is the $(1-\alpha)$th quantile of the χ^2 distribution with $k-1$ degrees of freedom, and

$$\chi_{2s} = \frac{c^s}{(k-1)(k+1)\cdots(k-2s-3)},$$

$$R_{st} = \sum_i \frac{1}{v_i^s}\left(\frac{w_i}{w}\right).$$

The null hypothesis is rejected in favor of the alternative if $J \geq 2h(s^2)$. This statistic has not been getting much attention in practice because of the complicated nature of the critical value approximation. Recently, however, some have shown interest in this test (Wilcox 1988; Dijkstra and Werter, 1981). To simplify it, James (1951) suggested a first-order approximation, which is

$$2h(s^2) = \chi^2_{k-1,1-\alpha}\left[1 + \frac{3\chi^2_{k-1,1-\alpha} + (k+1)}{2(k^2-1)} \sum \frac{1}{v_i}\left(1 - \frac{w_i}{w}\right)\right]. \tag{7.6}$$

The numerator of the *ANOVA* F test weights the means by n_i, while both James and Welch test use n_i/s_i^2. Brown and Forsythe (1974b, 1974c) use n_i as weights but substitute a denominator whose expectation equals the numerator under the null distribution. Brown and Forsythe defined their test by

$$F^* = \frac{\sum_i n_i(\bar{x}_{1.} - \bar{x}_{..})^2}{\sum_i (1 - n_i/N)^2 s_i^2}. \tag{7.7}$$

Brown and Forsythe use the Satterthwaite's (1941) approximation procedure to compute the critical values of their test. That is, critical values of the F^* test obtained from the F distribution with $k-1$ and f degrees of freedom, where f is determined by

$$\frac{1}{f} = \sum_i \frac{c_i^2}{n_i - 1} \quad \text{and} \quad c_i = \frac{(1 - n_i/N)s_i^2}{\sum_i (1 - n_i/N)s_i^2}. \tag{7.8}$$

When there are only two groups both F^* and W are equivalent to Welch's (1951) *APDF* test.

Rubin (1983) modified the Brown-Forsythe test by substituting

$$g = \left[\sum\left(1-\frac{n_i}{N}\right)s_i^2\right]^2\left[\left(\sum\frac{n_i}{N}s_i^2\right)^2+\sum\left(1-2\frac{n_i}{N}\right)s_i^2\right]^{-1}$$

for the numerator degrees of freedom. The Rubin's test uses the same statistic as the Brown and Forsythe's test, whereas the critical value is obtained from the F distribution with g and f degrees of freedom. Wilcox (1993) generalized his H test to the case of k groups. Wilcox's test statistic is based on the one-step M–estimator and uses a bootstrap confidence interval (Hall, 1988).

Brown and Forsythe (1974c) conducted a simulation, with $\vartheta \leq 3$, for comparing their test to the *ANOVA* F, Welch's and James's first-order test under normality. They summarized that the F test shows marked deviances from the nominal level when the variances are unequal. When the smaller samples have larger variances the F test is much too liberal but too conservative in the opposite situation. On the other hand, the F^* performs at least as good as W and better than J.

Wilcox, Charlin and Thompson (1986) conducted a similar Monte Carlo study as Brown and Forsythe but with $\vartheta \leq 4$. The Monte Carlo study of Wilcox *et al.* reached the same results as Brown and Forsythe (1974c). When the ratio $\vartheta = 4$ and the sample sizes are unequal, however, both F^* and W tests do not provide enough control over Type I errors.

Dijkstra and Werter (1981) compared James's second-order test to the Brown-Forsythe and Welch tests and argued that James's second-order test provided better results with $\vartheta \leq 3$. Roth (1983) and Tomarken and Serlin (1986) also noted the better performance of James's second-order test. Tomarken and Serlin (1986) also

noted that when extreme means are paired with large variances, the Brown-Forsythe test is more powerful than the Welch test and, sometimes, the *ANOVA F*. Wilcox (1988) conducted a simulation with $\vartheta \leq 6$ and supported the results of Dijkstra and Werter.

Tan (1982) approximated the null and nonnull distribution of F^* under normal and nonnormal distributions by using a Laguerre polynomial expression and showed the F^* is quite robust with respect to departure from normality. Tan and Tabatabai (1986b) support the results of Brown and Forsythe by Monte Carlo studies. Tan and Tabatabai (1986a) proposed a test by applying Tiku's *MML* robust procedure to Brown and Forsythe's statistic. Let $x_{i(j)}$ be the jth order statistic in the ith group. Let $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ be defined by:

$$m_i\tilde{\mu}_i = \sum_{j=r_i+1}^{n_i-r_i} x_{i(j)} + r_i\beta_i\{x_{i(r_i+1)} + x_{i(n_i-r_i)}\}, \tag{7.9}$$

and

$$2A_i\tilde{\sigma}_i = B_i + \{B_i^2 + 4A_iC_i\}^{1/2}, \tag{7.10}$$

where

$$C_i = \sum_{j=r_i+1}^{n_i-r_i} (x_{i(j)} - \tilde{\mu}_i)^2 + r_i\beta_i\Big\{(x_{i(r_i+1)} - \tilde{\mu}_i)^2 + (x_{i(n_i+r_i)} - \tilde{\mu}_i)^2\Big\}.$$

For the bias correction let

$$\tilde{\sigma}_{ic} = \tilde{\sigma}_i\left[\frac{A_i}{A_i - 1}\right]^{1/2}.$$

Then Tan and Tabatabai's statistic

$$F_\beta = \frac{\sum_{i=1}^{k} m_i(\tilde{\mu}_i - \tilde{\mu})^2}{\sum_{i=1}^{k}(1 - m_i/m)\tilde{\sigma}_{ic}^2}, \tag{7.11}$$

where

$$m = \sum_{i=1}^{k} m_i \quad \text{and} \quad \tilde{\mu} = \frac{\sum_{i=1}^{k} m_i \tilde{\mu}_i}{m}$$

has approximately an F distribution with $k-1$ and d degrees of freedom, where d is approximated by

$$d = \frac{\left\{\sum_{i=1}^{k}\left(1 - \frac{m_i}{m}\right)\tilde{\sigma}_{ic}^2\right\}^2}{\sum_{i=1}^{k}\left(1 - \frac{m_i}{m}\right)^2 \frac{\tilde{\sigma}_{ic}^4}{A_i - 1}}.$$

Tan and Tabatabai (1986a) compared their test to Brown-Forsythe using Monte Carlo studies under various models, such as normal, chi-squared, uniform, Student's t and mixture of two normal distributions. They concluded that

1. F_β test has essentially the same power as the Brown-Forsythe under the normal distribution.

2. Except for the uniform distribution, the power of the F_β test is considerably higher than the Brown-Forsythe.

Our proposed test statistic is analogous to the Tan-Tabatabai. We will construct our test by applying the weighted likelihood estimation procedure to the Brown-Forsythe and will compare it to the *ANOVA* F, Brown-Forsythe and Tan-Tabatabai with the same simulation scenario as Tan and Tabatabai (1986a).

Let

$$w(\delta^*(x)) = \frac{A(\delta^*(x)) + 1}{\delta^*(x) + 1} \qquad \text{and} \qquad \delta^*(x) = \frac{f^*(x) - m^*_\theta(x)}{m^*_\theta(x)}$$

be the weight function and the Pearson residual, respectively. To find the *WLE*s we need to solve the weighted likelihood estimating equation

$$\int w(\delta^*(x))\nabla_\theta\{\log m_\theta(x)\}dF_n(x) = \frac{1}{n}\sum w(\delta^*(X_j))u(X_j,\theta) = 0$$

where ∇_θ is the (partial) differential operator with respect to the parameter θ, $u(x,\theta)$ is the likelihood score function and F_n is the empirical distribution function. For the k samples case with parameter $\theta_i' = (\mu_i, \sigma_i^2)$ $i = 1, 2, \ldots, k$ we have

$$\mu_i: \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij},\theta_i) = 0, \quad i = 1, 2, \ldots, k,$$

$$\sigma_i^2: \quad \int w_i(\delta_i^*(x))\nabla_{\theta_i}\{\log m_{\theta_i}\}dF_{n_i} = \frac{1}{n_i}\sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))u(X_{ij},\theta_i) = 0, \quad i = 1, 2, \ldots, k.$$

These $2k$ estimating equations can be simplified as

$$\begin{aligned} \mu_i: &\quad \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))(X_{ij} - \mu_i) = 0, \quad i = 1, 2, \ldots, k, \\ \sigma_i^2: &\quad \sum_{j=1}^{n_i} w_i(\delta_i^*(X_{ij}))\{(X_{ij} - \mu_i)^2 - \sigma_i^2\} = 0, \quad i = 1, 2, \ldots, k. \end{aligned} \tag{7.12}$$

The above equations in (7.12) are solved iteratively by creating new weights $w_i(\cdot)$ at

each stage. The solutions of the equation (7.12) are of the form

$$\hat{\mu}_i = \frac{\sum_{j=1}^{n_i} w_i^{1/2}\big(\delta_i^*(X_{ij})\big)X_{ij}}{\sum_{j=1}^{n_i} w_i^{1/2}\big(\delta_i^*(X_{ij})\big)}, \quad i = 1, 2, \ldots, k,$$

$$\hat{\sigma}_i^2 = \frac{\sum_{j=1}^{n_i} w_i^{1/2}\big(\delta_i^*(X_{ij})\big)\big(X_{ij} - \hat{\mu}_i\big)^2}{\sum_{j=1}^{n_i} w_i^{1/2}\big(\delta_i^*(X_{ij})\big)}, \quad i = 1, 2, \ldots, k. \tag{7.13}$$

Note that the square root of the weight $w(\delta^*)$ is taken at the final stage. That is, the original weights are used to solve the equations (7.12). However, square root of the weights are taken after the equations are solved to get the square root weighted *WLE*s.

Let $\hat{\mu}_i$ and $\hat{\sigma}_i^2$, $i = 1, 2, \ldots, k$, be *WLE*s for means and variances from k groups. Let $w_i(\delta_i^*)$ be the sum of n_i square root weights produced by the weighted likelihood estimating equation to estimate $\hat{\mu}_i$. Then the test statistic based on the weighted likelihood estimates with the Hellinger distance weights is

$$F_{HD} = \frac{\sum_{i=1}^{k} n_i(\hat{\mu}_i - \hat{\mu})^2}{\sum_{i=1}^{k}\Big(1 - \frac{n_i}{N}\Big)\hat{\sigma}_i^2}, \tag{7.14}$$

where

$$\hat{\mu} = \frac{\sum_{i=1}^{k} \hat{\mu}_i}{\sum_{i=1}^{k} w_i(\delta_i^*)}.$$

Critical values of the test are computed from the F distribution with $k-1$ and ξ degrees of freedom where ξ is implicitly defined by the Satterthwaite's (1941) approximation as that of the Brown-Forsythe, i.e.,

$$\frac{1}{\xi} = \sum_{i=1}^{k} \frac{b_i^2}{n_i - 1},$$

and

$$b_i = \frac{(1 - n_i/N)\hat{\sigma}_i^2}{\sum_i (1 - n_i/N)\hat{\sigma}_i^2}.$$

Simulations are done for the comparative studies. James's second order statistic is not included in the simulations because of the complicated procedure. The program codes are attached in Appendix M and Appendix N.

7.3 Simulation and Results

Tan and Tabatabai's (1986a, 1986b) simulation setup for comparing three population means is followed here. Four distributions are considered: (1) $N(0,1)$; (2) $t(4)$; (3) $\chi^2(3)$; and (4) $pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where $p = 0.1$, $\Delta = 2$, $r^2 = 0.25$ and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$. These distributions are properly centered and scaled so that for the ith population, the mean is μ_i and variance is σ_i^2. The parameter values are taken arbitrarily as $k = 3$, (n_1, n_2, n_3)=(12,10,8), and (24,20,16); $(\sigma_1^2, \sigma_2^2, \sigma_1^3)$=(11,11,11), (71,8,16) and (12,56,16). The σ_i^2 values are chosen to cover three patterns (i) $\sigma_1^2 = \sigma_2^2 = \sigma_3^2$, (ii) $\sigma_2^2 > \sigma_i^2$ $i = 1, 3$, and (iii) $\sigma_2^2 < \sigma_i^2$, $i = 1, 3$. Tan and Tabatabai noted that they chose $\sigma_i^2 > 10$ because their statistic is so powerful, otherwise all the power values equal one. The value of the μ_i are chosen to correspond to $\sum n_i(\mu_i - \mu)^2 / \sum n_i = 0$, 2.19, 5.16, 8.97.

Five thousand samples were generated for each distribution and each parameter setup. Using Borland® C^{++} and IMSL® C/Stat/Library in programming and run on the Pentium 133Hz PC. The number of rejections out of 5,000 runs were counted for each of the tests at the nominal levels of $\alpha = 0.5$ and 0.1. The sizes and powers of the tests are given in Tables G.1 and G.2 in Appendix G when samples sizes are (12,10,8) and Table G.3 and Table G.4 when samples sizes are (24,20,16).

The following observations were made by investigating the simulation results.

- Under the mixture distribution, F_{HD} and F_β showed basically the same power in all cases, but F^* showed slightly lower power than these two.

- For the chi-squared and uniform distributions, the F_{HD} appears considerably more powerful.

- There is a slight difference in power under normal and Student's t distributions. The pattern of the power is $F_\beta < F^* < F_{HD}$ for normal distribution and $F^* < F_{HD} < F_\beta$ for Student's t distribution.

For $k = 4$, i.e., when we have four groups, we will use a different simulation setup. In this simulation, we will keep the same distributions as above. The values of the μ_i and σ_i^2 are selected arbitrarily, without any restriction. The setup for these simulations was made by referring to Brown and Forsythe (1974b) and Wilcox(1988).

The selected sample sizes are (n_1, n_2, n_3, n_4) = (11,11,11,11), (11,16,16,21) and (50,50,50,50) and the parameter values are $(\mu_1, \mu_2, \mu_3, \mu_4)$ = (0,0,0,0), (0,5,0,0,0) and (0,0,0,0.7); $(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$ = (1:1:1:1), (16:1:1:1), (1:1:1:16), (1:4:9:16) and (16:9:4:1). Note that the given values of σ_i^2 are not the assumed values of population variances, unlike in Tan and Tabatabai's setup, but simply ratios of population variances.

The results are given in Tables H.1, H.2 and H.3 for the nominal level $\alpha = 0.05$ in Appendix H and in Tables H.4, H.5, and H.6, for $\alpha = 0.01$. The tables are summarized as follows:

- The results are basically the same as when $k = 3$.

- One striking result is that the F_{HD} shows considerably higher powers than F_β while they are alike in the Tan and Tabatabai setup.

- Both F_β and F_{HD} show slightly higher empirical levels than the nominal level, especially for the chi-squared distribution.

7.4 Concluding Remarks

Simulation results for testing equality of more than two population means showed that the overall performance of the proposed test based on the *WLE*s is satisfactory in all cases. Results are encouraging in that the tests based on the Hellinger distance weighted *WLE*s are robust and powerful. The *HD* tests are shown to be a robust alternative to the Tan-Tabatabai test based on Tiku's *MMLE*s.

BIBLIOGRAPHY

Andrews, D. F., Bickel, P. J., Hampel, F. R., Huber, P. J. Rogers, W. H. and Tukey, J. W. (1972). *Robust Estimates of Location: Survey and Advances.* Princeton University Press. Princeton, New Jersey.

Aspin, A. A. (1948). An examination and further development of a formula arising in the problem of comparing two mean values. *Biometrika,* **35**, 88-96.

Aspin, A. A. (1949). Tables for use in comparisons whose accuracy involves two variances, separately estimated. (with Appendix by B. L. Welch). *Biometrika,* **36**, 290–296.

Ash, R. B. (1972). *Real Analysis and Probability.* Academic Press. New York, New York.

Banerjee, S. K. (1960). Approximate confidence interval for linear functions of means of k populations when the population variances are unequal. *Sankhyā,* **22**, 357–358.

Banerjee, S. K. (1961). On confidence interval for two-means problem based on separate estimates of variances and tabulated values of t-table. *Sankhyā,* **23**, 359–378.

Basu, A. (1991). *Minimum Disparity Estimation in the Continuous Case: Efficiency, Distributions, Robustness, and Algorithms* Unpublished Ph.D. Dissertation, Pennsylvania State University, University Park, Pennsylvania.

Basu, A., Harris, I. R. and Basu, S. (1996). Test of hypotheses in discrete models based on the penalized Hellinger distance. *Statistics & Probability Letters,* **27**, 367–373.

Basu, A. and Lindsay, B. G. (1993). The iteratively reweighted estimating equation in minimum distance problems. *Technical Report,* Department of Mathematics, University of Texas, Austin, Texas.

Basu, A. and Lindsay, B. G. (1994). Minimum Disparity estimation in the continuous Case: Efficiency, Distributions, Robustness. *Annals of the Institute of Statistical Mathematics,* **46**, 683–705.

Basu, A., Markatou, M. and Lindsay, B. G. (1995). Weighted likelihood estimating equations: The continuous case. *Technical Report,* Department of Statistics, Columbia University, New York.

Basu, A., and Sarkar, S. (1994a). On disparity based goodness-of-fit tests for multinomial models. *Statistics & Probability Letters,* **19**, 307–312.

Basu, A., and Sarkar, S. (1994b). Minimum disparity estimation in the errors-in-variable model. *Statistics & Probability Letters,* **20**, 69–73.

Basu, A., and Sarkar, S. (1994c). The trade-off between robustness and efficiency and the effect of model smoothing in minimum disparity inference. *Journal of Statistical Computation and Simulation,* **50**, 173–185.

Basu, A., and Sarkar, S. (1997). Robust estimation in the errors-in-variable model via weighted likelihood estimating equation. *Test,* to appear.

Basu, A., Sarkar, S. and Basu, B. (1997). Robust tests for equality of two population means under the normal model. *Communications in Statistics: Simulation and Computation,* **26(1)**, 333–353.

Basu, A., Sarkar, S. and Vidyashankar, A. N. (1997). Minimum Negative Exponential Disparity Estimation in Parametric Models. *Journal of Statistical Planning and Inference,* **58**, 349–370.

Bartlett, B. A. (1935). The effect of non-normality on the t distribution. *Proceedings of the Cambridge Philosophical Society,* **31**, 223-231.

Bartlett, B. A. (1937). Properties of sufficiency and statistical tests. *Proceedings of the Royal Society of London,* **A 160**, 268–282.

Behrens, W. V. (1929). Ein Beitrag zur Fehlerberechung bei wenigen Beobachtungen. *Landwirtsch. Jahrbucher,* **68**, 807–837.

Beran, R. J. (1977a). Robust location estimates. *Annals of Statistics,* **5**, 431–444.

Beran, R. J. (1977b). Minimum Hellinger distance estimates for parametric models. *Annals of Statistics,* **5**, 445–463.

Berkson, J. (1980). Minimum chi-square, not maximum likelihood! (with discussion), *Annals of Statistics,* **8**, 457–487.

Best, D. J. and Rayner, J. C. W. (1987). Welch's approximation solution for the Behrens-Fisher problem. *Technometrics,* **29**, 205–210.

Bhattacharyya, A. (1943). On a measure of divergence between two statistical populations defined by their probability distribution. *Bulletin of the Calcutta Mathematical Society,* **35**, 99–109.

Borland® C^{++}. (1994). *Programmer's Guide*, 1994, Borland International,Inc., Scotts Valley, California.

Boneau, C. A. (1960). The effects of violations of assumptions underlying the t test. *Psychological Bulletin,* **57**, 49–64.

Box, G. E. P. (1953). Non-normality and tests on variances, *Biometrika,* **40**, 318–335.

Box, G. E. P. (1954a). Some theorems on quadratic forms applied in the study of analysis of variance problem, I. Effect of inequality of variance in the one-way classification. *Annals of Mathematical Statistics,* **25**, 484–498.

Box, G. E. P. (1954b). Some theorems on quadratic forms applied in the study of analysis of variance problem, II. Effect of inequality of variance and of correlation between errors in the two-way classification. *Annals of Mathematical Statistics,* **25**, 290–302.

Box, G. E. P. and Anderson, S. L. (1955). Permutation theory in the derivation of robust criteria and the study of departures from assumption. *Journal of the Royal Statistical Society,* **17**, 1–34.

Box, G. E. P. and Tiao, G. C. (1964). A note on criterion robustness and inference robustness. *Biometrics,* **51**, 169–173.

Bradley, J. V. (1978). Robustness? *British Journal of Mathematical and Statistical Psychology,* **34**, 144–152.

Brown, B. M. (1982). Robustness against inequality of variances. *Australian Journal of Statistics,* **24**, 283–295.

Brown, M. B. and Forsythe, A. B. (1974a). Robust test for the equality of variance, *Journal of the American Statistical Association,* **69**, 364–367.

Brown, M. B. and Forsythe, A. B. (1974b). The small sample behavior of some statistics which test several means. *Technometrics,* **16**, 129–132.

Brown, M. B. and Forsythe, A. B. (1974c). The ANOVA and multiple comparison for data with heterogeneous variance, *Biometrics,* **40**, 717–724.

Clinch, J. J. and Keselman, H. J. (1982). Parametric alternatives to the analysis of variances. *Journal of Educational Statistics,* **7**, 207–214.

Cochran, W. G. (1947). Some consequences when assumptions for the analysis of variance are not satisfied. *Biometrics,* **3**, 22–38.

Cochran, W. G. (1951). Testing a linear relation among variances. *Biometrics,* **7**, 17–32.

Cochran, W. G. (1964). Approximate significant levels of the Behrens-Fisher problem. *Biometrics,* **20**, 191–195.

Cohen, A. C. Jr. (1950). Estimating the mean and variance of normal populations from singly truncated and doubly truncated samples. *Annals of Mathematical Statistics,* **21**, 557–569.

Cohen, A. C. Jr. (1957). On the solution of estimating equations from truncated and censored samples from normal populations. *Biometrics,* **44**, 225–236.

Cohen, A. C. Jr. (1959). Simplified estimators for the normal distribution when samples are singly censored or truncated. *Technometrics,* **1**, 217–237.

Cohen, A. C. Jr. (1961). Tables for maximum likelihood estimates; singly censored sample. *Technometrics,* **3**, 535–541.

Cohen, A. C. Jr. and Woodward, J. (1953). Tables of Pearson-Lee-Fisher functions of singly truncated normal distributions. *Biometrics,* **9**, 489–497.

Conover, W. J. (1980). *Practical Nonparametric Statistics*, Wiley, New York, New York.

Conover, W. J., Johnson, M. E. and Johnson, M. M. (1981) A comparative study of tests for homogeneity of variances, with applications to the outer continental shelf bidding data, *Technometrics,* **23**, 315–361.

Coombs, W. T., Algina, J. and Oltman, D. O. (1996). Univariate and multivariate omnibus hypothesis tests selected to control type I error rates when population variances are not necessarily equal. *Review of Educational Research,* **66**, 137–179.

Cressie, N. A. C. and Read, T. R. C. (1984). Multinomial goodness-of-fit tests. *Journal of the Royal Statistical Society,* **B 46**, 440–464.

Cressie, N. A. C. and and Whitford, H. J. (1986). How to use the two-sample t-test. *Biometrical Journal,* **18**, 131–148.

Daniell, P. J. (1920). Observations weighted according to order. *Journal of the American Statistical Association,* **42**, 222–36.

Daniels, H. E. (1938). The effect of departures from ideal conditions other than non-normality on the t and z tests of significance. *Proceedings of the Cambridge Philosophical Society.* **34**, 321–328.

Davenport, J. M. and Webster, J. R. (1975). The Behrens-Fisher problem, an old solution revisited. *Metrika,* **22**, 47–54.

David, H. A. (1981). *Order Statistics.* Wiley, New York, New York.

Dijkstra, J. B. and Werter, P. S. P. J. (1981). Testing the equality of several means when the population variances are unequal. *Communications in Statistics: Simulation and Computation,* B **10**, 557–569.

Devroye, L. and Györfi, L. (1985). *Nonparametric Density Estimation: The L_1 view.* Wiley, New York, New York.

Donoho, D. L. and Huber, P. J. (1983). The notion of breakdown points. in *A Festschrift for Erich L. Lehmann,* (edited by P. J. Bickel, K. A. Doksum, and J. L. Hodges). Wadsworth, Belmont, California.

Donoho, D. L. and Liu, R. C. (1988). The automatic robustness of minimum distance functionals. *Annals of Statistics,* **16**, 552–568.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans. Society for Industrial and Applied Mathematics,* Philadelphia, Pennsylvania.

Efron, B. (1987). Better bootstrap confidence interval. *Journal of the American Statistical Association,* **82**, 171–185.

Fairfield-Smith, H. (1936). The problem of comparing the results of two experiments with unequal errors. *Journal of the Council of Scientific and Industrial Research,* **9**, 211–212.

Fenstad, G. U. (1983). A comparison between the U and V tests in the Behrens-Fisher problem. *Biometrika,* **70**, 300-302.

Fisher, R. A. (1935). The fiducial argument in statistical inference. *Annals of Eugenics*, **6**, 391–398.

Fisher, R. A. (1939). The comparison of samples with possible unequal variances. *Annals of Eugenics*, **9**, 174–180.

Fligner, M. A. and Killeen, T. J. (1976). Distribution-free two sample tests for scale. *Journal of the American Statistical Association,* **71**, 210–213.

Gartside, P. S. (1972). A study of methods for comparing several variances. *Journal of the American Statistical Association,* **67**, 342–346.

Gayen, A. K. (1950). The distribution of the variance ratio in random samples of any size drawn from non-normal universes. *Biometrics,* **37**, 236–255.

Geary, R. C. (1947). Testing for normality. *Biometrika,* **34**, 209–242.

Gupta, A. K. (1952). Estimation of the mean and standard deviation of a normal population from a censored sample. *Biometrika,* **39**, 260–273.

Hall, P. (1983). Large sample optimality of least squares cross-validation in density estimation. *Annals of Statistics,* **11**, 1156–1174.

Hall, P. (1986). On the number of bootstrap simulations required to construct a confidence interval. *Annals of Statistics,* **14**, 1453–1462.

Hall, P. (1988). On symmetric bootstrap confidence intervals. *Journal of the Royal Statistical Society,* **B 50**, 35–45.

Hall, P., Hu, T. C. and Marron, J. S. (1995). Improved variable window kernel estimates of probability densities. *Annals of Statistics,* **23**, 1–10.

Hampel, F. R. (1968). *Contributions to the theory of robust estimation.* Unpublished Ph.D Thesis. University of California, Berkeley, California.

Hampel, F. R. (1971). A general qualitative definition of robustness. *Annals of Mathematical Statistics,* **42**, 1887–1896.

Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association,* **69**, 383–393.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J. and Stahel, W. A. (1986). *Robust Statistics: The Approach Based On Influence Functions.* Wiley, New York, New York.

Harrell, F. E. and Davis, C. E. (1982). A new distribution free quantile estimator. *Biometrika,* **69**, 635–640.

Harris, I. R. and Basu, A. (1994). Hellinger distance as a penalized log likelihood, *Communications in Statistics: Simulation and Computation,* **23**, 1097–1113.

Harter, H. L. and Moore, A. H. (1966). Local maximum likelihood estimation of the parameters of three-parameter log-normal populations from complete and censored samples, *Journal of the American Statistical Association,* **61**, 842–851.

Havlicek, L.L. and Peterson, N. L. (1974). Robustness of the t test: A guide for researchers on effect of violations of assumptions. *Psychological Reports,* **34**, 1095–1114.

He, X., Simpson, D. G. and Portnoy, S. L. (1990). Breakdown robustness of tests. *Journal of the American Statistical Association,* **85**, 446–452.

Horsnell, G. (1953). The effect of unequal group variances on the F-test for the homogeneity of group means. *Biometrics,* **40**, 128–136.

Huber, P. J. (1964). Robust estimation of location parameter. *Annals of Mathematical Statistics,* **35**, 73–101.

Huber, P. J. (1972). Robust statistics: A review. *Annals of Mathematical Statistics,* **43**, 1041–1067.

Huber, P. J. (1977). *Robust Statistical Procedures. Society for Industrial and Applied Mathematics,* Philadelphia, Pennsylvania.

Huber, P. J. (1981). *Robust Statistics.* Wiley, New York, New York.

IMSL® C/Stat/Library. (1995) *C functions for statistical analysis.* Visual Numerics, Houston, Texas.

IMSL® C/Math/Library. (1995) *C functions for mathematical applications.* Visual Numerics, Houston, Texas.

James, A. T. (1965). Distributions of matrix variates and laten roots derived from normal sample. *Annals of Mathematical Statistics,* **34**, 475–501.

James, G. S. (1951). The comparison of several groups of observations when the ratios of the population variances are unknown. *Biometrics,* **38**, 324–329.

James, G. S. (1954). Test of linear hypotheses in univariate and multivariate analysis when the ratios of the population variances are unknown. *Biometrika,* **41**, 19–43.

Johnson R. A. (1988) *Unpublished lecture notes*, University of Wisconsin–Madison, Madison, Wisconsin.

Lawless, J. F. (1982). *Statistical Models and Methods for Lifetime Data.* Wiley, New York, New York.

Layard, M. W. J. (1973). Robust large-sample tests for homogeneity of variance. *Journal of the American Statistical Association,* **68**, 195–198.

Levene, H. (1960). Robust tests for equality of variances, in *Contributions to Probability and Statistics* (edited by I. Olkin, S. G. Ghurye, W. Hoeffding, W. G. Madow, H. B. Mann). Stanford University Press, Stanford, California.

Lindquist, E. F. (1953). *Design and analysis of experiments in psychology and education.* Houghton Mifflin, Boston, Massachusetts.

Lindsay, B. G. (1994). Efficiency versus robustness: The case for minimum Hellinger distance and related methods. *Annals of Statistics,* **22**, 1081–1114.

Mason, R. L., Gunst, R. F. and Hess, J. L. (1989). *Statistical Design and Analysis of Experiment with Applications to Engineering and Science.* Wiley, New York, New York.

Markatou, M., Basu, A. and Lindsay, B. (1995) Weighted likelihood estimating equations: The discrete case with applications to logistic regression. *Technical Report,* Department of Statistics, Columbia University, New York, New York.

Matusita, K. (1955). Decision rules, based on the distance, for problems of fit, two samples and estimation. *Annals of Mathematical Statistics,* **26**, 631–640.

Miller, R. G. (1968). Jackknifing variance. *Annals of Mathematical Statistics,* **39**, 567–582.

Milliken G. A. and Johnson D. E. (1984). *Analysis of Messy Data, Volume 1:Designed Experiments.* Van Nostrand Reinhold Company, New York, New York.

Moser B. K., Stevens G. R. and Watts C. L. (1989). The two-sample t test versus Satterthwaite's approximate F test. *Communications in Statistics: Theory and Method,* **18(11)**, 3963–3975.

Nel, D. G., van der Merwe, C. A. and Moser B. K. (1990). The exact distributions of the univariate and multivariate Behrens-Fisher statistics with a comparison of several solutions in the univariate case. *Communications in Statistics: Theory and Method,* **19(1)**, 279–298.

Newcomb, S. (1878). Researches on the motion of the Moon, I. *Washington Observations for 1875*–Appendix II; U.S. Naval Observatory, Washington.

Ostle, B. and Mensing, R. W. (1975). *Statistics in research.* Iowa State University Press, Ames, Iowa.

Papaioannou, P. C. and Kempthorne, O. (1971). *On statistical information theory and related measures of information.* Aerospace Research Laboratories Report ARL71-0059, U.S. Defense Documentation Center, Alexandrian, Virginia.

Prakasa Rao, B. L. S. (1983). *Nonparametric Functional Estimation.* Academic Press. New York, New York.

Pearson, E. S. (1931). The analysis of variance in the case of non-normal variation. *Biometrika,* **23**, 114–133.

Pearson, E. S. and Hartley, H. O. (1972). *Biometrika Tables for Statisticians, Vol. 2.* Cambridge University Press, Cambridge.

Pitman, E. J. G. (1979) *Some basic theory for statistical inferences.* Chapman and Hall, London.

Pratt, J. W. (1964). Robustness of some procedures for the two-sample location problem. *Journal of the American Statistical Association,* **59**, 665–680.

Ramsey, P. H. (1980). Exact Type T error rates for robustness of Student's t test with unequal variances. *Journal of Educational Statistics,* **5**, 337–349.

Randles, R. H. and Wolfe, D. A. (1979). *Introduction to the theory of nonparametric statistics.* Wiley, New York, New York.

Rao, C. R. (1961). Asymptotic efficiency and limiting information, *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability,* University of California, **1**, 531–546.

Rao, C. R. (1963) Criteria of estimation in large samples. *Sankhyā,* **25**, 189–206.

Rao, C. R. (1973) *Linear Statistical Inference and Its Applications.* Wiley, New York, New York.

Read, T. R. C. and Cressie, N. (1988). *Goodness-of-Fit Statistics for Discrete Multivariate Data.* Springer-Verlag, New York, New York.

Rogan, J. C. and Keselman, H. J. (1977). Is then ANOVA F-test robust to variance heterogeneity when sample sizes are equal?: An investigation via coefficient of variation. *American Educational Research Journal,* **14**, 493–498.

Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics,* **27**, 832–837.

Roth, A. J. (1983). Robust trend test derived and simulated: Analogs of the Welch and Brown-Forsythe tests. *Journal of the American Statistical Association,* **78**, 972–980.

Rubin, A. S. (1983). The use of weighted contrast in analysis of models with heterogeneity of variance. *American Statistical association: Proceedings of the Business and Economics Statistics Section,* 347–352.

Ryan, J. A., Joiner, B. L. and Ryan, B. F. (1982), *MINITAB Reference Manual,* Buxbury Press, Boston, Massachusetts.

Sarkar, S. and Basu, A. (1995). On disparity based robust tests for two discrete populations. *Sankhyā* **B 57**, 353–364.

Satterthwaite, F. E. (1941). Synthesis of variance. *Psychometrika,* **6**, 309–316.

Scheffé, H. (1943). On solutions of the Behrens-Fisher problem. *Annals of Mathematical Statistics,* **10**, 129–137.

Scheffé, H. (1959). *The Analysis of Variance.* Wiley, New York, New York.

Serfling R. J. (1980). *Approximation theorems of Mathematical Statistics.* Wiley, New York, New York.

Shin, D. W., Basu, A. and Sarkar, S. (1995). Comparisons of the blended weight Hellinger distance based goodness-of-fit test statistics. *Sankhyā,* **B 57**, 365–376.

Shin, D. W., Basu, A. and Sarkar, S. (1995). Small sample comparison for the blended weight chi-square goodness-of-fit test statistics. *Communications in Statistics: Theory and Method,* **25**, 211–226.

Simonoff, J. S. (1996) *Smoothing Methods in Statistics.* Springer-Verlag, New York, New York.

Simpson, D. G. (1987). Minimum Hellinger distance estimation for the analysis of count data. *Journal of the American Statistical Association,* **82**, 802-807.

Simpson, D. G. (1989). Hellinger deviance test: Efficiency, breakdown points and examples. *Journal of the American Statistical Association,* **84**, 107–113.

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis.* Chapman and Hall. New York, New York.

Srivastava, A. B. L. (1959). Effect of non-normality on the power of the analysis of variances. *Biometrika,* **46**, 114–122.

Stather C. R. (1981) *Robust statistical inference using Hellinger distance methods* Unpublished Ph.D. dissertation. Latrobe University, Melbourne, Australia.

Staudte, R. G. and Sheather, S. J. (1990). *Robust Estimation and Testing.* Wiley, New York, New York.

Stigler, S. M. (1973) Simon Newcombe, Percy Daniell, and History of Robust estimation 1885–1920, *Journal of the American Statistical Association,* **68**, 872–879.

Tamura, R. N. and Boos, D. D. (1986). Minimum Hellinger distance estimation for multivariate location and covariance. *Journal of the American Statistical Association,* **81**, 223–229.

Tan, W. Y. (1982). Sampling distributions and robustness of t, F and variance-ratio in two samples and ANOVA models with respect to departure from normality. *Communications in Statistics: Theory and Method,* **11(22)**, 2485–2511.

Tan, W. Y. and Tabatabai, M. A. (1985). Some robust ANOVA procedures under heteroscedasticity and nonnormality. *Communications in Statistics: Simulation and Computation,* **14(4)**, 1007–1026.

Tan, W. Y. and Tabatabai, M. A. (1986a). A robust procedure for comparing several means under heteroscedasticity and nonnormality. *Communications in Statistics: Simulation and Computation,* **15(3)**, 733–745.

Tan, W. Y. and Tabatabai, M. A. (1986b). Some Monte Carlo studies on the comparison of several means under heteroscedasticity and robustness with respect to departure from normality. *Biometrical Journal,* **28**, 801–814.

Tiku, M.L. (1964). Approximating the general nonnormal variance-ratio sampling distributions, *Biometrika*, **51**, 83–95.

Tiku, M. L. (1967). Estimating the mean and standard deviation from the censored normal sample. *Biometrika*, **54**, 155–165.

Tiku, M. L. (1970). Mote Carlo study of some simple estimators in censored normal samples. *Biometrika*, **57**, 207–211.

Tiku, M. L. (1971). Power function of the F-test under non-normal situations. *Journal of the American Statistical Association,* **66**, 913–916.

Tiku, M. L. (1980). Robustness of MML Estimators based on Censored samples and Robust test statistics. *Journal of Statistical Planning and Inference,* **4**, 123–143.

Tiku, M. L. (1982). Robust statistics for testing equality of means or variances. *Communications in Statistics: Theory and Method,* **11(22)**, 2543–2558.

Tiku, M. L. and Balakrishnan N. (1984). Testing equality of population variance the robust way. *Communications in Statistics: Theory and Method,* **13(17)**, 2143–2159.

Tiku, M. L. and Sing, M. (1981). Robust test for means when population variances are unequal. *Communications in Statistics: Theory and Method,* **A 10(13)**, 2057–2071.

Tiku, M. L. and Tan, W. Y., and Balakrishnan N. (1986). *Robust Inference.* Marcel Dekker, New York, New York.

Tomarken, A. and Serlin, M. (1986). Comparison of ANOVA alternatives under variance heterogeneity and specific noncentrality structures. *Psychological Bulletin,* **99**, 90–99.

Tukey, J. W. (1960). A survey of sampling from contaminated distributions, in *Contributions to Probability and Statistics* (edited by I. Olkin, S. G. Ghurye, W. Hoeffding, W. G. Madow, H. B. Mann). Stanford University Press, Stanford, California.

Tukey, J. W. and McLaughlin, D. H. (1963). Less vulnerable confidence and significance procedures for location based on a single sample: Trimming/Wisorizing. *Sankhyā,* **A25**, 331–352.

von Mises, R. (1947). On the asymptotic distribution of differentiable statistical functionals, *Annals of Mathematical Statistics,* **18**, 309–348.

Wang, Y. Y. (1971). Probabilities of the type I errors of the Welch tests for the Behrens-Fisher problem. *Journal of the American Statistical Association,* **66**, 605–608.

Welch, B. L. (1938). The significance of the difference between two means when the population variances are unequal. *Biometrika,* **29**, 2057–2071.

Welch, B. L. (1947). The generalization of student's problem when several different population variances are involved. *Biometrika,* **34**, 28–35.

Welch, B. L. (1949). Appendix to A. A. Aspin's tables. *Biometrika,* **36**, 293–6.

Welch, B. L. (1951). On the comparison of several mean values: An alternative approach. *Biometrika,* **38**, 330–336.

Welsh, A. H. and Morrison, H. L. (1990). Robust L estimation of scale with an application in astronomy. *Journal of the American Statistical Association,* **85**, 729–743.

Wilcox, R. R. (1987). New designs in analysis of variance. *Annual Review of Psychology,* **38**, 29–60.

Wilcox, R. R. (1988). A new alternative to the ANOVA F and new results on James's second-order method. *British Journal of Mathematical and Statistical Psychology,* **41**, 109–117.

Wilcox, R. R. (1990). Comparing the means of two independent groups. *Biometrical Journal,* **32**, 771–780.

Wilcox, R. R. (1992). Comparing one-step M-estimator of location corresponding to two independent groups. (1993). *Psychometrika,* **57**, 141-154.

Wilcox, R. R. (1993). Comparing one-step M-estimator of location when there are more than two groups. (1993). *Psychometrika,* **58**, 71–78.

Wilcox, R. R. (1994). Some results on the Tukey-McLaughlin and Yuen methods for trimmed means when distributions are skewed. *Biometrical Journal,* **36**, 259–273.

Wilcox, R. R. (1995a). ANOVA: A paradigm for low power and misleading measures of effect size. *Review of Educational Research,* **65**, 51–77.

Wilcox, R. R. (1995b) A note on testing hypotheses about trimmed mean. *Biometrical Journal,* **38**, 173–180.

Wilcox, R. R. (1996). A note on testing hypotheses about trimmed means. *Biometrical Journal,* **38**, 173–180.

Wilcox, R. R. (1997). Simulation results on performing pairwise comparisons of trimmed means. *Submitted to Communication in Statistics.*

Wilcox R. R., Charlin V. L. and Thompson K. L. (1986). New Monte Carlo results on the robustness of the ANOVA F, W, and F^* statistics. *Communications in Statistics: Simulation and Computation,* **15(4)**, 933-943.

Wolfowitz, J. (1953). Estimation by the minimum distance method. *Annals of the Institute of Statistical Mathematics,* **5**, 9–23.

Wolfowitz, J. (1954). Estimation by the minimum distance method in nonparametric difference equations. *Annals of Mathematical Statistics,* **25**, 132–157.

Wolfowitz, J. (1957). The minimum distance method. *Annals of Mathematical Statistics,* **28**, 75–88

Yuen, K. K. and Dixon, W. J. (1973). The approximate behavior and performance of the two-sample trimmed t. *Biometrika,* **60**, 369–374.

Yuen, K. K. (1974). The two-sample trimmed t for unequal population variances. *Biometrika*, **61**, 165–170.

APPENDIX A

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR THE EQUALITY OF TWO POPULATION MEANS WHEN THEIR VARIANCES ARE EQUAL

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.084	0.312	0.670	0.918	0.989	1.000	1.000
	NED(0.15)	0.061	0.264	0.618	0.899	0.987	0.999	1.000
	NED(0.30)	0.070	0.281	0.634	0.910	0.988	0.999	1.000
	NED(0.50)	0.086	0.307	0.661	0.913	0.989	1.000	1.000
	NED(1.00)	0.114	0.347	0.689	0.914	0.989	1.000	1.000
	NED(2.00)	0.149	0.392	0.707	0.916	0.988	0.999	1.000
	LD(MLE)	0.057	0.278	0.642	0.914	0.989	1.000	1.000
	TIKU	0.057	0.244	0.572	0.857	0.979	0.998	1.000
	t	0.049	0.246	0.595	0.889	0.987	0.999	1.000
$\alpha = 0.01$	HD	0.029	0.136	0.393	0.743	0.937	0.992	0.999
	NED(0.15)	0.016	0.092	0.315	0.678	0.913	0.986	0.999
	NED(0.30)	0.021	0.109	0.333	0.700	0.920	0.988	0.999
	NED(0.50)	0.032	0.137	0.372	0.725	0.929	0.990	0.999
	NED(1.00)	0.053	0.179	0.447	0.764	0.940	0.992	0.999
	NED(2.00)	0.081	0.231	0.503	0.783	0.950	0.992	0.999
	LD(MLE)	0.014	0.101	0.337	0.702	0.925	0.989	0.999
	TIKU	0.013	0.076	0.264	0.587	0.852	0.968	0.996
	t	0.010	0.079	0.280	0.650	0.899	0.984	0.999

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$0.9N(0,1) + 0.1t(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.081	0.310	0.654	0.891	0.982	0.998	1.000
	NED(0.15)	0.061	0.268	0.601	0.852	0.962	0.991	0.998
	NED(0.30)	0.069	0.283	0.622	0.868	0.973	0.995	0.999
	NED(0.50)	0.080	0.305	0.646	0.885	0.980	0.998	1.000
	NED(1.00)	0.113	0.353	0.674	0.898	0.981	0.998	1.000
	NED(2.00)	0.140	0.393	0.694	0.904	0.983	0.998	1.000
	LD(MLE)	0.049	0.219	0.503	0.720	0.837	0.880	0.907
	TIKU	0.050	0.232	0.543	0.820	0.957	0.993	0.998
	t	0.038	0.191	0.462	0.691	0.823	0.872	0.901
$\alpha = 0.01$	HD	0.026	0.139	0.398	0.706	0.916	0.985	0.997
	NED(0.15)	0.013	0.091	0.313	0.631	0.864	0.963	0.988
	NED(0.30)	0.019	0.109	0.340	0.665	0.890	0.975	0.994
	NED(0.50)	0.026	0.133	0.386	0.693	0.910	0.982	0.996
	NED(1.00)	0.052	0.177	0.443	0.737	0.927	0.989	0.998
	NED(2.00)	0.072	0.231	0.494	0.761	0.934	0.990	0.998
	LD(MLE)	0.009	0.068	0.247	0.498	0.703	0.813	0.857
	TIKU	0.009	0.072	0.244	0.538	0.806	0.943	0.987
	t	0.007	0.049	0.201	0.449	0.665	0.794	0.847

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.091	0.311	0.626	0.881	0.980	0.998	0.999
	NED(0.15)	0.074	0.265	0.567	0.837	0.953	0.987	0.998
	NED(0.30)	0.081	0.283	0.591	0.858	0.968	0.994	0.999
	NED(0.50)	0.093	0.307	0.617	0.874	0.976	0.998	0.999
	NED(1.00)	0.119	0.348	0.648	0.889	0.979	0.999	1.000
	NED(2.00)	0.152	0.389	0.675	0.898	0.981	0.999	1.000
	LD(MLE)	0.056	0.198	0.444	0.674	0.800	0.851	0.881
	TIKU	0.055	0.218	0.512	0.805	0.947	0.993	0.999
	t	0.045	0.170	0.407	0.640	0.784	0.841	0.873
$\alpha = 0.01$	HD	0.032	0.133	0.366	0.687	0.896	0.981	0.997
	NED(0.15)	0.021	0.096	0.309	0.607	0.833	0.950	0.984
	NED(0.30)	0.026	0.112	0.335	0.639	0.863	0.969	0.993
	NED(0.50)	0.035	0.132	0.365	0.676	0.887	0.977	0.996
	NED(1.00)	0.052	0.177	0.417	0.719	0.911	0.986	0.998
	NED(2.00)	0.076	0.229	0.462	0.750	0.925	0.988	0.998
	LD(MLE)	0.013	0.059	0.211	0.435	0.644	0.762	0.810
	TIKU	0.014	0.068	0.224	0.501	0.775	0.927	0.980
	t	0.009	0.048	0.171	0.388	0.610	0.739	0.796

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.097	0.259	0.512	0.736	0.871	0.952	0.983
	NED(0.15)	0.090	0.226	0.438	0.643	0.792	0.896	0.949
	NED(0.30)	0.097	0.244	0.476	0.694	0.836	0.930	0.973
	NED(0.50)	0.105	0.269	0.516	0.735	0.873	0.949	0.981
	NED(1.00)	0.127	0.310	0.566	0.789	0.903	0.971	0.988
	NED(2.00)	0.151	0.346	0.607	0.819	0.919	0.977	0.990
	LD(MLE)	0.053	0.162	0.325	0.516	0.671	0.790	0.848
	TIKU	0.057	0.173	0.394	0.625	0.793	0.902	0.946
	t	0.041	0.134	0.286	0.482	0.641	0.767	0.831
$\alpha = 0.01$	HD	0.033	0.111	0.287	0.526	0.718	0.864	0.932
	NED(0.15)	0.026	0.081	0.223	0.422	0.611	0.764	0.859
	NED(0.30)	0.031	0.098	0.256	0.470	0.670	0.824	0.907
	NED(0.50)	0.038	0.121	0.303	0.536	0.729	0.860	0.927
	NED(1.00)	0.056	0.164	0.363	0.617	0.782	0.900	0.956
	NED(2.00)	0.076	0.191	0.417	0.667	0.817	0.925	0.968
	LD(MLE)	0.011	0.042	0.121	0.281	0.454	0.596	0.701
	TIKU	0.008	0.044	0.159	0.349	0.555	0.731	0.833
	t	0.006	0.033	0.098	0.243	0.409	0.551	0.664

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.103	0.258	0.436	0.611	0.768	0.849	0.901
	NED(0.15)	0.104	0.237	0.378	0.541	0.671	0.776	0.847
	NED(0.30)	0.111	0.258	0.416	0.589	0.735	0.819	0.883
	NED(0.50)	0.121	0.282	0.455	0.628	0.779	0.856	0.913
	NED(1.00)	0.131	0.321	0.496	0.683	0.825	0.891	0.931
	NED(2.00)	0.144	0.341	0.525	0.718	0.846	0.910	0.946
	LD(MLE)	0.037	0.095	0.144	0.227	0.307	0.374	0.444
	TIKU	0.051	0.148	0.264	0.421	0.557	0.657	0.750
	t	0.025	0.074	0.119	0.199	0.277	0.344	0.413
$\alpha = 0.01$	HD	0.034	0.109	0.232	0.414	0.597	0.708	0.802
	NED(0.15)	0.030	0.091	0.185	0.345	0.490	0.610	0.713
	NED(0.30)	0.034	0.110	0.218	0.390	0.565	0.673	0.771
	NED(0.50)	0.040	0.132	0.257	0.440	0.622	0.729	0.819
	NED(1.00)	0.054	0.160	0.306	0.508	0.684	0.788	0.855
	NED(2.00)	0.064	0.182	0.343	0.550	0.720	0.816	0.882
	LD(MLE)	0.003	0.015	0.037	0.076	0.141	0.199	0.261
	TIKU	0.005	0.031	0.079	0.181	0.302	0.419	0.539
	t	0.002	0.010	0.030	0.062	0.114	0.171	0.232

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

$Uniform(0, 1)$

	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.078	0.200	0.400	0.671	0.869	0.963	0.995
	NED(0.15)	0.051	0.163	0.364	0.657	0.868	0.970	0.997
	NED(0.30)	0.060	0.173	0.372	0.663	0.867	0.969	0.995
	NED(0.50)	0.074	0.187	0.385	0.668	0.865	0.963	0.994
	NED(1.00)	0.106	0.228	0.411	0.670	0.859	0.951	0.990
	NED(2.00)	0.147	0.265	0.442	0.679	0.850	0.940	0.982
	LD(MLE)	0.059	0.181	0.381	0.670	0.871	0.969	0.996
	TIKU	0.053	0.130	0.271	0.493	0.718	0.884	0.974
	t	0.048	0.154	0.338	0.623	0.844	0.959	0.995
$\alpha = 0.01$	HD	0.032	0.077	0.176	0.368	0.620	0.834	0.956
	NED(0.15)	0.012	0.042	0.131	0.326	0.595	0.829	0.955
	NED(0.30)	0.018	0.051	0.139	0.336	0.603	0.831	0.955
	NED(0.50)	0.028	0.068	0.162	0.355	0.610	0.832	0.955
	NED(1.00)	0.055	0.107	0.208	0.387	0.627	0.832	0.952
	NED(2.00)	0.090	0.150	0.257	0.427	0.652	0.826	0.945
	LD(MLE)	0.015	0.054	0.149	0.348	0.614	0.836	0.958
	TIKU	0.014	0.040	0.090	0.201	0.379	0.596	0.780
	t	0.012	0.039	0.119	0.292	0.552	0.788	0.931

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.092	0.176	0.308	0.434	0.578	0.714	0.810
	NED(0.15)	0.083	0.156	0.266	0.388	0.516	0.641	0.752
	NED(0.30)	0.092	0.177	0.291	0.417	0.558	0.680	0.786
	NED(0.50)	0.100	0.198	0.320	0.445	0.589	0.709	0.811
	NED(1.00)	0.123	0.220	0.359	0.488	0.632	0.752	0.843
	NED(2.00)	0.145	0.244	0.392	0.527	0.662	0.778	0.863
	LD(MLE)	0.062	0.122	0.225	0.328	0.463	0.588	0.710
	TIKU	0.051	0.105	0.192	0.307	0.431	0.564	0.687
	t	0.050	0.100	0.196	0.292	0.424	0.554	0.676
$\alpha = 0.01$	HD	0.029	0.062	0.133	0.229	0.351	0.477	0.618
	NED(0.15)	0.021	0.046	0.105	0.193	0.280	0.396	0.529
	NED(0.30)	0.026	0.058	0.124	0.216	0.322	0.448	0.582
	NED(0.50)	0.034	0.071	0.147	0.242	0.364	0.494	0.624
	NED(1.00)	0.053	0.097	0.186	0.295	0.418	0.548	0.685
	NED(2.00)	0.076	0.124	0.229	0.343	0.472	0.593	0.727
	LD(MLE)	0.013	0.031	0.065	0.132	0.215	0.330	0.446
	TIKU	0.011	0.023	0.052	0.103	0.172	0.265	0.391
	t	0.010	0.023	0.053	0.104	0.174	0.283	0.398

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		Exponential(1)						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.096	0.510	0.850	0.968	0.995	1.000	1.000
	NED(0.15)	0.087	0.418	0.748	0.929	0.984	0.998	1.000
	NED(0.30)	0.097	0.458	0.801	0.951	0.991	0.999	1.000
	NED(0.50)	0.110	0.511	0.842	0.965	0.995	0.999	1.000
	NED(1.00)	0.135	0.585	0.884	0.976	0.995	1.000	1.000
	NED(2.00)	0.155	0.635	0.903	0.976	0.996	1.000	1.000
	LD(MLE)	0.056	0.338	0.692	0.903	0.979	0.993	0.999
	TIKU	0.048	0.330	0.696	0.906	0.975	0.994	0.998
	t	0.044	0.305	0.663	0.886	0.974	0.991	0.998
$\alpha = 0.01$	HD	0.034	0.300	0.690	0.904	0.977	0.996	0.998
	NED(0.15)	0.023	0.214	0.547	0.807	0.935	0.981	0.996
	NED(0.30)	0.028	0.261	0.620	0.862	0.960	0.991	0.997
	NED(0.50)	0.037	0.318	0.691	0.893	0.974	0.996	0.998
	NED(1.00)	0.057	0.399	0.765	0.927	0.984	0.998	1.000
	NED(2.00)	0.073	0.452	0.808	0.942	0.987	0.998	1.000
	LD(MLE)	0.012	0.132	0.453	0.745	0.914	0.969	0.986
	TIKU	0.010	0.115	0.425	0.728	0.899	0.965	0.987
	t	0.008	0.107	0.397	0.704	0.884	0.959	0.984

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.116	0.558	0.820	0.928	0.976	0.990	0.997
	NED(0.15)	0.121	0.448	0.713	0.881	0.951	0.980	0.993
	NED(0.30)	0.139	0.513	0.772	0.912	0.968	0.988	0.996
	NED(0.50)	0.149	0.573	0.814	0.930	0.974	0.991	0.997
	NED(1.00)	0.156	0.637	0.860	0.947	0.982	0.993	0.998
	NED(2.00)	0.153	0.673	0.882	0.956	0.984	0.993	0.998
	LD(MLE)	0.054	0.272	0.526	0.753	0.884	0.939	0.975
	TIKU	0.046	0.285	0.579	0.795	0.903	0.951	0.981
	t	0.043	0.240	0.492	0.729	0.869	0.929	0.968
$\alpha = 0.01$	HD	0.034	0.375	0.697	0.861	0.940	0.972	0.991
	NED(0.15)	0.048	0.263	0.562	0.764	0.889	0.944	0.976
	NED(0.30)	0.045	0.335	0.646	0.824	0.923	0.963	0.987
	NED(0.50)	0.048	0.401	0.710	0.865	0.942	0.975	0.992
	NED(1.00)	0.052	0.480	0.774	0.902	0.960	0.983	0.993
	NED(2.00)	0.055	0.521	0.808	0.923	0.965	0.986	0.994
	LD(MLE)	0.009	0.092	0.294	0.549	0.735	0.843	0.918
	TIKU	0.005	0.091	0.328	0.589	0.762	0.868	0.933
	t	0.005	0.075	0.253	0.503	0.695	0.813	0.900

Table A.1: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 8$. *(continued)*

		$LogNormal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.096	0.429	0.735	0.894	0.957	0.983	0.992
	NED(0.15)	0.095	0.343	0.624	0.816	0.912	0.962	0.981
	NED(0.30)	0.106	0.386	0.685	0.863	0.941	0.976	0.991
	NED(0.50)	0.114	0.438	0.740	0.894	0.959	0.984	0.993
	NED(1.00)	0.132	0.496	0.795	0.919	0.971	0.989	0.996
	NED(2.00)	0.148	0.533	0.821	0.934	0.978	0.990	0.996
	LD(MLE)	0.056	0.198	0.425	0.615	0.755	0.841	0.893
	TIKU	0.041	0.222	0.489	0.707	0.836	0.905	0.952
	t	0.045	0.170	0.392	0.584	0.734	0.820	0.885
$\alpha = 0.01$	HD	0.030	0.240	0.552	0.780	0.893	0.950	0.980
	NED(0.15)	0.022	0.172	0.436	0.647	0.818	0.904	0.953
	NED(0.30)	0.028	0.215	0.502	0.722	0.866	0.937	0.973
	NED(0.50)	0.038	0.262	0.569	0.789	0.899	0.953	0.981
	NED(1.00)	0.051	0.319	0.649	0.843	0.932	0.968	0.988
	NED(2.00)	0.064	0.362	0.698	0.869	0.949	0.974	0.990
	LD(MLE)	0.008	0.066	0.209	0.390	0.562	0.688	0.798
	TIKU	0.007	0.062	0.236	0.461	0.648	0.779	0.869
	t	0.006	0.049	0.174	0.349	0.518	0.654	0.767

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.057	0.502	0.935	1.000	1.000	1.000	1.000
	NED(0.15)	0.050	0.469	0.922	0.998	1.000	1.000	1.000
	NED(0.30)	0.052	0.478	0.930	0.999	1.000	1.000	1.000
	NED(0.50)	0.054	0.490	0.933	1.000	1.000	1.000	1.000
	NED(1.00)	0.064	0.512	0.937	1.000	1.000	1.000	1.000
	NED(2.00)	0.088	0.534	0.936	0.999	1.000	1.000	1.000
	LD(MLE)	0.052	0.484	0.931	1.000	1.000	1.000	1.000
	TIKU	0.049	0.455	0.916	0.999	1.000	1.000	1.000
	t	0.048	0.466	0.925	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.014	0.252	0.804	0.992	1.000	1.000	1.000
	NED(0.15)	0.012	0.210	0.760	0.986	1.000	1.000	1.000
	NED(0.30)	0.012	0.220	0.773	0.991	1.000	1.000	1.000
	NED(0.50)	0.013	0.235	0.790	0.991	1.000	1.000	1.000
	NED(1.00)	0.019	0.268	0.806	0.992	1.000	1.000	1.000
	NED(2.00)	0.028	0.306	0.818	0.992	1.000	1.000	1.000
	LD(MLE)	0.012	0.224	0.786	0.991	1.000	1.000	1.000
	TIKU	0.010	0.203	0.737	0.984	1.000	1.000	1.000
	t	0.011	0.208	0.765	0.989	1.000	1.000	1.000

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		$0.9N(0,1) + 0.1t(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.065	0.482	0.919	0.998	1.000	1.000	1.000
	NED(0.15)	0.061	0.437	0.884	0.990	0.999	1.000	1.000
	NED(0.30)	0.058	0.454	0.905	0.996	1.000	1.000	1.000
	NED(0.50)	0.062	0.471	0.916	0.998	1.000	1.000	1.000
	NED(1.00)	0.074	0.498	0.923	0.998	1.000	1.000	1.000
	NED(2.00)	0.093	0.522	0.923	0.997	1.000	1.000	1.000
	LD(MLE)	0.042	0.335	0.704	0.850	0.890	0.911	0.926
	TIKU	0.051	0.424	0.889	0.997	1.000	1.000	1.000
	t	0.037	0.321	0.693	0.845	0.888	0.907	0.924
$\alpha = 0.01$	HD	0.014	0.221	0.776	0.987	1.000	1.000	1.000
	NED(0.15)	0.012	0.191	0.712	0.958	0.996	1.000	1.000
	NED(0.30)	0.012	0.200	0.739	0.978	1.000	1.000	1.000
	NED(0.50)	0.014	0.211	0.763	0.984	1.000	1.000	1.000
	NED(1.00)	0.020	0.245	0.785	0.988	1.000	1.000	1.000
	NED(2.00)	0.029	0.286	0.801	0.987	1.000	1.000	1.000
	LD(MLE)	0.007	0.131	0.508	0.756	0.838	0.867	0.890
	TIKU	0.009	0.173	0.697	0.971	0.999	1.000	1.000
	t	0.005	0.121	0.486	0.744	0.835	0.863	0.888

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.064	0.462	0.915	0.998	1.000	1.000	1.000
	NED(0.15)	0.059	0.423	0.870	0.987	1.000	1.000	1.000
	NED(0.30)	0.058	0.433	0.897	0.995	1.000	1.000	1.000
	NED(0.50)	0.063	0.452	0.910	0.996	1.000	1.000	1.000
	NED(1.00)	0.074	0.476	0.919	0.998	1.000	1.000	1.000
	NED(2.00)	0.090	0.499	0.922	0.998	1.000	1.000	1.000
	LD(MLE)	0.041	0.296	0.650	0.798	0.858	0.897	0.912
	TIKU	0.052	0.405	0.880	0.995	1.000	1.000	1.000
	t	0.037	0.280	0.636	0.792	0.854	0.893	0.910
$\alpha = 0.01$	HD	0.015	0.227	0.759	0.979	0.999	1.000	1.000
	NED(0.15)	0.012	0.187	0.687	0.946	0.995	1.000	1.000
	NED(0.30)	0.012	0.197	0.719	0.968	0.999	1.000	1.000
	NED(0.50)	0.014	0.216	0.748	0.976	0.999	1.000	1.000
	NED(1.00)	0.021	0.249	0.772	0.980	1.000	1.000	1.000
	NED(2.00)	0.033	0.288	0.787	0.981	0.999	1.000	1.000
	LD(MLE)	0.005	0.108	0.440	0.681	0.793	0.848	0.868
	TIKU	0.010	0.171	0.672	0.957	0.998	1.000	1.000
	t	0.004	0.097	0.421	0.674	0.786	0.843	0.864

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.072	0.357	0.742	0.942	0.994	0.999	1.000
	NED(0.15)	0.083	0.303	0.617	0.830	0.955	0.988	0.997
	NED(0.30)	0.075	0.328	0.673	0.895	0.981	0.996	0.999
	NED(0.50)	0.079	0.360	0.723	0.931	0.991	0.999	1.000
	NED(1.00)	0.085	0.402	0.778	0.959	0.997	1.000	1.000
	NED(2.00)	0.097	0.447	0.815	0.970	0.998	1.000	1.000
	LD(MLE)	0.055	0.212	0.478	0.698	0.843	0.906	0.944
	TIKU	0.064	0.315	0.670	0.900	0.978	0.994	0.998
	t	0.051	0.200	0.464	0.688	0.832	0.903	0.941
$\alpha = 0.01$	HD	0.018	0.165	0.514	0.840	0.973	0.996	1.000
	NED(0.15)	0.021	0.131	0.391	0.664	0.867	0.952	0.986
	NED(0.30)	0.021	0.148	0.446	0.750	0.930	0.981	0.998
	NED(0.50)	0.024	0.171	0.505	0.821	0.961	0.993	1.000
	NED(1.00)	0.029	0.211	0.582	0.879	0.980	0.998	1.000
	NED(2.00)	0.035	0.248	0.634	0.909	0.988	0.999	1.000
	LD(MLE)	0.010	0.065	0.238	0.482	0.700	0.808	0.877
	TIKU	0.014	0.125	0.423	0.748	0.927	0.979	0.994
	t	0.008	0.057	0.222	0.464	0.686	0.798	0.872

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.078	0.296	0.585	0.831	0.941	0.980	0.995
	NED(0.15)	0.091	0.263	0.455	0.660	0.800	0.898	0.941
	NED(0.30)	0.091	0.283	0.534	0.752	0.884	0.953	0.980
	NED(0.50)	0.094	0.318	0.586	0.816	0.927	0.976	0.993
	NED(1.00)	0.104	0.359	0.655	0.879	0.964	0.990	0.998
	NED(2.00)	0.108	0.393	0.713	0.916	0.979	0.996	0.999
	LD(MLE)	0.031	0.079	0.142	0.214	0.297	0.372	0.436
	TIKU	0.077	0.230	0.410	0.615	0.754	0.847	0.902
	t	0.026	0.073	0.132	0.202	0.287	0.360	0.427
$\alpha = 0.01$	HD	0.022	0.133	0.374	0.657	0.850	0.939	0.980
	NED(0.15)	0.030	0.109	0.263	0.469	0.651	0.786	0.871
	NED(0.30)	0.028	0.130	0.327	0.573	0.755	0.883	0.949
	NED(0.50)	0.029	0.154	0.388	0.657	0.837	0.931	0.975
	NED(1.00)	0.036	0.192	0.463	0.750	0.903	0.966	0.991
	NED(2.00)	0.041	0.221	0.522	0.815	0.939	0.981	0.995
	LD(MLE)	0.003	0.014	0.036	0.080	0.140	0.207	0.274
	TIKU	0.012	0.077	0.207	0.412	0.600	0.728	0.820
	t	0.003	0.012	0.032	0.074	0.128	0.196	0.263

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

$Uniform(0, 1)$

	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.054	0.295	0.705	0.953	0.996	1.000	1.000
	NED(0.15)	0.047	0.285	0.702	0.954	0.997	1.000	1.000
	NED(0.30)	0.047	0.287	0.702	0.954	0.997	1.000	1.000
	NED(0.50)	0.048	0.291	0.704	0.955	0.997	1.000	1.000
	NED(1.00)	0.054	0.297	0.704	0.954	0.995	1.000	1.000
	NED(2.00)	0.072	0.311	0.702	0.941	0.992	0.999	1.000
	LD(MLE)	0.049	0.293	0.706	0.957	0.997	1.000	1.000
	TIKU	0.039	0.222	0.590	0.891	0.988	1.000	1.000
	t	0.044	0.279	0.691	0.949	0.997	1.000	1.000
$\alpha = 0.01$	HD	0.013	0.112	0.438	0.820	0.976	0.999	1.000
	NED(0.15)	0.010	0.100	0.424	0.815	0.977	0.999	1.000
	NED(0.30)	0.010	0.102	0.425	0.819	0.977	0.999	1.000
	NED(0.50)	0.011	0.106	0.428	0.821	0.977	0.999	1.000
	NED(1.00)	0.013	0.115	0.437	0.822	0.976	0.999	1.000
	NED(2.00)	0.025	0.139	0.451	0.819	0.970	0.998	1.000
	LD(MLE)	0.011	0.107	0.434	0.823	0.977	0.999	1.000
	TIKU	0.007	0.066	0.290	0.662	0.927	0.994	1.000
	t	0.010	0.095	0.404	0.804	0.973	0.999	1.000

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.070	0.199	0.389	0.641	0.832	0.935	0.979
	NED(0.15)	0.076	0.180	0.344	0.552	0.726	0.852	0.931
	NED(0.30)	0.075	0.195	0.362	0.592	0.771	0.895	0.958
	NED(0.50)	0.078	0.209	0.392	0.631	0.812	0.921	0.972
	NED(1.00)	0.087	0.236	0.433	0.684	0.857	0.949	0.984
	NED(2.00)	0.102	0.260	0.468	0.720	0.881	0.961	0.987
	LD(MLE)	0.058	0.158	0.314	0.546	0.743	0.866	0.946
	TIKU	0.057	0.156	0.308	0.546	0.749	0.876	0.951
	t	0.051	0.148	0.296	0.528	0.730	0.857	0.941
$\alpha = 0.01$	HD	0.017	0.073	0.179	0.380	0.623	0.806	0.919
	NED(0.15)	0.018	0.063	0.145	0.313	0.498	0.683	0.811
	NED(0.30)	0.019	0.069	0.164	0.349	0.555	0.746	0.862
	NED(0.50)	0.021	0.080	0.189	0.383	0.606	0.788	0.903
	NED(1.00)	0.025	0.103	0.231	0.448	0.676	0.848	0.938
	NED(2.00)	0.033	0.126	0.264	0.505	0.726	0.883	0.957
	LD(MLE)	0.011	0.039	0.112	0.283	0.492	0.678	0.829
	TIKU	0.012	0.041	0.115	0.282	0.487	0.688	0.836
	t	0.011	0.033	0.103	0.263	0.469	0.657	0.816

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		Exponential(1)						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.076	0.696	0.983	1.000	1.000	1.000	1.000
	NED(0.15)	0.087	0.557	0.931	0.996	1.000	1.000	1.000
	NED(0.30)	0.091	0.630	0.964	0.998	1.000	1.000	1.000
	NED(0.50)	0.092	0.684	0.976	0.999	1.000	1.000	1.000
	NED(1.00)	0.095	0.756	0.989	1.000	1.000	1.000	1.000
	NED(2.00)	0.107	0.800	0.994	1.000	1.000	1.000	1.000
	LD(MLE)	0.055	0.509	0.923	0.996	1.000	1.000	1.000
	TIKU	0.057	0.553	0.943	0.997	1.000	1.000	1.000
	t	0.051	0.494	0.920	0.996	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.023	0.476	0.939	0.998	1.000	1.000	1.000
	NED(0.15)	0.025	0.339	0.812	0.980	0.999	1.000	1.000
	NED(0.30)	0.026	0.402	0.887	0.993	1.000	1.000	1.000
	NED(0.50)	0.029	0.474	0.929	0.997	1.000	1.000	1.000
	NED(1.00)	0.032	0.577	0.958	0.999	1.000	1.000	1.000
	NED(2.00)	0.043	0.644	0.975	1.000	1.000	1.000	1.000
	LD(MLE)	0.009	0.265	0.790	0.977	0.999	1.000	1.000
	TIKU	0.012	0.293	0.836	0.986	0.999	1.000	1.000
	t	0.008	0.246	0.773	0.976	0.999	1.000	1.000

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.090	0.698	0.959	0.997	1.000	1.000	1.000
	NED(0.15)	0.118	0.527	0.863	0.980	0.998	0.999	1.000
	NED(0.30)	0.124	0.631	0.929	0.993	0.999	1.000	1.000
	NED(0.50)	0.133	0.705	0.959	0.997	1.000	1.000	1.000
	NED(1.00)	0.134	0.796	0.980	0.999	1.000	1.000	1.000
	NED(2.00)	0.130	0.849	0.988	1.000	1.000	1.000	1.000
	LD(MLE)	0.050	0.364	0.737	0.938	0.990	0.998	1.000
	TIKU	0.063	0.444	0.828	0.972	0.996	1.000	1.000
	t	0.046	0.349	0.727	0.936	0.989	0.998	0.999
$\alpha = 0.01$	HD	0.024	0.510	0.913	0.988	1.000	1.000	1.000
	NED(0.15)	0.040	0.334	0.747	0.936	0.988	0.998	1.000
	NED(0.30)	0.044	0.441	0.851	0.972	0.997	0.999	1.000
	NED(0.50)	0.046	0.543	0.913	0.986	0.999	1.000	1.000
	NED(1.00)	0.045	0.665	0.954	0.995	1.000	1.000	1.000
	NED(2.00)	0.049	0.741	0.969	0.997	1.000	1.000	1.000
	LD(MLE)	0.007	0.154	0.532	0.828	0.956	0.989	0.997
	TIKU	0.008	0.220	0.661	0.908	0.983	0.997	0.999
	t	0.005	0.140	0.513	0.817	0.951	0.986	0.997

Table A.2: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 20$. *(continued)*

$LogNormal(0, 1)$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.091	0.576	0.915	0.990	0.999	1.000	1.000
	NED(0.15)	0.100	0.434	0.783	0.941	0.986	0.996	1.000
	NED(0.30)	0.105	0.513	0.862	0.976	0.995	0.999	1.000
	NED(0.50)	0.111	0.579	0.906	0.988	0.999	1.000	1.000
	NED(1.00)	0.116	0.664	0.948	0.995	1.000	1.000	1.000
	NED(2.00)	0.118	0.716	0.966	0.999	1.000	1.000	1.000
	LD(MLE)	0.052	0.264	0.552	0.790	0.906	0.954	0.974
	TIKU	0.066	0.368	0.722	0.912	0.976	0.992	0.998
	t	0.047	0.249	0.538	0.781	0.900	0.952	0.973
$\alpha = 0.01$	HD	0.027	0.345	0.811	0.970	0.996	0.999	1.000
	NED(0.15)	0.032	0.244	0.612	0.860	0.957	0.988	0.998
	NED(0.30)	0.033	0.309	0.722	0.928	0.984	0.996	1.000
	NED(0.50)	0.038	0.370	0.799	0.962	0.995	0.999	1.000
	NED(1.00)	0.041	0.459	0.879	0.985	0.999	1.000	1.000
	NED(2.00)	0.048	0.530	0.918	0.991	1.000	1.000	1.000
	LD(MLE)	0.008	0.098	0.329	0.620	0.799	0.895	0.941
	TIKU	0.011	0.152	0.507	0.807	0.931	0.977	0.992
	t	0.006	0.088	0.314	0.604	0.786	0.890	0.936

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.054	0.732	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.052	0.709	0.998	1.000	1.000	1.000	1.000
	NED(0.30)	0.052	0.718	0.999	1.000	1.000	1.000	1.000
	NED(0.50)	0.050	0.722	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.057	0.734	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.072	0.742	0.998	1.000	1.000	1.000	1.000
	LD(MLE)	0.050	0.724	0.999	1.000	1.000	1.000	1.000
	TIKU	0.048	0.708	0.997	1.000	1.000	1.000	1.000
	t	0.049	0.717	0.999	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.013	0.475	0.981	1.000	1.000	1.000	1.000
	NED(0.15)	0.011	0.445	0.974	1.000	1.000	1.000	1.000
	NED(0.30)	0.011	0.451	0.977	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.459	0.980	1.000	1.000	1.000	1.000
	NED(1.00)	0.015	0.477	0.981	1.000	1.000	1.000	1.000
	NED(2.00)	0.020	0.498	0.982	1.000	1.000	1.000	1.000
	LD(MLE)	0.013	0.459	0.980	1.000	1.000	1.000	1.000
	TIKU	0.012	0.432	0.975	1.000	1.000	1.000	1.000
	t	0.012	0.446	0.979	1.000	1.000	1.000	1.000

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$0.9N(0,1) + 0.1t(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.057	0.707	0.997	1.000	1.000	1.000	1.000
	NED(0.15)	0.060	0.662	0.985	1.000	1.000	1.000	1.000
	NED(0.30)	0.052	0.685	0.993	1.000	1.000	1.000	1.000
	NED(0.50)	0.054	0.698	0.996	1.000	1.000	1.000	1.000
	NED(1.00)	0.061	0.714	0.997	1.000	1.000	1.000	1.000
	NED(2.00)	0.071	0.723	0.997	1.000	1.000	1.000	1.000
	LD(MLE)	0.035	0.448	0.779	0.858	0.891	0.915	0.933
	TIKU	0.048	0.668	0.994	1.000	1.000	1.000	1.000
	t	0.034	0.440	0.775	0.857	0.890	0.914	0.931
$\alpha = 0.01$	HD	0.015	0.462	0.974	1.000	1.000	1.000	1.000
	NED(0.15)	0.013	0.410	0.943	0.999	1.000	1.000	1.000
	NED(0.30)	0.013	0.430	0.963	1.000	1.000	1.000	1.000
	NED(0.50)	0.014	0.449	0.971	1.000	1.000	1.000	1.000
	NED(1.00)	0.017	0.475	0.976	1.000	1.000	1.000	1.000
	NED(2.00)	0.023	0.496	0.977	1.000	1.000	1.000	1.000
	LD(MLE)	0.004	0.232	0.651	0.787	0.843	0.881	0.902
	TIKU	0.012	0.403	0.961	1.000	1.000	1.000	1.000
	t	0.004	0.226	0.646	0.784	0.841	0.880	0.900

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.061	0.688	0.995	1.000	1.000	1.000	1.000
	NED(0.15)	0.068	0.632	0.975	0.999	1.000	1.000	1.000
	NED(0.30)	0.061	0.655	0.991	1.000	1.000	1.000	1.000
	NED(0.50)	0.061	0.675	0.994	1.000	1.000	1.000	1.000
	NED(1.00)	0.066	0.698	0.995	1.000	1.000	1.000	1.000
	NED(2.00)	0.074	0.711	0.995	1.000	1.000	1.000	1.000
	LD(MLE)	0.041	0.377	0.720	0.812	0.867	0.899	0.919
	TIKU	0.056	0.642	0.991	1.000	1.000	1.000	1.000
	t	0.039	0.371	0.714	0.810	0.865	0.898	0.917
$\alpha = 0.01$	HD	0.013	0.425	0.966	1.000	1.000	1.000	1.000
	NED(0.15)	0.014	0.383	0.924	0.999	1.000	1.000	1.000
	NED(0.30)	0.012	0.405	0.954	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.415	0.964	1.000	1.000	1.000	1.000
	NED(1.00)	0.015	0.439	0.969	1.000	1.000	1.000	1.000
	NED(2.00)	0.020	0.464	0.973	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.186	0.569	0.733	0.810	0.854	0.875
	TIKU	0.012	0.371	0.952	1.000	1.000	1.000	1.000
	t	0.006	0.180	0.564	0.729	0.808	0.853	0.873

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.060	0.510	0.924	0.998	1.000	1.000	1.000
	NED(0.15)	0.087	0.404	0.770	0.957	0.994	1.000	1.000
	NED(0.30)	0.070	0.454	0.852	0.985	0.999	1.000	1.000
	NED(0.50)	0.069	0.500	0.902	0.995	1.000	1.000	1.000
	NED(1.00)	0.074	0.556	0.941	0.999	1.000	1.000	1.000
	NED(2.00)	0.085	0.595	0.961	1.000	1.000	1.000	1.000
	LD(MLE)	0.048	0.285	0.625	0.837	0.923	0.958	0.973
	TIKU	0.068	0.478	0.889	0.991	0.999	1.000	1.000
	t	0.046	0.277	0.620	0.833	0.922	0.957	0.972
$\alpha = 0.01$	HD	0.014	0.266	0.789	0.983	1.000	1.000	1.000
	NED(0.15)	0.024	0.213	0.586	0.877	0.976	0.998	1.000
	NED(0.30)	0.017	0.237	0.688	0.948	0.996	1.000	1.000
	NED(0.50)	0.016	0.268	0.762	0.974	0.999	1.000	1.000
	NED(1.00)	0.020	0.321	0.829	0.988	1.000	1.000	1.000
	NED(2.00)	0.026	0.369	0.876	0.995	1.000	1.000	1.000
	LD(MLE)	0.008	0.103	0.386	0.683	0.847	0.916	0.949
	TIKU	0.015	0.245	0.735	0.967	0.997	1.000	1.000
	t	0.008	0.099	0.377	0.675	0.843	0.915	0.946

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.073	0.383	0.785	0.957	0.994	0.999	1.000
	NED(0.15)	0.111	0.304	0.566	0.779	0.908	0.960	0.986
	NED(0.30)	0.092	0.342	0.677	0.888	0.970	0.993	0.998
	NED(0.50)	0.095	0.386	0.753	0.937	0.991	0.999	1.000
	NED(1.00)	0.093	0.453	0.841	0.974	0.998	1.000	1.000
	NED(2.00)	0.096	0.505	0.885	0.989	0.999	1.000	1.000
	LD(MLE)	0.032	0.077	0.141	0.222	0.297	0.369	0.439
	TIKU	0.097	0.318	0.610	0.817	0.921	0.965	0.989
	t	0.029	0.073	0.135	0.216	0.291	0.362	0.433
$\alpha = 0.01$	HD	0.020	0.173	0.572	0.882	0.979	0.997	1.000
	NED(0.15)	0.034	0.146	0.364	0.617	0.808	0.907	0.967
	NED(0.30)	0.032	0.168	0.472	0.776	0.923	0.974	0.995
	NED(0.50)	0.029	0.191	0.565	0.856	0.965	0.994	0.999
	NED(1.00)	0.029	0.242	0.673	0.926	0.990	0.999	1.000
	NED(2.00)	0.031	0.294	0.747	0.959	0.996	1.000	1.000
	LD(MLE)	0.003	0.013	0.039	0.085	0.138	0.200	0.263
	TIKU	0.027	0.138	0.399	0.668	0.837	0.924	0.973
	t	0.002	0.012	0.036	0.082	0.135	0.195	0.256

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$Uniform(0,1)$						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.051	0.461	0.931	0.999	1.000	1.000	1.000
	NED(0.15)	0.049	0.455	0.925	0.999	1.000	1.000	1.000
	NED(0.30)	0.049	0.455	0.926	0.999	1.000	1.000	1.000
	NED(0.50)	0.048	0.457	0.928	0.999	1.000	1.000	1.000
	NED(1.00)	0.050	0.460	0.931	0.999	1.000	1.000	1.000
	NED(2.00)	0.059	0.467	0.928	0.998	1.000	1.000	1.000
	LD(MLE)	0.048	0.459	0.929	0.999	1.000	1.000	1.000
	TIKU	0.039	0.357	0.862	0.996	1.000	1.000	1.000
	t	0.047	0.450	0.927	0.999	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.013	0.221	0.785	0.991	1.000	1.000	1.000
	NED(0.15)	0.012	0.216	0.775	0.991	1.000	1.000	1.000
	NED(0.30)	0.012	0.216	0.776	0.990	1.000	1.000	1.000
	NED(0.50)	0.012	0.217	0.779	0.991	1.000	1.000	1.000
	NED(1.00)	0.013	0.219	0.784	0.992	1.000	1.000	1.000
	NED(2.00)	0.016	0.227	0.786	0.990	1.000	1.000	1.000
	LD(MLE)	0.012	0.218	0.783	0.991	1.000	1.000	1.000
	TIKU	0.009	0.135	0.628	0.958	1.000	1.000	1.000
	t	0.011	0.211	0.772	0.991	1.000	1.000	1.000

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.065	0.255	0.587	0.851	0.968	0.996	1.000
	NED(0.15)	0.078	0.227	0.496	0.741	0.908	0.974	0.994
	NED(0.30)	0.071	0.242	0.539	0.794	0.942	0.988	0.998
	NED(0.50)	0.072	0.258	0.573	0.834	0.959	0.993	0.999
	NED(1.00)	0.076	0.279	0.620	0.876	0.973	0.997	1.000
	NED(2.00)	0.084	0.312	0.657	0.898	0.981	0.998	1.000
	LD(MLE)	0.056	0.206	0.501	0.768	0.938	0.986	0.997
	TIKU	0.058	0.213	0.519	0.788	0.944	0.989	0.998
	t	0.054	0.200	0.492	0.761	0.935	0.985	0.997
$\alpha = 0.01$	HD	0.016	0.099	0.344	0.655	0.885	0.977	0.997
	NED(0.15)	0.021	0.086	0.269	0.520	0.770	0.914	0.975
	NED(0.30)	0.017	0.089	0.302	0.580	0.828	0.947	0.989
	NED(0.50)	0.020	0.101	0.338	0.628	0.867	0.968	0.995
	NED(1.00)	0.022	0.120	0.387	0.705	0.905	0.984	0.998
	NED(2.00)	0.027	0.143	0.437	0.754	0.927	0.988	0.999
	LD(MLE)	0.012	0.068	0.253	0.530	0.791	0.939	0.983
	TIKU	0.012	0.071	0.269	0.552	0.814	0.944	0.989
	t	0.011	0.064	0.245	0.519	0.784	0.936	0.982

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		Exponential(1)						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.070	0.885	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.083	0.736	0.994	1.000	1.000	1.000	1.000
	NED(0.30)	0.078	0.816	0.998	1.000	1.000	1.000	1.000
	NED(0.50)	0.086	0.867	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.085	0.918	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.095	0.944	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.055	0.742	0.995	1.000	1.000	1.000	1.000
	TIKU	0.058	0.790	0.998	1.000	1.000	1.000	1.000
	t	0.051	0.733	0.995	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.015	0.729	0.997	1.000	1.000	1.000	1.000
	NED(0.15)	0.023	0.532	0.970	1.000	1.000	1.000	1.000
	NED(0.30)	0.021	0.628	0.989	1.000	1.000	1.000	1.000
	NED(0.50)	0.022	0.711	0.996	1.000	1.000	1.000	1.000
	NED(1.00)	0.024	0.804	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.030	0.855	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.010	0.484	0.976	1.000	1.000	1.000	1.000
	TIKU	0.012	0.574	0.985	1.000	1.000	1.000	1.000
	t	0.009	0.474	0.974	1.000	1.000	1.000	1.000

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.076	0.842	0.997	1.000	1.000	1.000	1.000
	NED(0.15)	0.113	0.647	0.962	0.998	1.000	1.000	1.000
	NED(0.30)	0.113	0.770	0.989	1.000	1.000	1.000	1.000
	NED(0.50)	0.113	0.847	0.997	1.000	1.000	1.000	1.000
	NED(1.00)	0.098	0.929	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.108	0.958	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.048	0.514	0.930	0.995	1.000	1.000	1.000
	TIKU	0.066	0.633	0.972	1.000	1.000	1.000	1.000
	t	0.045	0.505	0.928	0.995	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.019	0.689	0.991	1.000	1.000	1.000	1.000
	NED(0.15)	0.043	0.462	0.905	0.991	1.000	1.000	1.000
	NED(0.30)	0.044	0.605	0.970	0.999	1.000	1.000	1.000
	NED(0.50)	0.036	0.711	0.990	1.000	1.000	1.000	1.000
	NED(1.00)	0.034	0.832	0.997	1.000	1.000	1.000	1.000
	NED(2.00)	0.040	0.898	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.008	0.270	0.802	0.979	1.000	1.000	1.000
	TIKU	0.013	0.407	0.913	0.994	1.000	1.000	1.000
	t	0.007	0.262	0.794	0.978	1.000	1.000	1.000

Table A.3: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 40$. *(continued)*

$LogNormal(0, 1)$

	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.070	0.742	0.993	1.000	1.000	1.000	1.000
	NED(0.15)	0.090	0.550	0.913	0.991	0.999	1.000	1.000
	NED(0.30)	0.085	0.656	0.968	0.999	1.000	1.000	1.000
	NED(0.50)	0.083	0.738	0.989	1.000	1.000	1.000	1.000
	NED(1.00)	0.092	0.829	0.997	1.000	1.000	1.000	1.000
	NED(2.00)	0.102	0.871	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.046	0.353	0.729	0.923	0.974	0.991	0.995
	TIKU	0.065	0.536	0.906	0.991	0.999	1.000	1.000
	t	0.044	0.345	0.724	0.920	0.974	0.990	0.995
$\alpha = 0.01$	HD	0.018	0.535	0.970	0.999	1.000	1.000	1.000
	NED(0.15)	0.028	0.355	0.803	0.971	0.997	0.999	1.000
	NED(0.30)	0.025	0.453	0.916	0.993	1.000	1.000	1.000
	NED(0.50)	0.025	0.535	0.962	0.998	1.000	1.000	1.000
	NED(1.00)	0.029	0.657	0.988	1.000	1.000	1.000	1.000
	NED(2.00)	0.035	0.731	0.994	1.000	1.000	1.000	1.000
	LD(MLE)	0.008	0.149	0.519	0.817	0.932	0.976	0.987
	TIKU	0.013	0.302	0.789	0.966	0.997	1.000	1.000
	t	0.008	0.143	0.509	0.810	0.928	0.975	0.986

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.055	0.803	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.045	0.788	0.999	1.000	1.000	1.000	1.000
	NED(0.30)	0.046	0.790	0.999	1.000	1.000	1.000	1.000
	NED(0.50)	0.048	0.796	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.057	0.802	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.066	0.804	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.048	0.800	0.999	1.000	1.000	1.000	1.000
	TIKU	0.049	0.784	0.999	1.000	1.000	1.000	1.000
	t	0.045	0.794	0.999	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.010	0.570	0.996	1.000	1.000	1.000	1.000
	NED(0.15)	0.009	0.537	0.994	1.000	1.000	1.000	1.000
	NED(0.30)	0.009	0.548	0.996	1.000	1.000	1.000	1.000
	NED(0.50)	0.008	0.558	0.996	1.000	1.000	1.000	1.000
	NED(1.00)	0.010	0.572	0.996	1.000	1.000	1.000	1.000
	NED(2.00)	0.014	0.586	0.995	1.000	1.000	1.000	1.000
	LD(MLE)	0.008	0.559	0.996	1.000	1.000	1.000	1.000
	TIKU	0.009	0.526	0.993	1.000	1.000	1.000	1.000
	t	0.007	0.551	0.996	1.000	1.000	1.000	1.000

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

$0.9N(0,1) + 0.1t(1)$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.054	0.777	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.053	0.718	0.994	1.000	1.000	1.000	1.000
	NED(0.30)	0.051	0.753	0.999	1.000	1.000	1.000	1.000
	NED(0.50)	0.052	0.769	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.058	0.784	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.068	0.791	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.042	0.471	0.769	0.853	0.895	0.910	0.923
	TIKU	0.052	0.745	0.998	1.000	1.000	1.000	1.000
	t	0.040	0.466	0.766	0.852	0.894	0.909	0.922
$\alpha = 0.01$	HD	0.012	0.548	0.990	1.000	1.000	1.000	1.000
	NED(0.15)	0.012	0.488	0.970	1.000	1.000	1.000	1.000
	NED(0.30)	0.010	0.515	0.986	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.534	0.990	1.000	1.000	1.000	1.000
	NED(1.00)	0.014	0.559	0.992	1.000	1.000	1.000	1.000
	NED(2.00)	0.019	0.580	0.992	1.000	1.000	1.000	1.000
	LD(MLE)	0.005	0.263	0.657	0.785	0.845	0.873	0.898
	TIKU	0.009	0.496	0.986	1.000	1.000	1.000	1.000
	t	0.004	0.257	0.653	0.783	0.844	0.872	0.897

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.060	0.763	0.998	1.000	1.000	1.000	1.000
	NED(0.15)	0.070	0.697	0.991	1.000	1.000	1.000	1.000
	NED(0.30)	0.060	0.735	0.997	1.000	1.000	1.000	1.000
	NED(0.50)	0.058	0.753	0.998	1.000	1.000	1.000	1.000
	NED(1.00)	0.061	0.769	0.998	1.000	1.000	1.000	1.000
	NED(2.00)	0.069	0.775	0.998	1.000	1.000	1.000	1.000
	LD(MLE)	0.038	0.405	0.724	0.812	0.869	0.897	0.914
	TIKU	0.057	0.720	0.998	1.000	1.000	1.000	1.000
	t	0.036	0.397	0.720	0.810	0.867	0.895	0.913
$\alpha = 0.01$	HD	0.015	0.527	0.991	1.000	1.000	1.000	1.000
	NED(0.15)	0.016	0.465	0.963	0.999	1.000	1.000	1.000
	NED(0.30)	0.014	0.488	0.984	1.000	1.000	1.000	1.000
	NED(0.50)	0.014	0.511	0.989	1.000	1.000	1.000	1.000
	NED(1.00)	0.016	0.532	0.992	1.000	1.000	1.000	1.000
	NED(2.00)	0.019	0.549	0.992	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.205	0.588	0.729	0.810	0.851	0.877
	TIKU	0.011	0.470	0.985	1.000	1.000	1.000	1.000
	t	0.007	0.201	0.583	0.727	0.807	0.849	0.876

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.059	0.554	0.957	1.000	1.000	1.000	1.000
	NED(0.15)	0.092	0.454	0.827	0.974	0.998	1.000	1.000
	NED(0.30)	0.073	0.502	0.905	0.994	1.000	1.000	1.000
	NED(0.50)	0.072	0.533	0.941	0.999	1.000	1.000	1.000
	NED(1.00)	0.071	0.586	0.969	1.000	1.000	1.000	1.000
	NED(2.00)	0.080	0.634	0.980	1.000	1.000	1.000	1.000
	LD(MLE)	0.053	0.302	0.673	0.870	0.931	0.971	0.979
	TIKU	0.072	0.532	0.936	0.998	1.000	1.000	1.000
	t	0.051	0.294	0.669	0.867	0.931	0.969	0.979
$\alpha = 0.01$	HD	0.013	0.305	0.866	0.994	1.000	1.000	1.000
	NED(0.15)	0.027	0.243	0.675	0.919	0.993	0.999	1.000
	NED(0.30)	0.019	0.269	0.776	0.973	0.999	1.000	1.000
	NED(0.50)	0.019	0.300	0.837	0.988	1.000	1.000	1.000
	NED(1.00)	0.020	0.359	0.897	0.996	1.000	1.000	1.000
	NED(2.00)	0.021	0.406	0.928	0.999	1.000	1.000	1.000
	LD(MLE)	0.011	0.112	0.452	0.749	0.870	0.940	0.956
	TIKU	0.018	0.299	0.826	0.988	1.000	1.000	1.000
	t	0.011	0.107	0.443	0.746	0.867	0.939	0.955

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.069	0.431	0.843	0.982	0.998	1.000	1.000
	NED(0.15)	0.110	0.334	0.608	0.824	0.940	0.983	0.995
	NED(0.30)	0.095	0.375	0.730	0.928	0.989	0.997	1.000
	NED(0.50)	0.087	0.436	0.812	0.967	0.997	0.999	1.000
	NED(1.00)	0.086	0.493	0.888	0.990	0.999	1.000	1.000
	NED(2.00)	0.095	0.552	0.934	0.997	1.000	1.000	1.000
	LD(MLE)	0.035	0.074	0.140	0.232	0.298	0.372	0.440
	TIKU	0.097	0.361	0.672	0.867	0.958	0.981	0.995
	t	0.034	0.072	0.137	0.228	0.291	0.369	0.437
$\alpha = 0.01$	HD	0.020	0.220	0.660	0.934	0.994	0.998	1.000
	NED(0.15)	0.037	0.174	0.406	0.677	0.862	0.943	0.982
	NED(0.30)	0.032	0.198	0.525	0.834	0.961	0.988	0.998
	NED(0.50)	0.027	0.234	0.634	0.910	0.988	0.995	1.000
	NED(1.00)	0.027	0.282	0.741	0.965	0.997	0.999	1.000
	NED(2.00)	0.030	0.347	0.819	0.985	1.000	1.000	1.000
	LD(MLE)	0.003	0.011	0.037	0.089	0.142	0.210	0.273
	TIKU	0.027	0.171	0.476	0.746	0.897	0.959	0.987
	t	0.003	0.010	0.035	0.086	0.139	0.203	0.268

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

		$Uniform(0,1)$						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.052	0.527	0.963	1.000	1.000	1.000	1.000
	NED(0.15)	0.053	0.521	0.961	1.000	1.000	1.000	1.000
	NED(0.30)	0.053	0.521	0.962	1.000	1.000	1.000	1.000
	NED(0.50)	0.052	0.522	0.962	1.000	1.000	1.000	1.000
	NED(1.00)	0.052	0.526	0.963	1.000	1.000	1.000	1.000
	NED(2.00)	0.054	0.530	0.962	1.000	1.000	1.000	1.000
	LD(MLE)	0.052	0.526	0.962	1.000	1.000	1.000	1.000
	TIKU	0.038	0.413	0.922	1.000	1.000	1.000	1.000
	t	0.049	0.517	0.962	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.010	0.265	0.867	0.999	1.000	1.000	1.000
	NED(0.15)	0.009	0.262	0.861	0.998	1.000	1.000	1.000
	NED(0.30)	0.009	0.262	0.863	0.998	1.000	1.000	1.000
	NED(0.50)	0.009	0.264	0.864	0.998	1.000	1.000	1.000
	NED(1.00)	0.010	0.266	0.868	0.999	1.000	1.000	1.000
	NED(2.00)	0.011	0.269	0.869	0.999	1.000	1.000	1.000
	LD(MLE)	0.009	0.264	0.866	0.998	1.000	1.000	1.000
	TIKU	0.006	0.170	0.743	0.988	1.000	1.000	1.000
	t	0.009	0.259	0.861	0.998	1.000	1.000	1.000

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.058	0.278	0.648	0.909	0.989	0.999	1.000
	NED(0.15)	0.071	0.246	0.552	0.805	0.948	0.991	0.998
	NED(0.30)	0.066	0.253	0.592	0.855	0.973	0.996	0.999
	NED(0.50)	0.061	0.270	0.631	0.891	0.985	0.999	1.000
	NED(1.00)	0.066	0.307	0.686	0.922	0.991	1.000	1.000
	NED(2.00)	0.080	0.344	0.727	0.940	0.994	1.000	1.000
	LD(MLE)	0.047	0.230	0.551	0.844	0.972	0.998	1.000
	TIKU	0.051	0.238	0.584	0.861	0.976	0.998	0.999
	t	0.044	0.224	0.546	0.840	0.970	0.997	0.999
$\alpha = 0.01$	HD	0.015	0.112	0.403	0.747	0.949	0.995	0.999
	NED(0.15)	0.017	0.100	0.309	0.605	0.838	0.961	0.991
	NED(0.30)	0.017	0.108	0.348	0.679	0.900	0.982	0.999
	NED(0.50)	0.016	0.114	0.386	0.720	0.930	0.991	0.999
	NED(1.00)	0.020	0.135	0.447	0.788	0.961	0.998	1.000
	NED(2.00)	0.027	0.160	0.503	0.834	0.976	0.999	1.000
	LD(MLE)	0.011	0.077	0.294	0.629	0.882	0.977	0.998
	TIKU	0.012	0.087	0.319	0.659	0.898	0.984	0.998
	t	0.010	0.073	0.285	0.621	0.877	0.976	0.998

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

		Exponential(1)						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.070	0.935	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.093	0.803	0.997	1.000	1.000	1.000	1.000
	NED(0.30)	0.087	0.876	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.083	0.917	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.085	0.955	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.095	0.974	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.060	0.809	0.999	1.000	1.000	1.000	1.000
	TIKU	0.064	0.861	1.000	1.000	1.000	1.000	1.000
	t	0.059	0.806	0.999	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.015	0.819	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.029	0.617	0.987	1.000	1.000	1.000	1.000
	NED(0.30)	0.025	0.724	0.997	1.000	1.000	1.000	1.000
	NED(0.50)	0.022	0.798	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.022	0.876	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.031	0.922	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.011	0.594	0.992	1.000	1.000	1.000	1.000
	TIKU	0.012	0.682	0.997	1.000	1.000	1.000	1.000
	t	0.010	0.586	0.991	1.000	1.000	1.000	1.000

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.076	0.909	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.111	0.712	0.979	1.000	1.000	1.000	1.000
	NED(0.30)	0.108	0.829	0.997	1.000	1.000	1.000	1.000
	NED(0.50)	0.106	0.901	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.094	0.960	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.100	0.976	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.056	0.575	0.964	0.999	1.000	1.000	1.000
	TIKU	0.075	0.704	0.988	1.000	1.000	1.000	1.000
	t	0.053	0.567	0.964	0.999	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.021	0.774	0.998	1.000	1.000	1.000	1.000
	NED(0.15)	0.040	0.530	0.937	0.999	1.000	1.000	1.000
	NED(0.30)	0.038	0.678	0.981	1.000	1.000	1.000	1.000
	NED(0.50)	0.036	0.782	0.996	1.000	1.000	1.000	1.000
	NED(1.00)	0.033	0.897	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.034	0.943	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.014	0.318	0.877	0.993	1.000	1.000	1.000
	TIKU	0.021	0.490	0.956	0.999	1.000	1.000	1.000
	t	0.012	0.310	0.870	0.992	1.000	1.000	1.000

Table A.4: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 50$. *(continued)*

$LogNormal(0, 1)$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.065	0.795	0.996	1.000	1.000	1.000	1.000
	NED(0.15)	0.098	0.597	0.947	0.997	1.000	1.000	1.000
	NED(0.30)	0.087	0.713	0.987	1.000	1.000	1.000	1.000
	NED(0.50)	0.084	0.786	0.995	1.000	1.000	1.000	1.000
	NED(1.00)	0.083	0.868	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.090	0.915	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.051	0.392	0.806	0.945	0.982	0.996	0.999
	TIKU	0.071	0.585	0.953	0.997	1.000	1.000	1.000
	t	0.050	0.385	0.802	0.944	0.981	0.996	0.998
$\alpha = 0.01$	HD	0.015	0.612	0.985	1.000	1.000	1.000	1.000
	NED(0.15)	0.028	0.402	0.874	0.985	0.998	1.000	1.000
	NED(0.30)	0.026	0.516	0.955	0.998	1.000	1.000	1.000
	NED(0.50)	0.022	0.608	0.980	1.000	1.000	1.000	1.000
	NED(1.00)	0.024	0.724	0.995	1.000	1.000	1.000	1.000
	NED(2.00)	0.032	0.801	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.010	0.184	0.616	0.880	0.954	0.987	0.996
	TIKU	0.016	0.364	0.875	0.987	0.999	1.000	1.000
	t	0.010	0.178	0.608	0.877	0.953	0.986	0.996

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.051	0.920	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.046	0.911	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.045	0.916	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.048	0.917	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.052	0.917	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.059	0.918	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.048	0.917	1.000	1.000	1.000	1.000	1.000
	TIKU	0.049	0.903	1.000	1.000	1.000	1.000	1.000
	t	0.048	0.915	1.000	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.010	0.766	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.009	0.745	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.008	0.753	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.009	0.759	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.010	0.767	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.013	0.775	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.009	0.759	1.000	1.000	1.000	1.000	1.000
	TIKU	0.008	0.733	1.000	1.000	1.000	1.000	1.000
	t	0.009	0.753	1.000	1.000	1.000	1.000	1.000

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

		$0.9N(0,1) + 0.1t(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.061	0.906	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.067	0.853	0.999	1.000	1.000	1.000	1.000
	NED(0.30)	0.064	0.890	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.062	0.901	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.063	0.909	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.070	0.913	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.038	0.510	0.771	0.848	0.889	0.910	0.930
	TIKU	0.056	0.890	1.000	1.000	1.000	1.000	1.000
	t	0.037	0.507	0.769	0.846	0.888	0.909	0.929
$\alpha = 0.01$	HD	0.013	0.742	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.014	0.675	0.997	1.000	1.000	1.000	1.000
	NED(0.30)	0.013	0.714	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.731	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.014	0.751	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.016	0.762	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.006	0.312	0.662	0.780	0.836	0.868	0.901
	TIKU	0.012	0.703	1.000	1.000	1.000	1.000	1.000
	t	0.006	0.307	0.659	0.780	0.835	0.867	0.901

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.055	0.889	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.067	0.819	0.999	1.000	1.000	1.000	1.000
	NED(0.30)	0.058	0.864	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.054	0.879	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.057	0.894	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.063	0.899	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.033	0.440	0.732	0.799	0.855	0.886	0.908
	TIKU	0.049	0.865	1.000	1.000	1.000	1.000	1.000
	t	0.031	0.437	0.730	0.798	0.853	0.884	0.908
$\alpha = 0.01$	HD	0.011	0.713	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.016	0.624	0.993	1.000	1.000	1.000	1.000
	NED(0.30)	0.013	0.679	0.999	1.000	1.000	1.000	1.000
	NED(0.50)	0.011	0.699	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.011	0.719	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.014	0.735	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.004	0.245	0.601	0.720	0.801	0.839	0.870
	TIKU	0.010	0.662	0.999	1.000	1.000	1.000	1.000
	t	0.004	0.241	0.598	0.720	0.800	0.837	0.868

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.061	0.682	0.994	1.000	1.000	1.000	1.000
	NED(0.15)	0.092	0.537	0.918	0.996	1.000	1.000	1.000
	NED(0.30)	0.073	0.607	0.972	1.000	1.000	1.000	1.000
	NED(0.50)	0.072	0.659	0.987	1.000	1.000	1.000	1.000
	NED(1.00)	0.072	0.711	0.995	1.000	1.000	1.000	1.000
	NED(2.00)	0.080	0.756	0.997	1.000	1.000	1.000	1.000
	LD(MLE)	0.051	0.376	0.758	0.922	0.965	0.978	0.987
	TIKU	0.073	0.668	0.989	1.000	1.000	1.000	1.000
	t	0.050	0.371	0.754	0.921	0.964	0.978	0.987
$\alpha = 0.01$	HD	0.014	0.442	0.962	1.000	1.000	1.000	1.000
	NED(0.15)	0.028	0.324	0.808	0.982	0.999	1.000	1.000
	NED(0.30)	0.021	0.377	0.903	0.999	1.000	1.000	1.000
	NED(0.50)	0.017	0.423	0.946	1.000	1.000	1.000	1.000
	NED(1.00)	0.019	0.484	0.973	1.000	1.000	1.000	1.000
	NED(2.00)	0.021	0.548	0.985	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.166	0.569	0.843	0.929	0.959	0.975
	TIKU	0.018	0.444	0.949	1.000	1.000	1.000	1.000
	t	0.007	0.164	0.564	0.841	0.928	0.959	0.975

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.069	0.523	0.929	0.997	1.000	1.000	1.000
	NED(0.15)	0.105	0.368	0.678	0.902	0.974	0.996	1.000
	NED(0.30)	0.094	0.440	0.818	0.970	0.997	1.000	1.000
	NED(0.50)	0.085	0.501	0.900	0.992	1.000	1.000	1.000
	NED(1.00)	0.083	0.593	0.951	0.998	1.000	1.000	1.000
	NED(2.00)	0.091	0.661	0.976	0.999	1.000	1.000	1.000
	LD(MLE)	0.030	0.080	0.135	0.224	0.293	0.365	0.435
	TIKU	0.106	0.446	0.793	0.959	0.992	0.999	1.000
	t	0.028	0.079	0.131	0.222	0.291	0.361	0.434
$\alpha = 0.01$	HD	0.018	0.287	0.808	0.983	1.000	1.000	1.000
	NED(0.15)	0.042	0.203	0.492	0.784	0.930	0.981	0.997
	NED(0.30)	0.032	0.242	0.647	0.922	0.989	0.998	1.000
	NED(0.50)	0.026	0.284	0.761	0.970	0.998	1.000	1.000
	NED(1.00)	0.028	0.363	0.872	0.993	1.000	1.000	1.000
	NED(2.00)	0.031	0.433	0.922	0.997	1.000	1.000	1.000
	LD(MLE)	0.002	0.014	0.035	0.085	0.131	0.205	0.268
	TIKU	0.035	0.252	0.628	0.899	0.977	0.996	0.999
	t	0.002	0.013	0.034	0.083	0.130	0.202	0.266

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

$Uniform(0, 1)$

	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.050	0.682	0.994	1.000	1.000	1.000	1.000
	NED(0.15)	0.050	0.670	0.993	1.000	1.000	1.000	1.000
	NED(0.30)	0.050	0.672	0.993	1.000	1.000	1.000	1.000
	NED(0.50)	0.049	0.674	0.994	1.000	1.000	1.000	1.000
	NED(1.00)	0.050	0.680	0.994	1.000	1.000	1.000	1.000
	NED(2.00)	0.054	0.687	0.994	1.000	1.000	1.000	1.000
	LD(MLE)	0.049	0.679	0.994	1.000	1.000	1.000	1.000
	TIKU	0.038	0.566	0.980	1.000	1.000	1.000	1.000
	t	0.048	0.674	0.994	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.009	0.421	0.971	1.000	1.000	1.000	1.000
	NED(0.15)	0.009	0.412	0.968	1.000	1.000	1.000	1.000
	NED(0.30)	0.009	0.414	0.969	1.000	1.000	1.000	1.000
	NED(0.50)	0.008	0.417	0.969	1.000	1.000	1.000	1.000
	NED(1.00)	0.010	0.421	0.971	1.000	1.000	1.000	1.000
	NED(2.00)	0.009	0.428	0.971	1.000	1.000	1.000	1.000
	LD(MLE)	0.010	0.419	0.970	1.000	1.000	1.000	1.000
	TIKU	0.005	0.288	0.923	1.000	1.000	1.000	1.000
	t	0.008	0.414	0.969	1.000	1.000	1.000	1.000

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.059	0.348	0.795	0.978	0.999	1.000	1.000
	NED(0.15)	0.073	0.306	0.684	0.922	0.990	0.999	1.000
	NED(0.30)	0.067	0.321	0.730	0.951	0.996	1.000	1.000
	NED(0.50)	0.061	0.342	0.775	0.969	0.998	1.000	1.000
	NED(1.00)	0.064	0.377	0.821	0.983	0.999	1.000	1.000
	NED(2.00)	0.078	0.409	0.848	0.988	1.000	1.000	1.000
	LD(MLE)	0.051	0.288	0.706	0.949	0.996	1.000	1.000
	TIKU	0.057	0.308	0.736	0.957	0.998	1.000	1.000
	t	0.049	0.282	0.703	0.949	0.996	1.000	1.000
$\alpha = 0.01$	HD	0.015	0.147	0.559	0.902	0.995	1.000	1.000
	NED(0.15)	0.020	0.131	0.441	0.782	0.954	0.992	1.000
	NED(0.30)	0.016	0.135	0.491	0.844	0.981	0.998	1.000
	NED(0.50)	0.016	0.148	0.535	0.883	0.991	1.000	1.000
	NED(1.00)	0.019	0.173	0.603	0.922	0.995	1.000	1.000
	NED(2.00)	0.024	0.210	0.666	0.943	0.997	1.000	1.000
	LD(MLE)	0.011	0.107	0.444	0.821	0.977	0.998	1.000
	TIKU	0.012	0.120	0.482	0.847	0.984	0.999	1.000
	t	0.010	0.104	0.437	0.816	0.977	0.998	1.000

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

		Exponential(1)						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.064	0.985	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.078	0.897	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.071	0.955	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.075	0.978	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.079	0.992	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.084	0.998	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.050	0.916	1.000	1.000	1.000	1.000	1.000
	TIKU	0.058	0.951	1.000	1.000	1.000	1.000	1.000
	t	0.049	0.914	1.000	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.013	0.930	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.022	0.761	0.998	1.000	1.000	1.000	1.000
	NED(0.30)	0.018	0.856	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.018	0.917	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.018	0.965	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.022	0.981	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.771	1.000	1.000	1.000	1.000	1.000
	TIKU	0.012	0.841	1.000	1.000	1.000	1.000	1.000
	t	0.007	0.765	1.000	1.000	1.000	1.000	1.000

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.063	0.965	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.113	0.812	0.997	1.000	1.000	1.000	1.000
	NED(0.30)	0.103	0.909	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.095	0.962	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.081	0.991	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.096	0.996	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.051	0.712	0.993	1.000	1.000	1.000	1.000
	TIKU	0.072	0.847	0.999	1.000	1.000	1.000	1.000
	t	0.050	0.709	0.993	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.018	0.892	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.042	0.650	0.989	1.000	1.000	1.000	1.000
	NED(0.30)	0.038	0.800	0.999	1.000	1.000	1.000	1.000
	NED(0.50)	0.034	0.896	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.030	0.968	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.035	0.985	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.011	0.461	0.967	1.000	1.000	1.000	1.000
	TIKU	0.019	0.665	0.993	1.000	1.000	1.000	1.000
	t	0.010	0.455	0.965	1.000	1.000	1.000	1.000

Table A.5: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 75$. *(continued)*

$LogNormal(0, 1)$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.064	0.912	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.098	0.702	0.981	1.000	1.000	1.000	1.000
	NED(0.30)	0.085	0.829	0.997	1.000	1.000	1.000	1.000
	NED(0.50)	0.080	0.898	0.999	1.000	1.000	1.000	1.000
	NED(1.00)	0.085	0.950	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.097	0.975	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.054	0.478	0.882	0.980	0.998	0.999	0.999
	TIKU	0.076	0.730	0.988	1.000	1.000	1.000	1.000
	t	0.053	0.473	0.879	0.980	0.998	0.999	0.999
$\alpha = 0.01$	HD	0.018	0.770	0.999	1.000	1.000	1.000	1.000
	NED(0.15)	0.031	0.499	0.946	0.999	1.000	1.000	1.000
	NED(0.30)	0.024	0.651	0.990	1.000	1.000	1.000	1.000
	NED(0.50)	0.024	0.755	0.997	1.000	1.000	1.000	1.000
	NED(1.00)	0.024	0.867	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.032	0.917	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.011	0.240	0.737	0.945	0.989	0.996	0.998
	TIKU	0.020	0.510	0.962	0.999	1.000	1.000	1.000
	t	0.011	0.235	0.732	0.943	0.989	0.996	0.998

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$.

		$Normal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.053	0.975	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.048	0.971	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.048	0.972	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.050	0.973	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.052	0.975	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.057	0.975	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.050	0.975	1.000	1.000	1.000	1.000	1.000
	TIKU	0.052	0.966	1.000	1.000	1.000	1.000	1.000
	t	0.048	0.974	1.000	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.012	0.889	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.011	0.870	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.011	0.877	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.881	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.012	0.887	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.013	0.891	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.012	0.883	1.000	1.000	1.000	1.000	1.000
	TIKU	0.011	0.871	1.000	1.000	1.000	1.000	1.000
	t	0.011	0.880	1.000	1.000	1.000	1.000	1.000

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$0.9N(0,1) + 0.1t(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.055	0.959	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.066	0.922	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.055	0.944	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.054	0.954	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.056	0.959	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.059	0.960	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.033	0.547	0.781	0.851	0.898	0.920	0.928
	TIKU	0.051	0.948	1.000	1.000	1.000	1.000	1.000
	t	0.033	0.544	0.780	0.851	0.898	0.919	0.928
$\alpha = 0.01$	HD	0.014	0.850	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.018	0.793	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.013	0.827	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.014	0.840	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.014	0.855	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.017	0.863	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.004	0.358	0.688	0.794	0.851	0.885	0.897
	TIKU	0.012	0.827	1.000	1.000	1.000	1.000	1.000
	t	0.004	0.356	0.687	0.793	0.850	0.884	0.896

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.060	0.956	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.067	0.903	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.063	0.934	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.058	0.949	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.062	0.959	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.067	0.960	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.039	0.482	0.744	0.829	0.868	0.892	0.911
	TIKU	0.056	0.943	1.000	1.000	1.000	1.000	1.000
	t	0.038	0.479	0.742	0.828	0.867	0.892	0.911
$\alpha = 0.01$	HD	0.013	0.837	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.015	0.757	0.999	1.000	1.000	1.000	1.000
	NED(0.30)	0.013	0.808	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.830	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.014	0.845	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.016	0.856	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.004	0.289	0.636	0.751	0.814	0.844	0.871
	TIKU	0.011	0.805	1.000	1.000	1.000	1.000	1.000
	t	0.004	0.287	0.634	0.749	0.813	0.844	0.870

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$t(2)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.058	0.784	0.998	1.000	1.000	1.000	1.000
	NED(0.15)	0.087	0.609	0.959	0.999	1.000	1.000	1.000
	NED(0.30)	0.074	0.694	0.993	1.000	1.000	1.000	1.000
	NED(0.50)	0.067	0.751	0.997	1.000	1.000	1.000	1.000
	NED(1.00)	0.066	0.806	0.999	1.000	1.000	1.000	1.000
	NED(2.00)	0.074	0.848	0.999	1.000	1.000	1.000	1.000
	LD(MLE)	0.046	0.432	0.827	0.942	0.975	0.985	0.992
	TIKU	0.071	0.763	0.997	1.000	1.000	1.000	1.000
	t	0.045	0.429	0.825	0.941	0.975	0.985	0.992
$\alpha = 0.01$	HD	0.015	0.545	0.988	1.000	1.000	1.000	1.000
	NED(0.15)	0.027	0.390	0.893	0.996	1.000	1.000	1.000
	NED(0.30)	0.018	0.460	0.957	1.000	1.000	1.000	1.000
	NED(0.50)	0.018	0.519	0.981	1.000	1.000	1.000	1.000
	NED(1.00)	0.018	0.589	0.993	1.000	1.000	1.000	1.000
	NED(2.00)	0.020	0.657	0.997	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.204	0.666	0.880	0.953	0.974	0.986
	TIKU	0.019	0.538	0.988	1.000	1.000	1.000	1.000
	t	0.007	0.201	0.663	0.880	0.952	0.974	0.986

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

Cauchy

	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.064	0.598	0.973	1.000	1.000	1.000	1.000
	NED(0.15)	0.105	0.404	0.758	0.943	0.990	0.999	1.000
	NED(0.30)	0.086	0.494	0.886	0.989	0.999	1.000	1.000
	NED(0.50)	0.076	0.566	0.949	0.997	1.000	1.000	1.000
	NED(1.00)	0.078	0.656	0.986	0.999	1.000	1.000	1.000
	NED(2.00)	0.081	0.730	0.994	1.000	1.000	1.000	1.000
	LD(MLE)	0.035	0.072	0.134	0.221	0.305	0.375	0.446
	TIKU	0.105	0.494	0.857	0.979	0.998	1.000	1.000
	t	0.034	0.071	0.133	0.219	0.302	0.372	0.443
$\alpha = 0.01$	HD	0.016	0.346	0.893	0.996	1.000	1.000	1.000
	NED(0.15)	0.039	0.236	0.593	0.864	0.971	0.995	0.999
	NED(0.30)	0.025	0.288	0.749	0.964	0.997	1.000	1.000
	NED(0.50)	0.022	0.340	0.852	0.990	0.999	1.000	1.000
	NED(1.00)	0.021	0.429	0.937	0.999	1.000	1.000	1.000
	NED(2.00)	0.025	0.509	0.973	1.000	1.000	1.000	1.000
	LD(MLE)	0.001	0.012	0.039	0.083	0.144	0.208	0.276
	TIKU	0.035	0.293	0.726	0.943	0.992	0.999	1.000
	t	0.001	0.011	0.038	0.083	0.142	0.206	0.274

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$Uniform(0,1)$						
	d	0.0	0.1	0.2	0.3	0.4	0.5	0.6
$\alpha = 0.05$	HD	0.049	0.800	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.051	0.791	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.051	0.791	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.051	0.793	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.050	0.797	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.050	0.806	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.051	0.795	1.000	1.000	1.000	1.000	1.000
	TIKU	0.037	0.694	0.999	1.000	1.000	1.000	1.000
	t	0.050	0.792	1.000	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.012	0.549	0.997	1.000	1.000	1.000	1.000
	NED(0.15)	0.012	0.541	0.996	1.000	1.000	1.000	1.000
	NED(0.30)	0.012	0.543	0.996	1.000	1.000	1.000	1.000
	NED(0.50)	0.012	0.546	0.996	1.000	1.000	1.000	1.000
	NED(1.00)	0.012	0.550	0.997	1.000	1.000	1.000	1.000
	NED(2.00)	0.011	0.555	0.998	1.000	1.000	1.000	1.000
	LD(MLE)	0.012	0.548	0.996	1.000	1.000	1.000	1.000
	TIKU	0.007	0.399	0.981	1.000	1.000	1.000	1.000
	t	0.011	0.544	0.996	1.000	1.000	1.000	1.000

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$\chi^2(4)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.051	0.430	0.881	0.993	1.000	1.000	1.000
	NED(0.15)	0.067	0.369	0.773	0.963	0.997	1.000	1.000
	NED(0.30)	0.058	0.391	0.825	0.984	0.999	1.000	1.000
	NED(0.50)	0.055	0.417	0.862	0.990	0.999	1.000	1.000
	NED(1.00)	0.058	0.456	0.898	0.995	1.000	1.000	1.000
	NED(2.00)	0.067	0.488	0.919	0.997	1.000	1.000	1.000
	LD(MLE)	0.047	0.368	0.808	0.983	1.000	1.000	1.000
	TIKU	0.050	0.386	0.835	0.986	1.000	1.000	1.000
	t	0.046	0.363	0.805	0.982	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.013	0.196	0.686	0.963	0.999	1.000	1.000
	NED(0.15)	0.015	0.167	0.547	0.886	0.985	0.999	1.000
	NED(0.30)	0.013	0.179	0.618	0.931	0.995	1.000	1.000
	NED(0.50)	0.013	0.197	0.666	0.953	0.998	1.000	1.000
	NED(1.00)	0.016	0.223	0.723	0.972	0.999	1.000	1.000
	NED(2.00)	0.020	0.262	0.778	0.983	1.000	1.000	1.000
	LD(MLE)	0.009	0.151	0.575	0.916	0.994	1.000	1.000
	TIKU	0.011	0.167	0.614	0.936	0.996	1.000	1.000
	t	0.008	0.148	0.570	0.916	0.994	1.000	1.000

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		Exponential(1)						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.054	0.995	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.084	0.951	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.071	0.984	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.066	0.993	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.066	0.999	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.079	0.999	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.050	0.970	1.000	1.000	1.000	1.000	1.000
	TIKU	0.061	0.982	1.000	1.000	1.000	1.000	1.000
	t	0.049	0.970	1.000	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.012	0.974	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.024	0.858	1.000	1.000	1.000	1.000	1.000
	NED(0.30)	0.018	0.931	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.015	0.965	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.018	0.987	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.025	0.995	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.009	0.877	1.000	1.000	1.000	1.000	1.000
	TIKU	0.012	0.930	1.000	1.000	1.000	1.000	1.000
	t	0.008	0.876	1.000	1.000	1.000	1.000	1.000

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$\chi^2(1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.060	0.988	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.112	0.872	0.999	1.000	1.000	1.000	1.000
	NED(0.30)	0.095	0.954	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.087	0.987	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.080	0.999	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.098	0.999	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.047	0.798	0.998	1.000	1.000	1.000	1.000
	TIKU	0.071	0.907	1.000	1.000	1.000	1.000	1.000
	t	0.047	0.795	0.998	1.000	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.014	0.950	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.042	0.731	0.997	1.000	1.000	1.000	1.000
	NED(0.30)	0.033	0.882	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.028	0.951	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.025	0.990	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.029	0.997	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.007	0.567	0.991	1.000	1.000	1.000	1.000
	TIKU	0.016	0.771	0.999	1.000	1.000	1.000	1.000
	t	0.007	0.562	0.990	1.000	1.000	1.000	1.000

Table A.6: Empirical levels and powers: Tests of two population means with equal variances for $n_1 = n_2 = 100$. *(continued)*

		$LogNormal(0,1)$						
	d	0.0	0.5	1.0	1.5	2.0	2.5	3.0
$\alpha = 0.05$	HD	0.067	0.956	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.095	0.774	0.996	1.000	1.000	1.000	1.000
	NED(0.30)	0.083	0.890	1.000	1.000	1.000	1.000	1.000
	NED(0.50)	0.080	0.943	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.079	0.980	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.091	0.992	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.054	0.536	0.933	0.994	1.000	1.000	1.000
	TIKU	0.077	0.802	0.998	1.000	1.000	1.000	1.000
	t	0.053	0.532	0.933	0.994	1.000	1.000	1.000
$\alpha = 0.01$	HD	0.015	0.863	1.000	1.000	1.000	1.000	1.000
	NED(0.15)	0.027	0.585	0.978	0.999	1.000	1.000	1.000
	NED(0.30)	0.024	0.745	0.997	1.000	1.000	1.000	1.000
	NED(0.50)	0.019	0.849	1.000	1.000	1.000	1.000	1.000
	NED(1.00)	0.021	0.934	1.000	1.000	1.000	1.000	1.000
	NED(2.00)	0.029	0.967	1.000	1.000	1.000	1.000	1.000
	LD(MLE)	0.008	0.302	0.821	0.980	0.997	0.999	1.000
	TIKU	0.018	0.608	0.986	1.000	1.000	1.000	1.000
	t	0.007	0.296	0.820	0.979	0.997	0.999	1.000

Table A.7: Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 8$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$.

k=0.9 if $n_i < 30$ $i = 1, 2$		$\alpha = 0.05$			$\alpha = 0.01$		
($n_1 = n_2$=8)	d	0.0	1.0	2.5	0.0	1.0	2.5
$N(0, 1)$	HD	0.053	0.596	1.000	0.013	0.303	0.985
	NED(0.15)	0.050	0.590	0.999	0.012	0.293	0.984
	NED(0.30)	0.052	0.591	0.999	0.014	0.298	0.985
	NED(0.50)	0.056	0.593	1.000	0.014	0.304	0.985
	NED(1.00)	0.064	0.603	1.000	0.018	0.318	0.985
	NED(2.00)	0.070	0.613	1.000	0.023	0.334	0.985
	LD(MLE)	0.058	0.632	1.000	0.014	0.345	0.989
	TIKU	0.055	0.564	0.998	0.012	0.264	0.967
	t	0.049	0.590	1.000	0.011	0.289	0.984
$t(2)$	HD	0.058	0.398	0.907	0.013	0.176	0.747
	NED(0.15)	0.062	0.390	0.885	0.013	0.177	0.725
	NED(0.30)	0.065	0.403	0.899	0.015	0.187	0.743
	NED(0.50)	0.067	0.413	0.911	0.016	0.197	0.762
	NED(1.00)	0.072	0.442	0.924	0.019	0.219	0.794
	NED(2.00)	0.077	0.463	0.937	0.022	0.234	0.817
	LD(MLE)	0.053	0.329	0.794	0.010	0.126	0.600
	TIKU	0.054	0.397	0.906	0.009	0.159	0.726
	t	0.042	0.291	0.768	0.006	0.102	0.555

Table A.7: Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 8$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. *(continued)*

		$\alpha = 0.05$			$\alpha = 0.01$		
	d	0.0	1.0	2.5	0.0	1.0	2.5
$0.9N(0,1)+$	HD	0.056	0.560	0.990	0.012	0.282	0.948
$0.1N(0,1)/U(0,1)$	NED(0.15)	0.052	0.554	0.986	0.011	0.271	0.942
	NED(0.30)	0.056	0.558	0.988	0.012	0.277	0.946
	NED(0.50)	0.059	0.566	0.990	0.014	0.287	0.951
	NED(1.00)	0.064	0.578	0.993	0.017	0.308	0.957
	NED(2.00)	0.071	0.589	0.995	0.022	0.324	0.962
	LD(MLE)	0.052	0.500	0.885	0.011	0.251	0.812
	TIKU	0.053	0.515	0.986	0.011	0.244	0.922
	t	0.043	0.463	0.875	0.006	0.210	0.794
$0.9N(0,1) + 0.1t(1)$	HD	0.057	0.338	0.779	0.014	0.142	0.593
	NED(0.15)	0.066	0.341	0.757	0.015	0.148	0.578
	NED(0.30)	0.069	0.353	0.777	0.017	0.158	0.604
	NED(0.50)	0.069	0.367	0.801	0.018	0.170	0.633
	NED(1.00)	0.070	0.392	0.833	0.019	0.190	0.671
	NED(2.00)	0.075	0.410	0.850	0.022	0.204	0.700
	LD(MLE)	0.040	0.167	0.422	0.005	0.048	0.235
	TIKU	0.097	0.292	0.713	0.007	0.090	0.494
	t	0.029	0.140	0.395	0.003	0.037	0.208

Table A.8: Empirical levels and powers: Tests for equality of two population means for $n_1 = 20, n_2 = 20$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$.

($n_1 = n_2$=20)	d	0.0	1.0	2.5	0.0	1.0	2.5
$N(0,1)$	HD	0.049	0.929	1.000	0.009	0.929	1.000
	NED(0.15)	0.046	0.924	1.000	0.008	0.924	1.000
	NED(0.30)	0.047	0.925	1.000	0.008	0.925	1.000
	NED(0.50)	0.048	0.927	1.000	0.009	0.927	1.000
	NED(1.00)	0.050	0.929	1.000	0.010	0.929	1.000
	NED(2.00)	0.054	0.930	1.000	0.011	0.930	1.000
	LD(MLE)	0.052	0.933	1.000	0.009	0.933	1.000
	TIKU	0.047	0.919	1.000	0.010	0.919	1.000
	t	0.047	0.928	1.000	0.008	0.928	1.000
$t(2)$	HD	0.055	0.659	0.997	0.011	0.400	0.981
	NED(0.15)	0.063	0.602	0.986	0.014	0.361	0.951
	NED(0.30)	0.061	0.629	0.992	0.013	0.390	0.967
	NED(0.50)	0.063	0.660	0.996	0.013	0.420	0.980
	NED(1.00)	0.066	0.692	0.998	0.013	0.455	0.989
	NED(2.00)	0.066	0.723	0.999	0.014	0.484	0.991
	LD(MLE)	0.052	0.477	0.907	0.008	0.252	0.818
	TIKU	0.063	0.676	0.994	0.013	0.434	0.976
	t	0.047	0.462	0.903	0.006	0.234	0.807

Table A.8: Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 20$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. *(continued)*

		$\alpha = 0.05$			$\alpha = 0.01$		
	d	0.0	1.0	2.5	0.0	1.0	2.5
$0.9N(0,1)+$	HD	0.055	0.878	1.000	0.009	0.672	1.000
$0.1N(0,1)/U(0,1)$	NED(0.15)	0.059	0.850	1.000	0.011	0.644	0.999
	NED(0.30)	0.057	0.868	1.000	0.011	0.661	1.000
	NED(0.50)	0.056	0.878	1.000	0.010	0.673	1.000
	NED(1.00)	0.058	0.892	1.000	0.011	0.693	1.000
	NED(2.00)	0.061	0.899	1.000	0.011	0.705	1.000
	LD(MLE)	0.046	0.658	0.894	0.006	0.465	0.839
	TIKU	0.055	0.868	1.000	0.014	0.651	0.999
	t	0.041	0.648	0.889	0.006	0.443	0.834
$0.9N(0,1) + 0.1t(1)$	HD	0.054	0.508	0.959	0.010	0.274	0.891
	NED(0.15)	0.069	0.451	0.905	0.015	0.245	0.797
	NED(0.30)	0.065	0.496	0.940	0.017	0.279	0.863
	NED(0.50)	0.066	0.537	0.962	0.016	0.304	0.901
	NED(1.00)	0.066	0.585	0.979	0.017	0.347	0.938
	NED(2.00)	0.067	0.618	0.987	0.018	0.387	0.956
	LD(MLE)	0.031	0.157	0.408	0.003	0.046	0.246
	TIKU	0.074	0.472	0.877	0.010	0.248	0.777
	t	0.027	0.147	0.395	0.002	0.040	0.233

Table A.9: Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 40$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$.

$k = 0.9$ if $n_i < 30$ $i = 1, 2$		$\alpha = 0.05$			$\alpha = 0.01$		
$(n1 = n2 = 40)$	d	0.0	1.0	2.5	0.0	1.0	2.5
$N(0, 1)$	HD	0.051	0.997	1.000	0.012	0.981	1.000
	NED(0.15)	0.049	0.997	1.000	0.011	0.977	1.000
	NED(0.30)	0.048	0.997	1.000	0.011	0.979	1.000
	NED(0.50)	0.049	0.997	1.000	0.012	0.980	1.000
	NED(1.00)	0.052	0.997	1.000	0.013	0.981	1.000
	NED(2.00)	0.059	0.997	1.000	0.014	0.980	1.000
	LD(MLE)	0.050	0.997	1.000	0.013	0.982	1.000
	TIKU	0.048	0.997	1.000	0.012	0.977	1.000
	t	0.049	0.997	1.000	0.012	0.980	1.000
$t(2)$	HD	0.049	0.900	1.000	0.010	0.736	1.000
	NED(0.15)	0.065	0.809	1.000	0.012	0.606	0.998
	NED(0.30)	0.062	0.859	1.000	0.011	0.680	1.000
	NED(0.50)	0.057	0.892	1.000	0.014	0.738	1.000
	NED(1.00)	0.058	0.926	1.000	0.014	0.798	1.000
	NED(2.00)	0.062	0.946	1.000	0.015	0.837	1.000
	LD(MLE)	0.050	0.617	0.958	0.007	0.388	0.914
	TIKU	0.066	0.890	1.000	0.014	0.731	1.000
	t	0.047	0.610	0.957	0.006	0.379	0.912

Table A.9: Empirical levels and powers: Tests for equality of two population means for $n_1 = n_2 = 40$, with square root weights and k=0.9 if $n_i < 30$for $i = 1, 2$. *(continued)*

		$\alpha = 0.05$			$\alpha = 0.01$		
	d	0.0	1.0	2.5	0.0	1.0	2.5
$0.9N(0,1)+$	HD	0.058	0.992	1.000	0.012	0.961	1.000
$0.1N(0,1)/U(0,1)$	NED(0.15)	0.054	0.984	1.000	0.012	0.937	1.000
	NED(0.30)	0.054	0.990	1.000	0.013	0.955	1.000
	NED(0.50)	0.057	0.991	1.000	0.013	0.960	1.000
	NED(1.00)	0.063	0.994	1.000	0.014	0.967	1.000
	NED(2.00)	0.070	0.994	1.000	0.015	0.970	1.000
	LD(MLE)	0.042	0.721	0.894	0.006	0.582	0.847
	TIKU	0.058	0.991	1.000	0.012	0.949	1.000
	t	0.040	0.717	0.893	0.006	0.577	0.846
$0.9N(0,1) + 0.1t(1)$	HD	0.055	0.798	1.000	0.013	0.578	0.998
	NED(0.15)	0.076	0.631	0.987	0.020	0.425	0.959
	NED(0.30)	0.070	0.733	0.998	0.018	0.519	0.989
	NED(0.50)	0.070	0.803	1.000	0.017	0.607	0.996
	NED(1.00)	0.074	0.862	1.000	0.019	0.698	0.999
	NED(2.00)	0.075	0.900	1.000	0.020	0.758	1.000
	LD(MLE)	0.028	0.153	0.415	0.002	0.045	0.244
	TIKU	0.097	0.670	0.977	0.021	0.468	0.952
	t	0.027	0.148	0.409	0.002	0.043	0.237

Table A.10: Empirical levels and powers: Tests for equality of two population means for $n_1 = 8, n_2 = 20$, with square root weight and k=0.9 if $n_i < 30$ for $i = 1, 2$.

$k = 0.9$ if $n_i < 30$ $i = 1, 2$		$\alpha = 0.05$			$\alpha = 0.01$		
$(n_1 = 8, n_2 = 20)$	d	0.0	1.0	2.5	0.0	1.0	2.5
$N(0,1)$	HD	0.050	0.757	1.000	0.010	0.476	1.000
	NED(0.15)	0.050	0.750	1.000	0.010	0.465	1.000
	NED(0.30)	0.050	0.752	1.000	0.010	0.467	1.000
	NED(0.50)	0.050	0.754	1.000	0.010	0.471	1.000
	NED(1.00)	0.052	0.758	1.000	0.012	0.481	1.000
	NED(2.00)	0.056	0.758	1.000	0.014	0.491	1.000
	LD(MLE)	0.057	0.776	1.000	0.012	0.501	1.000
	TIKU	0.052	0.732	1.000	0.011	0.443	0.998
	t	0.050	0.754	1.000	0.009	0.466	1.000
$t(2)$	HD	0.052	0.464	0.968	0.012	0.225	0.890
	NED(0.15)	0.057	0.442	0.945	0.012	0.215	0.846
	NED(0.30)	0.059	0.462	0.958	0.012	0.229	0.868
	NED(0.50)	0.061	0.479	0.967	0.014	0.243	0.892
	NED(1.00)	0.062	0.505	0.978	0.015	0.266	0.918
	NED(2.00)	0.067	0.525	0.983	0.017	0.290	0.934
	LD(MLE)	0.056	0.356	0.837	0.011	0.154	0.681
	TIKU	0.060	0.481	0.964	0.014	0.239	0.888
	t	0.051	0.336	0.825	0.008	0.137	0.659

Table A.10: Empirical levels and powers: Tests for equality of two population means for $n_1 = 8, n_2 = 20$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. *(continued)*

		$\alpha = 0.05$			$\alpha = 0.01$		
	d	0.0	1.0	2.5	0.0	1.0	2.5
$0.9N(0,1)+$	HD	0.056	0.698	0.999	0.012	0.416	0.993
$0.1N(0,1)/U(0,1)$	NED(0.15)	0.055	0.684	0.998	0.012	0.407	0.987
	NED(0.30)	0.056	0.689	0.998	0.013	0.413	0.991
	NED(0.50)	0.056	0.697	0.998	0.013	0.420	0.993
	NED(1.00)	0.058	0.710	0.999	0.014	0.433	0.995
	NED(2.00)	0.062	0.716	1.000	0.016	0.444	0.996
	LD(MLE)	0.053	0.574	0.878	0.009	0.321	0.807
	TIKU	0.052	0.679	0.997	0.011	0.384	0.990
	t	0.046	0.552	0.874	0.008	0.294	0.798
$0.9N(0,1) + 0.1t(1)$	HD	0.060	0.374	0.867	0.013	0.165	0.714
	NED(0.15)	0.066	0.359	0.817	0.016	0.157	0.651
	NED(0.30)	0.067	0.380	0.851	0.017	0.181	0.698
	NED(0.50)	0.067	0.403	0.877	0.017	0.197	0.742
	NED(1.00)	0.071	0.431	0.907	0.017	0.224	0.790
	NED(2.00)	0.068	0.455	0.923	0.020	0.243	0.825
	LD(MLE)	0.048	0.164	0.381	0.005	0.044	0.206
	TIKU	0.064	0.349	0.787	0.011	0.141	0.612
	t	0.038	0.148	0.362	0.004	0.036	0.191

Table A.11: Empirical levels and powers: Tests for equality of two population means for $n_i = 40, n_2 = 10$, with square root weights and k=0.9 if $n_1 < 30$ for $i = 1, 2$.

$k = 0.9$ if $n_i < 30$ $i = 1, 2$		$\alpha = 0.05$			$\alpha = 0.01$		
$(n_1 = 40, n_2 = 10)$	d	0.0	1.0	2.5	0.0	1.0	2.5
$N(0,1)$	HD	0.050	0.880	1.000	0.012	0.675	1.000
	NED(0.15)	0.050	0.874	1.000	0.010	0.655	1.000
	NED(0.30)	0.051	0.875	1.000	0.010	0.660	1.000
	NED(0.50)	0.050	0.876	1.000	0.011	0.665	1.000
	NED(1.00)	0.052	0.880	1.000	0.013	0.678	1.000
	NED(2.00)	0.059	0.884	1.000	0.014	0.693	1.000
	LD(MLE)	0.053	0.883	1.000	0.011	0.684	1.000
	TIKU	0.051	0.864	1.000	0.012	0.644	1.000
	t	0.050	0.876	1.000	0.011	0.665	1.000
$t(2)$	HD	0.056	0.604	0.995	0.012	0.325	0.976
	NED(0.15)	0.061	0.537	0.980	0.012	0.279	0.928
	NED(0.30)	0.061	0.578	0.990	0.016	0.313	0.957
	NED(0.50)	0.062	0.613	0.995	0.017	0.347	0.974
	NED(1.00)	0.066	0.659	0.998	0.020	0.396	0.984
	NED(2.00)	0.073	0.694	0.999	0.021	0.440	0.992
	LD(MLE)	0.052	0.393	0.873	0.009	0.178	0.760
	TIKU	0.065	0.608	0.992	0.015	0.350	0.966
	t	0.048	0.379	0.870	0.008	0.165	0.749

Table A.11: Empirical levels and powers: Tests for equality of two population means for $n_1 = 40, n_2 = 10$, with square root weights and k=0.9 if $n_i < 30$ for $i = 1, 2$. *(continued)*

		$\alpha = 0.05$			$\alpha = 0.01$		
	d	0.0	1.0	2.5	0.0	1.0	2.5
$0.9N(0,1)+$	HD	0.056	0.823	1.000	0.013	0.597	1.000
$0.1N(0,1)/U(0,1)$	NED(0.15)	0.055	0.794	1.000	0.014	0.558	0.999
	NED(0.30)	0.057	0.809	1.000	0.014	0.578	1.000
	NED(0.50)	0.057	0.819	1.000	0.013	0.598	1.000
	NED(1.00)	0.060	0.831	1.000	0.015	0.619	1.000
	NED(2.00)	0.065	0.839	1.000	0.019	0.636	1.000
	LD(MLE)	0.061	0.562	0.862	0.012	0.357	0.802
	TIKU	0.059	0.798	0.999	0.014	0.560	0.998
	t	0.058	0.553	0.859	0.011	0.345	0.795
$0.9N(0,1) + 0.1t(1)$	HD	0.060	0.482	0.955	0.017	0.252	0.879
	NED(0.15)	0.069	0.403	0.887	0.019	0.201	0.757
	NED(0.30)	0.069	0.460	0.929	0.020	0.246	0.839
	NED(0.50)	0.069	0.502	0.953	0.021	0.281	0.884
	NED(1.00)	0.076	0.559	0.973	0.024	0.337	0.929
	NED(2.00)	0.079	0.608	0.982	0.026	0.382	0.950
	LD(MLE)	0.064	0.142	0.331	0.008	0.042	0.179
	TIKU	0.087	0.413	0.871	0.027	0.211	0.749
	t	0.062	0.138	0.322	0.007	0.037	0.171

APPENDIX B

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF TWO POPULATION MEANS WHEN THEIR VARIANCES ARE UNEQUAL

Table B.1: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
(σ_1, σ_2)	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.059	0.558	0.999	0.011	0.299	0.994
	NED	0.062	0.559	0.999	0.013	0.298	0.994
	WELCH	0.050	0.549	0.999	0.008	0.284	0.994
	TIKU	0.051	0.531	0.999	0.011	0.254	0.984
(1:2)	HD	0.062	0.270	0.914	0.016	0.097	0.707
	NED	0.059	0.267	0.909	0.014	0.095	0.700
	WELCH	0.053	0.261	0.911	0.010	0.088	0.696
	TIKU	0.059	0.245	0.886	0.013	0.088	0.639
(1:4)	HD	0.065	0.114	0.424	0.017	0.031	0.181
	NED	0.057	0.108	0.414	0.013	0.028	0.173
	WELCH	0.056	0.105	0.412	0.011	0.025	0.171
	TIKU	0.059	0.109	0.386	0.013	0.030	0.159

Table B.1: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.052	0.355	0.920	0.016	0.150	0.807
	NED	0.057	0.366	0.912	0.017	0.161	0.805
	WELCH	0.019	0.174	0.569	0.004	0.052	0.383
	TIKU	0.041	0.297	0.866	0.009	0.104	0.720
(1:2)	HD	0.061	0.180	0.662	0.019	0.064	0.424
	NED	0.058	0.179	0.648	0.017	0.064	0.410
	WELCH	0.028	0.088	0.361	0.005	0.021	0.177
	TIKU	0.050	0.144	0.586	0.012	0.047	0.307
(1:4)	HD	0.058	0.096	0.282	0.014	0.028	0.106
	NED	0.053	0.091	0.270	0.011	0.026	0.097
	WELCH	0.022	0.047	0.154	0.004	0.009	0.044
	TIKU	0.043	0.079	0.229	0.008	0.022	0.073

Table B.1: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.050	0.798	0.999	0.010	0.556	0.998
	NED	0.056	0.799	0.999	0.011	0.557	0.995
	WELCH	0.034	0.626	0.896	0.004	0.399	0.852
	TIKU	0.042	0.765	0.996	0.009	0.489	0.992
(1:2)	HD	0.055	0.417	0.980	0.015	0.199	0.917
	NED	0.055	0.411	0.973	0.015	0.191	0.905
	WELCH	0.037	0.321	0.834	0.008	0.142	0.735
	TIKU	0.049	0.385	0.973	0.012	0.165	0.879
(1:4)	HD	0.057	0.179	0.650	0.018	0.063	0.378
	NED	0.053	0.174	0.639	0.015	0.057	0.359
	WELCH	0.034	0.133	0.525	0.006	0.040	0.280
	TIKU	0.051	0.160	0.607	0.010	0.048	0.318

Table B.1: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$. *(continued)*

$t(4)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.054	0.683	0.999	0.012	0.427	0.995
	NED	0.059	0.693	0.999	0.014	0.433	0.993
	WELCH	0.044	0.631	0.994	0.008	0.367	0.978
	TIKU	0.053	0.677	0.999	0.011	0.402	0.993
(1:2)	HD	0.053	0.368	0.954	0.010	0.161	0.842
	NED	0.052	0.366	0.945	0.009	0.159	0.825
	WELCH	0.041	0.331	0.930	0.006	0.131	0.795
	TIKU	0.051	0.359	0.953	0.010	0.145	0.816
(1:4)	HD	0.060	0.146	0.572	0.016	0.050	0.303
	NED	0.055	0.142	0.553	0.015	0.045	0.288
	WELCH	0.046	0.132	0.529	0.009	0.036	0.261
	TIKU	0.052	0.139	0.561	0.011	0.042	0.278

Table B.1: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 10$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.061	0.176	0.538	0.018	0.062	0.266
	NED	0.066	0.179	0.539	0.021	0.063	0.267
	WELCH	0.046	0.171	0.537	0.009	0.055	0.261
	TIKU	0.047	0.139	0.424	0.010	0.039	0.172
(1:2)	HD	0.068	0.105	0.245	0.024	0.038	0.087
	NED	0.066	0.104	0.245	0.020	0.035	0.084
	WELCH	0.058	0.097	0.238	0.017	0.027	0.081
	TIKU	0.054	0.083	0.189	0.017	0.028	0.064
(1:4)	HD	0.063	0.075	0.113	0.021	0.029	0.037
	NED	0.057	0.069	0.108	0.017	0.024	0.031
	WELCH	0.054	0.066	0.105	0.013	0.019	0.027
	TIKU	0.057	0.068	0.096	0.019	0.023	0.030

Table B.2: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.060	0.700	1.000	0.014	0.431	1.000
	NED	0.072	0.704	1.000	0.019	0.447	1.000
	WELCH	0.053	0.693	1.000	0.011	0.420	1.000
	TIKU	0.061	0.688	1.000	0.015	0.426	0.998
(1:2)	HD	0.057	0.419	0.992	0.011	0.195	0.955
	NED	0.060	0.424	0.991	0.012	0.199	0.955
	WELCH	0.054	0.415	0.991	0.009	0.191	0.954
	TIKU	0.055	0.398	0.989	0.010	0.175	0.933
(1:4)	HD	0.051	0.179	0.718	0.012	0.058	0.467
	NED	0.051	0.175	0.717	0.011	0.057	0.459
	WELCH	0.049	0.174	0.714	0.010	0.055	0.459
	TIKU	0.053	0.173	0.689	0.010	0.053	0.418

Table B.2: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.067	0.455	0.954	0.016	0.230	0.889
	NED	0.075	0.458	0.953	0.019	0.238	0.894
	WELCH	0.029	0.216	0.634	0.004	0.079	0.454
	TIKU	0.063	0.422	0.929	0.014	0.209	0.862
(1:2)	HD	0.057	0.263	0.849	0.013	0.103	0.677
	NED	0.062	0.258	0.813	0.013	0.104	0.647
	WELCH	0.033	0.114	0.437	0.003	0.031	0.244
	TIKU	0.059	0.237	0.809	0.012	0.082	0.613
(1:4)	HD	0.048	0.129	0.462	0.012	0.035	0.239
	NED	0.052	0.123	0.435	0.011	0.031	0.225
	WELCH	0.023	0.060	0.218	0.003	0.007	0.079
	TIKU	0.044	0.119	0.428	0.008	0.024	0.195

Table B.2: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.060	0.889	1.000	0.013	0.692	0.999
	NED	0.071	0.893	1.000	0.018	0.701	0.999
	WELCH	0.038	0.690	0.901	0.005	0.491	0.860
	TIKU	0.060	0.892	0.998	0.013	0.703	0.996
(1:2)	HD	0.045	0.642	0.999	0.010	0.390	0.995
	NED	0.050	0.632	0.997	0.011	0.387	0.989
	WELCH	0.032	0.473	0.845	0.006	0.261	0.793
	TIKU	0.047	0.623	0.995	0.009	0.360	0.989
(1:4)	HD	0.054	0.283	0.908	0.012	0.115	0.748
	NED	0.053	0.275	0.894	0.012	0.110	0.730
	WELCH	0.037	0.203	0.697	0.007	0.073	0.525
	TIKU	0.054	0.269	0.900	0.012	0.100	0.715

Table B.2: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$. *(continued)*

$t(4)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.052	0.799	1.000	0.014	0.565	0.997
	NED	0.064	0.810	1.000	0.019	0.595	0.998
	WELCH	0.044	0.741	0.997	0.008	0.505	0.989
	TIKU	0.060	0.827	0.999	0.014	0.616	0.998
(1:2)	HD	0.053	0.543	0.997	0.009	0.296	0.983
	NED	0.057	0.539	0.996	0.010	0.294	0.979
	WELCH	0.044	0.496	0.987	0.006	0.257	0.954
	TIKU	0.054	0.563	0.997	0.010	0.303	0.981
(1:4)	HD	0.056	0.247	0.817	0.010	0.089	0.612
	NED	0.055	0.239	0.803	0.009	0.085	0.592
	WELCH	0.051	0.220	0.773	0.009	0.077	0.562
	TIKU	0.053	0.256	0.833	0.011	0.093	0.625

Table B.2: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 20$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.057	0.219	0.671	0.017	0.078	0.379
	NED	0.071	0.226	0.671	0.032	0.088	0.381
	WELCH	0.050	0.216	0.671	0.013	0.072	0.378
	TIKU	0.052	0.183	0.566	0.017	0.065	0.298
(1:2)	HD	0.055	0.138	0.405	0.013	0.045	0.181
	NED	0.062	0.142	0.407	0.016	0.049	0.185
	WELCH	0.050	0.134	0.403	0.010	0.039	0.179
	TIKU	0.047	0.102	0.302	0.009	0.028	0.117
(1:4)	HD	0.057	0.073	0.174	0.012	0.021	0.058
	NED	0.058	0.073	0.174	0.012	0.020	0.058
	WELCH	0.055	0.071	0.173	0.011	0.020	0.056
	TIKU	0.048	0.061	0.133	0.011	0.014	0.038

Table B.3: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.056	0.690	1.000	0.014	0.429	1.000
	NED	0.053	0.685	1.000	0.012	0.422	0.999
	WELCH	0.050	0.684	1.000	0.010	0.419	0.999
	TIKU	0.057	0.669	1.000	0.012	0.392	0.998
(1:2)	HD	0.056	0.288	0.919	0.016	0.111	0.722
	NED	0.050	0.278	0.916	0.012	0.103	0.714
	WELCH	0.047	0.275	0.916	0.009	0.099	0.715
	TIKU	0.059	0.282	0.898	0.015	0.119	0.694
(1:4)	HD	0.063	0.120	0.420	0.017	0.039	0.184
	NED	0.057	0.114	0.412	0.013	0.035	0.174
	WELCH	0.052	0.109	0.410	0.010	0.032	0.170
	TIKU	0.057	0.116	0.392	0.016	0.038	0.170

Table B.3: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.0576	0.4394	0.9554	0.0142	0.223	0.8964
	NED	0.0568	0.435	0.9396	0.015	0.2174	0.874
	WELCH	0.0256	0.2102	0.6442	0.0026	0.0742	0.4556
	TIKU	0.0426	0.3824	0.9214	0.0088	0.1762	0.8246
(1:2)	HD	0.056	0.189	0.712	0.017	0.070	0.460
	NED	0.054	0.183	0.687	0.015	0.064	0.440
	WELCH	0.032	0.095	0.404	0.004	0.021	0.215
	TIKU	0.050	0.167	0.650	0.013	0.056	0.394
(1:4)	HD	0.062	0.100	0.298	0.020	0.030	0.123
	NED	0.057	0.095	0.290	0.016	0.027	0.112
	WELCH	0.029	0.050	0.172	0.006	0.010	0.057
	TIKU	0.051	0.086	0.255	0.010	0.024	0.096

Table B.3: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.056	0.894	1.000	0.013	0.720	1.000
	NED	0.055	0.885	1.000	0.011	0.709	0.999
	WELCH	0.037	0.685	0.902	0.007	0.505	0.866
	TIKU	0.052	0.877	0.997	0.011	0.684	0.996
(1:2)	HD	0.062	0.466	0.986	0.014	0.226	0.925
	NED	0.058	0.453	0.977	0.012	0.212	0.909
	WELCH	0.038	0.346	0.847	0.008	0.154	0.750
	TIKU	0.057	0.440	0.978	0.013	0.219	0.903
(1:4)	HD	0.064	0.173	0.649	0.017	0.060	0.382
	NED	0.057	0.164	0.637	0.013	0.052	0.365
	WELCH	0.041	0.132	0.521	0.007	0.039	0.284
	TIKU	0.060	0.169	0.603	0.012	0.056	0.330

Table B.3: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$. *(continued)*

		$t(4)$					
		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.053	0.806	1.000	0.012	0.579	0.998
	NED	0.052	0.793	1.000	0.012	0.564	0.997
	WELCH	0.046	0.753	1.000	0.009	0.518	0.994
	TIKU	0.050	0.807	1.000	0.010	0.572	0.998
(1:2)	HD	0.058	0.409	0.967	0.017	0.191	0.863
	NED	0.053	0.395	0.956	0.014	0.179	0.846
	WELCH	0.046	0.373	0.941	0.009	0.162	0.820
	TIKU	0.065	0.427	0.966	0.017	0.207	0.860
(1:4)	HD	0.057	0.145	0.573	0.017	0.049	0.304
	NED	0.050	0.139	0.557	0.015	0.044	0.287
	WELCH	0.043	0.127	0.536	0.009	0.036	0.265
	TIKU	0.058	0.144	0.567	0.014	0.046	0.286

Table B.3: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 20, n_2 = 10$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.059	0.221	0.668	0.020	0.086	0.387
	NED	0.056	0.218	0.668	0.017	0.082	0.384
	WELCH	0.053	0.217	0.668	0.013	0.080	0.386
	TIKU	0.048	0.173	0.556	0.010	0.061	0.279
(1:2)	HD	0.062	0.110	0.252	0.025	0.039	0.093
	NED	0.055	0.105	0.247	0.021	0.034	0.087
	WELCH	0.053	0.102	0.248	0.017	0.031	0.087
	TIKU	0.061	0.102	0.206	0.019	0.039	0.085
(1:4)	HD	0.069	0.077	0.111	0.028	0.029	0.038
	NED	0.061	0.067	0.106	0.022	0.022	0.033
	WELCH	0.055	0.065	0.104	0.015	0.017	0.029
	TIKU	0.064	0.072	0.105	0.025	0.025	0.036

Table B.4: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.053	0.968	1.000	0.011	0.883	1.000
	NED	0.054	0.966	1.000	0.013	0.882	1.000
	WELCH	0.048	0.966	1.000	0.010	0.879	1.000
	TIKU	0.052	0.959	1.000	0.011	0.859	1.000
(1:2)	HD	0.053	0.681	1.000	0.011	0.437	1.000
	NED	0.051	0.677	1.000	0.010	0.428	1.000
	WELCH	0.049	0.677	1.000	0.010	0.424	1.000
	TIKU	0.051	0.655	1.000	0.013	0.401	1.000
(1:4)	HD	0.053	0.258	0.903	0.009	0.106	0.725
	NED	0.048	0.251	0.901	0.009	0.100	0.716
	WELCH	0.048	0.251	0.901	0.008	0.100	0.715
	TIKU	0.054	0.246	0.887	0.008	0.097	0.684

Table B.4: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.059	0.805	1.000	0.013	0.609	0.999
	NED	0.061	0.780	0.998	0.013	0.586	0.995
	WELCH	0.041	0.288	0.840	0.003	0.125	0.684
	TIKU	0.060	0.734	0.994	0.011	0.522	0.987
(1:2)	HD	0.053	0.464	0.990	0.011	0.243	0.954
	NED	0.053	0.438	0.966	0.011	0.220	0.906
	WELCH	0.032	0.167	0.581	0.002	0.049	0.375
	TIKU	0.060	0.404	0.961	0.009	0.199	0.900
(1:4)	HD	0.062	0.181	0.694	0.014	0.066	0.457
	NED	0.060	0.170	0.639	0.014	0.059	0.401
	WELCH	0.034	0.083	0.280	0.004	0.017	0.118
	TIKU	0.056	0.169	0.619	0.015	0.055	0.365

Table B.4: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.056	0.999	1.000	0.013	0.991	1.000
	NED	0.058	0.999	1.000	0.014	0.988	1.000
	WELCH	0.032	0.751	0.915	0.005	0.658	0.885
	TIKU	0.058	0.997	1.000	0.010	0.985	1.000
(1:2)	HD	0.059	0.897	1.000	0.012	0.732	1.000
	NED	0.059	0.877	1.000	0.012	0.702	1.000
	WELCH	0.037	0.600	0.865	0.006	0.421	0.808
	TIKU	0.053	0.879	1.000	0.010	0.690	1.000
(1:4)	HD	0.059	0.426	0.988	0.013	0.206	0.946
	NED	0.055	0.406	0.979	0.011	0.192	0.926
	WELCH	0.033	0.270	0.739	0.008	0.110	0.625
	TIKU	0.053	0.395	0.984	0.013	0.182	0.925

Table B.4: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$. *(continued)*

		$t(4)$					
		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.051	0.991	1.000	0.013	0.961	1.000
	NED	0.058	0.990	1.000	0.014	0.960	1.000
	WELCH	0.047	0.968	1.000	0.011	0.904	1.000
	TIKU	0.058	0.991	1.000	0.014	0.961	1.000
(1:2)	HD	0.052	0.809	1.000	0.012	0.592	1.000
	NED	0.053	0.787	1.000	0.011	0.572	1.000
	WELCH	0.049	0.744	0.999	0.009	0.515	0.998
	TIKU	0.056	0.815	1.000	0.010	0.601	1.000
(1:4)	HD	0.056	0.353	0.970	0.012	0.161	0.878
	NED	0.052	0.331	0.958	0.011	0.148	0.846
	WELCH	0.048	0.304	0.939	0.010	0.133	0.814
	TIKU	0.060	0.357	0.970	0.014	0.157	0.882

Table B.4: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 30$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.044	0.473	0.969	0.009	0.241	0.882
	NED	0.044	0.471	0.970	0.009	0.238	0.882
	WELCH	0.043	0.473	0.970	0.008	0.238	0.881
	TIKU	0.032	0.365	0.921	0.006	0.150	0.757
(1:2)	HD	0.056	0.227	0.657	0.011	0.086	0.403
	NED	0.052	0.224	0.658	0.009	0.085	0.398
	WELCH	0.053	0.225	0.657	0.010	0.085	0.399
	TIKU	0.043	0.166	0.532	0.010	0.055	0.266
(1:4)	HD	0.060	0.102	0.250	0.014	0.029	0.094
	NED	0.055	0.099	0.248	0.012	0.026	0.092
	WELCH	0.057	0.100	0.249	0.012	0.027	0.093
	TIKU	0.049	0.078	0.185	0.010	0.021	0.061

Table B.5: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.058	0.750	1.000	0.014	0.482	1.000
	NED	0.071	0.756	1.000	0.021	0.498	1.000
	WELCH	0.050	0.744	1.000	0.008	0.466	1.000
	TIKU	0.067	0.769	1.000	0.018	0.554	1.000
(1:2)	HD	0.056	0.586	1.000	0.010	0.337	0.997
	NED	0.060	0.590	1.000	0.013	0.342	0.997
	WELCH	0.050	0.580	1.000	0.008	0.326	0.996
	TIKU	0.056	0.579	1.000	0.011	0.343	0.996
(1:4)	HD	0.052	0.283	0.935	0.011	0.115	0.799
	NED	0.052	0.281	0.931	0.011	0.112	0.796
	WELCH	0.048	0.278	0.932	0.009	0.108	0.794
	TIKU	0.049	0.277	0.923	0.012	0.106	0.771

Table B.5: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.061	0.524	0.970	0.016	0.286	0.916
	NED	0.073	0.544	0.985	0.021	0.305	0.948
	WELCH	0.031	0.249	0.722	0.004	0.093	0.555
	TIKU	0.067	0.517	0.956	0.016	0.301	0.921
(1:2)	HD	0.054	0.374	0.944	0.010	0.169	0.864
	NED	0.057	0.370	0.946	0.014	0.169	0.860
	WELCH	0.033	0.140	0.521	0.003	0.040	0.322
	TIKU	0.057	0.347	0.920	0.012	0.156	0.827
(1:4)	HD	0.058	0.190	0.714	0.011	0.068	0.486
	NED	0.058	0.179	0.664	0.011	0.062	0.445
	WELCH	0.035	0.081	0.267	0.002	0.013	0.115
	TIKU	0.058	0.180	0.656	0.009	0.059	0.416

Table B.5: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.063	0.920	1.000	0.016	0.767	0.999
	NED	0.075	0.930	1.000	0.024	0.782	0.999
	WELCH	0.039	0.702	0.901	0.007	0.532	0.863
	TIKU	0.073	0.931	0.999	0.023	0.817	0.998
(1:2)	HD	0.058	0.823	1.000	0.012	0.611	0.999
	NED	0.067	0.822	1.000	0.015	0.607	0.999
	WELCH	0.037	0.552	0.860	0.006	0.351	0.809
	TIKU	0.061	0.821	0.998	0.011	0.608	0.997
(1:4)	HD	0.053	0.473	0.994	0.010	0.240	0.971
	NED	0.055	0.462	0.991	0.010	0.232	0.955
	WELCH	0.036	0.292	0.746	0.005	0.129	0.644
	TIKU	0.053	0.460	0.991	0.010	0.229	0.958

Table B.5: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$. *(continued)*

$t(4)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.057	0.840	1.000	0.015	0.633	0.998
	NED	0.071	0.854	1.000	0.023	0.665	0.999
	WELCH	0.043	0.793	0.997	0.009	0.568	0.989
	TIKU	0.068	0.875	1.000	0.020	0.734	0.999
(1:2)	HD	0.054	0.712	0.999	0.010	0.474	0.996
	NED	0.060	0.721	1.000	0.012	0.477	0.997
	WELCH	0.047	0.662	0.996	0.006	0.405	0.985
	TIKU	0.062	0.743	0.999	0.011	0.521	0.998
(1:4)	HD	0.057	0.388	0.978	0.011	0.182	0.918
	NED	0.054	0.378	0.972	0.012	0.174	0.902
	WELCH	0.050	0.342	0.953	0.008	0.147	0.861
	TIKU	0.057	0.409	0.979	0.010	0.189	0.923

Table B.5: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 10, n_2 = 40$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.062	0.246	0.736	0.026	0.097	0.441
	NED	0.054	0.255	0.737	0.019	0.110	0.444
	WELCH	0.053	0.239	0.734	0.017	0.090	0.439
	TIKU	0.064	0.224	0.673	0.027	0.107	0.416
(1:2)	HD	0.064	0.189	0.566	0.023	0.064	0.295
	NED	0.081	0.194	0.567	0.041	0.070	0.297
	WELCH	0.054	0.187	0.565	0.013	0.061	0.293
	TIKU	0.065	0.151	0.469	0.022	0.052	0.218
(1:4)	HD	0.053	0.107	0.284	0.013	0.029	0.116
	NED	0.066	0.108	0.283	0.022	0.031	0.118
	WELCH	0.048	0.103	0.279	0.010	0.028	0.113
	TIKU	0.045	0.077	0.211	0.009	0.017	0.071

Table B.6: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.057	0.762	1.000	0.015	0.486	0.999
	NED	0.053	0.757	1.000	0.014	0.476	0.999
	WELCH	0.048	0.755	1.000	0.011	0.473	0.999
	TIKU	0.053	0.747	0.999	0.011	0.478	0.988
(1:2)	HD	0.063	0.299	0.938	0.019	0.113	0.753
	NED	0.058	0.293	0.936	0.016	0.106	0.745
	WELCH	0.054	0.292	0.934	0.012	0.103	0.744
	TIKU	0.071	0.313	0.926	0.025	0.141	0.745
(1:4)	HD	0.064	0.114	0.439	0.019	0.033	0.195
	NED	0.056	0.107	0.428	0.015	0.029	0.182
	WELCH	0.051	0.104	0.423	0.011	0.026	0.177
	TIKU	0.065	0.116	0.416	0.019	0.035	0.180

Table B.6: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.062	0.538	0.966	0.018	0.283	0.918
	NED	0.060	0.520	0.939	0.016	0.267	0.880
	WELCH	0.032	0.243	0.715	0.004	0.097	0.545
	TIKU	0.046	0.461	0.932	0.009	0.210	0.858
(1:2)	HD	0.059	0.204	0.734	0.017	0.075	0.489
	NED	0.053	0.196	0.710	0.013	0.068	0.467
	WELCH	0.028	0.103	0.455	0.004	0.029	0.257
	TIKU	0.062	0.194	0.685	0.018	0.076	0.456
(1:4)	HD	0.065	0.096	0.302	0.019	0.029	0.117
	NED	0.057	0.089	0.289	0.016	0.025	0.109
	WELCH	0.032	0.049	0.174	0.005	0.012	0.060
	TIKU	0.060	0.084	0.252	0.019	0.026	0.090

Table B.6: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.057	0.932	1.000	0.015	0.779	0.999
	NED	0.055	0.921	1.000	0.012	0.764	0.997
	WELCH	0.034	0.710	0.905	0.006	0.537	0.867
	TIKU	0.053	0.914	0.998	0.008	0.755	0.990
(1:2)	HD	0.059	0.478	0.986	0.016	0.238	0.931
	NED	0.054	0.467	0.976	0.013	0.225	0.916
	WELCH	0.036	0.362	0.843	0.006	0.166	0.752
	TIKU	0.070	0.484	0.984	0.021	0.255	0.921
(1:4)	HD	0.071	0.161	0.670	0.021	0.054	0.383
	NED	0.063	0.152	0.655	0.018	0.047	0.365
	WELCH	0.045	0.122	0.553	0.009	0.034	0.298
	TIKU	0.066	0.152	0.638	0.019	0.054	0.349

Table B.6: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$. *(continued)*

$t(4)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.056	0.862	1.000	0.015	0.666	0.999
	NED	0.057	0.852	1.000	0.013	0.648	0.998
	WELCH	0.050	0.824	0.999	0.011	0.606	0.995
	TIKU	0.052	0.864	1.000	0.010	0.657	0.986
(1:2)	HD	0.058	0.411	0.965	0.015	0.194	0.865
	NED	0.053	0.398	0.958	0.013	0.181	0.848
	WELCH	0.047	0.372	0.949	0.011	0.159	0.829
	TIKU	0.065	0.439	0.969	0.019	0.235	0.878
(1:4)	HD	0.059	0.160	0.606	0.015	0.053	0.324
	NED	0.053	0.149	0.587	0.011	0.047	0.304
	WELCH	0.047	0.135	0.565	0.008	0.037	0.285
	TIKU	0.059	0.159	0.591	0.015	0.057	0.314

Table B.6: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = 40, n_2 = 10$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.060	0.242	0.752	0.020	0.088	0.442
	NED	0.051	0.238	0.753	0.014	0.082	0.441
	WELCH	0.049	0.237	0.752	0.011	0.081	0.442
	TIKU	0.045	0.205	0.665	0.007	0.071	0.367
(1:2)	HD	0.061	0.112	0.259	0.024	0.037	0.099
	NED	0.051	0.105	0.254	0.019	0.031	0.092
	WELCH	0.051	0.102	0.254	0.015	0.027	0.091
	TIKU	0.071	0.115	0.232	0.032	0.042	0.108
(1:4)	HD	0.062	0.078	0.117	0.026	0.030	0.042
	NED	0.054	0.070	0.109	0.019	0.022	0.035
	WELCH	0.053	0.067	0.107	0.017	0.016	0.031
	TIKU	0.064	0.073	0.109	0.027	0.027	0.037

Table B.7: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$.

$N(0,1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.055	1.000	1.000	0.013	0.999	1.000
	NED	0.055	1.000	1.000	0.013	0.999	1.000
	WELCH	0.052	1.000	1.000	0.013	0.999	1.000
	TIKU	0.054	1.000	1.000	0.012	0.999	1.000
(1:2)	HD	0.054	0.959	1.000	0.009	0.864	1.000
	NED	0.053	0.958	1.000	0.008	0.864	1.000
	WELCH	0.052	0.958	1.000	0.009	0.862	1.000
	TIKU	0.053	0.951	1.000	0.009	0.848	1.000
(1:4)	HD	0.047	0.513	0.998	0.012	0.274	0.990
	NED	0.046	0.506	0.998	0.011	0.269	0.989
	WELCH	0.047	0.506	0.998	0.011	0.268	0.989
	TIKU	0.046	0.495	0.998	0.010	0.262	0.987

Table B.7: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$. *(continued)*

$0.9N(0,1) + 0.1N(0,64)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.056	0.991	1.000	0.011	0.955	1.000
	NED	0.058	0.977	1.000	0.012	0.924	1.000
	WELCH	0.045	0.500	0.991	0.007	0.283	0.961
	TIKU	0.061	0.975	1.000	0.014	0.922	1.000
(1:2)	HD	0.056	0.788	1.000	0.014	0.571	1.000
	NED	0.056	0.713	0.997	0.013	0.495	0.992
	WELCH	0.055	0.246	0.814	0.006	0.103	0.658
	TIKU	0.065	0.745	0.999	0.015	0.512	0.996
(1:4)	HD	0.052	0.322	0.954	0.012	0.149	0.860
	NED	0.052	0.285	0.890	0.010	0.124	0.759
	WELCH	0.037	0.108	0.416	0.003	0.026	0.223
	TIKU	0.060	0.300	0.922	0.012	0.133	0.806

Table B.7: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$. *(continued)*

$0.9N(0,1) + 0.1N[(0,1)/U(0,1)]$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.053	1.000	1.000	0.011	1.000	1.000
	NED	0.053	1.000	1.000	0.012	1.000	1.000
	WELCH	0.031	0.770	0.913	0.003	0.703	0.884
	TIKU	0.055	1.000	1.000	0.011	1.000	1.000
(1:2)	HD	0.055	0.999	1.000	0.012	0.989	1.000
	NED	0.052	0.997	1.000	0.014	0.981	1.000
	WELCH	0.033	0.659	0.861	0.006	0.545	0.817
	TIKU	0.058	0.997	1.000	0.014	0.986	1.000
(1:4)	HD	0.051	0.761	1.000	0.011	0.538	1.000
	NED	0.048	0.736	1.000	0.011	0.503	0.999
	WELCH	0.031	0.387	0.771	0.007	0.216	0.688
	TIKU	0.055	0.747	1.000	0.012	0.513	1.000

Table B.7: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$. *(continued)*

$t(4)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	1.0	2.5	0.0	1.0	2.5
(1:1)	HD	0.050	1.000	1.000	0.011	1.000	1.000
	NED	0.053	1.000	1.000	0.012	1.000	1.000
	WELCH	0.051	0.999	1.000	0.009	0.998	1.000
	TIKU	0.057	1.000	1.000	0.013	1.000	1.000
(1:2)	HD	0.053	0.990	1.000	0.009	0.956	1.000
	NED	0.056	0.985	1.000	0.010	0.942	1.000
	WELCH	0.053	0.976	1.000	0.009	0.913	1.000
	TIKU	0.060	0.991	1.000	0.013	0.965	1.000
(1:4)	HD	0.047	0.660	1.000	0.009	0.412	0.999
	NED	0.049	0.624	0.999	0.009	0.384	0.998
	WELCH	0.045	0.590	0.999	0.007	0.357	0.994
	TIKU	0.054	0.684	1.000	0.010	0.444	0.999

Table B.7: Empirical levels and powers: Tests for equality of two population means with unequal variances for $n_1 = n_2 = 70$. *(continued)*

$Uniform(-1, 1)$

		$\alpha = 0.05$			$\alpha = 0.01$		
$(\sigma_1 : \sigma_2)$	d=	0.0	0.5	1.0	0.0	0.5	1.0
(1:1)	HD	0.051	0.837	1.000	0.008	0.637	1.000
	NED	0.049	0.838	1.000	0.008	0.637	1.000
	WELCH	0.051	0.836	1.000	0.008	0.636	1.000
	TIKU	0.034	0.729	1.000	0.005	0.472	0.998
(1:2)	HD	0.051	0.446	0.960	0.013	0.222	0.868
	NED	0.050	0.443	0.962	0.013	0.221	0.868
	WELCH	0.052	0.443	0.960	0.013	0.222	0.867
	TIKU	0.038	0.338	0.907	0.009	0.140	0.734
(1:4)	HD	0.049	0.163	0.518	0.009	0.051	0.268
	NED	0.047	0.162	0.516	0.008	0.049	0.265
	WELCH	0.048	0.163	0.517	0.009	0.051	0.265
	TIKU	0.035	0.110	0.397	0.006	0.034	0.170

APPENDIX C

MEANS, VARIANCES, AND MEAN SQUARED ERRORS OF THE *NED(a)* ESTIMATES FOR THE POPULATION MEANS

Table C.1: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $N(\mu_i, 1)$, $i = 1, 2$.

(n_1, n_2)=(10,10)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0002	0.0990	0.0990	-0.0015	0.1090	0.1090
	0.30	0.0022	0.0977	0.0977	-0.0013	0.1090	0.1090
	0.50	0.0030	0.1013	0.1013	-0.0030	0.1153	0.1153
	0.70	0.0032	0.1059	0.1059	-0.0044	0.1220	0.1220
	0.90	0.0025	0.1111	0.1111	-0.0036	0.1265	0.1265
	1.00	0.0037	0.1120	0.1121	-0.0042	0.1294	0.1295
	1.50	0.0044	0.1199	0.1199	-0.0052	0.1397	0.1397
	2.00	0.0057	0.1265	0.1266	-0.0062	0.1475	0.1475
	2.50	0.0054	0.1333	0.1333	-0.0073	0.1545	0.1546
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9980	0.0992	0.0992	-0.0018	0.1096	0.1096
	0.30	1.0026	0.1028	0.1028	-0.0007	0.1082	0.1082
	0.50	1.0027	0.1071	0.1071	-0.0029	0.1153	0.1153
	0.70	1.0030	0.1124	0.1124	-0.0044	0.1220	0.1221
	0.90	1.0023	0.1160	0.1160	-0.0040	0.1270	0.1270
	1.00	1.0035	0.1195	0.1195	-0.0039	0.1291	0.1291
	1.50	1.0041	0.1286	0.1286	-0.0055	0.1399	0.1399
	2.00	1.0054	0.1379	0.1379	-0.0063	0.1474	0.1475
	2.50	1.0054	0.1442	0.1443	-0.0071	0.1547	0.1548

Table C.1: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $N(\mu_i, 1)$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(30,30)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0028	0.0350	0.0350	-0.0008	0.0347	0.0347
	0.30	-0.0009	0.0349	0.0349	-0.0011	0.0341	0.0341
	0.50	-0.0014	0.0354	0.0354	-0.0012	0.0344	0.0344
	0.70	-0.0019	0.0363	0.0363	-0.0011	0.0347	0.0347
	0.90	-0.0020	0.0369	0.0369	-0.0008	0.0349	0.0349
	1.00	-0.0021	0.0374	0.0374	-0.0006	0.0349	0.0349
	1.50	-0.0026	0.0394	0.0394	-0.0002	0.0359	0.0359
	2.00	-0.0028	0.0410	0.0410	0.0002	0.0369	0.0369
	2.50	-0.0029	0.0427	0.0427	0.0005	0.0379	0.0379
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9939	0.0341	0.0341	-0.0004	0.0339	0.0339
	0.30	0.9989	0.0331	0.0331	-0.0011	0.0341	0.0341
	0.50	0.9984	0.0326	0.0326	-0.0012	0.0344	0.0344
	0.70	0.9981	0.0327	0.0327	-0.0011	0.0347	0.0347
	0.90	0.9978	0.0330	0.0330	-0.0009	0.0349	0.0349
	1.00	0.9976	0.0330	0.0330	-0.0007	0.0350	0.0350
	1.50	0.9972	0.0341	0.0341	-0.0002	0.0359	0.0359
	2.00	0.9970	0.0355	0.0355	0.0002	0.0369	0.0369
	2.50	0.9968	0.0367	0.0367	0.0004	0.0380	0.0380

Table C.1: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $N(\mu_i, 1)$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(50,50)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0052	0.0220	0.0220	0.0032	0.0174	0.0174
	0.30	-0.0021	0.0220	0.0220	0.0025	0.0180	0.0180
	0.50	-0.0022	0.0220	0.0221	0.0022	0.0183	0.0183
	0.70	-0.0021	0.0220	0.0220	0.0020	0.0186	0.0186
	0.90	-0.0020	0.0222	0.0222	0.0018	0.0189	0.0189
	1.00	-0.0020	0.0223	0.0223	0.0018	0.0191	0.0191
	1.50	-0.0019	0.0228	0.0228	0.0016	0.0197	0.0197
	2.00	-0.0018	0.0233	0.0234	0.0016	0.0202	0.0202
	2.50	-0.0018	0.0239	0.0239	0.0016	0.0208	0.0208
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9891	0.0183	0.0184	0.0026	0.0180	0.0180
	0.30	0.9976	0.0178	0.0178	0.0025	0.0179	0.0179
	0.50	0.9976	0.0176	0.0177	0.0022	0.0182	0.0182
	0.70	0.9978	0.0179	0.0179	0.0020	0.0186	0.0186
	0.90	0.9977	0.0181	0.0181	0.0018	0.0189	0.0189
	1.00	0.9978	0.0183	0.0183	0.0018	0.0191	0.0191
	1.50	0.9979	0.0190	0.0190	0.0016	0.0197	0.0197
	2.00	0.9979	0.0196	0.0196	0.0016	0.0203	0.0203
	2.50	0.9980	0.0203	0.0203	0.0016	0.0208	0.0208

Table C.2: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1t(1)$, $i = 1, 2$.

(n_1, n_2)=(10,10)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0042	0.6377	0.6378	-0.0094	0.6206	0.6207
	0.30	-0.0030	0.4952	0.4953	-0.0127	0.4680	0.4681
	0.50	-0.0008	0.4083	0.4083	-0.0135	0.3857	0.3858
	0.70	0.0013	0.3699	0.3699	-0.0089	0.3476	0.3477
	0.90	0.0012	0.3437	0.3437	-0.0077	0.3247	0.3248
	1.00	0.0025	0.3346	0.3346	-0.0095	0.3168	0.3169
	1.50	0.0009	0.3149	0.3149	-0.0081	0.2958	0.2958
	2.00	-0.0005	0.3053	0.3053	-0.0076	0.2850	0.2851
	2.50	-0.0002	0.2975	0.2975	-0.0078	0.2809	0.2809
$\mu_1 = 1, \mu_2 = 0$	0.15	1.0019	0.6452	0.6452	-0.0087	0.6189	0.6190
	0.30	0.9975	0.4897	0.4897	-0.0132	0.4700	0.4702
	0.50	0.9983	0.4062	0.4062	-0.0136	0.3862	0.3864
	0.70	1.0010	0.3725	0.3725	-0.0091	0.3471	0.3472
	0.90	1.0006	0.3458	0.3458	-0.0079	0.3246	0.3247
	1.00	1.0027	0.3400	0.3400	-0.0096	0.3167	0.3168
	1.50	1.0008	0.3170	0.3170	-0.0083	0.2964	0.2965
	2.00	0.9995	0.3044	0.3044	-0.0067	0.2849	0.2849
	2.50	0.9992	0.2969	0.2969	-0.0075	0.2809	0.2809

Table C.2: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1t(1)$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(30,30)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0053	0.2913	0.2913	-0.0133	0.2999	0.3001
	0.30	0.0057	0.1702	0.1702	-0.0041	0.1823	0.1823
	0.50	0.0021	0.1280	0.1281	-0.0053	0.1412	0.1412
	0.70	0.0027	0.1108	0.1108	-0.0044	0.1208	0.1208
	0.90	0.0027	0.1011	0.1011	-0.0039	0.1084	0.1085
	1.00	0.0032	0.0971	0.0971	-0.0036	0.1043	0.1043
	1.50	0.0040	0.0863	0.0863	-0.0041	0.0947	0.0947
	2.00	0.0045	0.0808	0.0808	-0.0031	0.0885	0.0885
	2.50	0.0048	0.0772	0.0772	-0.0025	0.0848	0.0848
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9958	0.2925	0.2925	-0.0127	0.2969	0.2971
	0.30	1.0056	0.1818	0.1818	-0.0048	0.1835	0.1835
	0.50	1.0019	0.1322	0.1322	-0.0051	0.1410	0.1410
	0.70	1.0026	0.1163	0.1163	-0.0043	0.1208	0.1208
	0.90	1.0026	0.1067	0.1067	-0.0037	0.1084	0.1084
	1.00	1.0028	0.1033	0.1033	-0.0037	0.1044	0.1044
	1.50	1.0042	0.0946	0.0946	-0.0040	0.0946	0.0947
	2.00	1.0041	0.0895	0.0896	-0.0030	0.0883	0.0883
	2.50	1.0046	0.0868	0.0869	-0.0026	0.0848	0.0848

Table C.2: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1t(1)$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(50,50)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	0.0000	0.2002	0.2002	0.0025	0.1981	0.1981
	0.30	0.0043	0.1115	0.1116	0.0059	0.1095	0.1096
	0.50	0.0046	0.0795	0.0795	0.0037	0.0808	0.0809
	0.70	0.0022	0.0689	0.0689	0.0048	0.0670	0.0670
	0.90	0.0029	0.0609	0.0609	0.0047	0.0599	0.0600
	1.00	0.0029	0.0588	0.0588	0.0043	0.0580	0.0580
	1.50	0.0025	0.0525	0.0525	0.0032	0.0520	0.0521
	2.00	0.0024	0.0495	0.0495	0.0020	0.0495	0.0495
	2.50	0.0028	0.0470	0.0470	0.0014	0.0479	0.0479
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9984	0.2000	0.2001	0.0023	0.1980	0.1980
	0.30	1.0034	0.1200	0.1200	0.0062	0.1095	0.1095
	0.50	1.0044	0.0887	0.0887	0.0033	0.0812	0.0812
	0.70	1.0021	0.0734	0.0734	0.0049	0.0669	0.0669
	0.90	1.0028	0.0668	0.0669	0.0049	0.0598	0.0598
	1.00	1.0027	0.0646	0.0646	0.0043	0.0580	0.0580
	1.50	1.0023	0.0575	0.0575	0.0033	0.0519	0.0519
	2.00	1.0022	0.0542	0.0542	0.0020	0.0495	0.0495
	2.50	1.0025	0.0525	0.0525	0.0013	0.0480	0.0480

Table C.3: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1[N(0,1)/U(0,1)]$, $i = 1, 2$.

(n_1, n_2)=(10,10)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	-0.0019	0.1342	0.1342	0.0006	0.1337	0.1337
	0.30	0.0119	0.1150	0.1152	0.0030	0.1271	0.1271
	0.50	0.0032	0.1241	0.1241	-0.0027	0.1310	0.1310
	0.70	-0.0068	0.1368	0.1369	0.0092	0.1174	0.1174
	0.90	0.0015	0.1267	0.1267	-0.0021	0.1322	0.1322
	1.00	0.0001	0.1279	0.1279	0.0005	0.1346	0.1346
	1.50	-0.0009	0.1394	0.1394	0.0016	0.1387	0.1387
	2.00	-0.0055	0.1520	0.1521	-0.0007	0.1453	0.1453
	2.50	-0.0036	0.1508	0.1508	-0.0010	0.1578	0.1578
$\mu_1 = 1, \mu_2 = 0$	0.15	1.0116	0.1530	0.1531	-0.0040	0.1364	0.1364
	0.30	1.0038	0.1352	0.1352	-0.0039	0.1304	0.1304
	0.50	0.9984	0.1230	0.1230	0.0004	0.1193	0.1193
	0.70	0.9953	0.1187	0.1187	0.0050	0.1240	0.1240
	0.90	1.0037	0.1342	0.1342	-0.0035	0.1357	0.1357
	1.00	1.0033	0.1346	0.1346	-0.0002	0.1328	0.1328
	1.50	0.9906	0.1293	0.1293	0.0096	0.1321	0.1322
	2.00	0.9927	0.1359	0.1360	-0.0026	0.1517	0.1517
	2.50	1.0084	0.1569	0.1570	0.0065	0.1458	0.1458

Table C.3: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1[N(0,1)/U(0,1)]$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(30,30)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	-0.0009	0.0565	0.0565	0.0000	0.0516	0.0516
	0.30	-0.0045	0.0474	0.0474	-0.0080	0.0493	0.0494
	0.50	-0.0019	0.0425	0.0425	-0.0032	0.0433	0.0434
	0.70	0.0040	0.0340	0.0340	-0.0008	0.0410	0.0410
	0.90	0.0067	0.0341	0.0341	-0.0011	0.0405	0.0405
	1.00	-0.0004	0.0388	0.0388	0.0034	0.0367	0.0367
	1.50	-0.0019	0.0430	0.0431	-0.0011	0.0412	0.0412
	2.00	-0.0019	0.0458	0.0458	-0.0029	0.0449	0.0449
	2.50	-0.0003	0.0427	0.0427	-0.0017	0.0439	0.0439
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9940	0.0507	0.0508	-0.0015	0.0543	0.0543
	0.30	1.0065	0.0500	0.0500	0.0018	0.0415	0.0415
	0.50	1.0011	0.0411	0.0411	-0.0058	0.0456	0.0457
	0.70	0.9990	0.0398	0.0398	0.0028	0.0383	0.0383
	0.90	1.0028	0.0430	0.0430	0.0024	0.0377	0.0377
	1.00	0.9925	0.0310	0.0311	0.0031	0.0359	0.0360
	1.50	1.0003	0.0408	0.0408	-0.0047	0.0435	0.0436
	2.00	0.9952	0.0390	0.0390	-0.0006	0.0415	0.0415
	2.50	1.0046	0.0473	0.0473	-0.0015	0.0445	0.0446

Table C.3: Means, variances, and MSEs of the $NED(a)$ estimates for the population means. Populations are $0.9N(\mu_i, 1) + 0.1[N(0,1)/U(0,1)]$, $i = 1, 2$. *(continued)*

(n_1, n_2)=(50,50)	a	Mean_1	Var_1	MSE_1	Mean_2	Var_2	MSE_2
$\mu_1 = 0, \mu_2 = 0$	0.15	-0.0032	0.0350	0.0350	0.0029	0.0275	0.0275
	0.30	0.0004	0.0258	0.0258	0.0009	0.0240	0.0240
	0.50	-0.0019	0.0259	0.0259	-0.0007	0.0251	0.0251
	0.70	0.0008	0.0232	0.0232	0.0022	0.0212	0.0212
	0.90	-0.0043	0.0282	0.0282	0.0014	0.0221	0.0221
	1.00	-0.0009	0.0243	0.0243	0.0027	0.0218	0.0218
	1.50	-0.0016	0.0249	0.0249	0.0036	0.0215	0.0215
	2.00	0.0021	0.0224	0.0224	0.0065	0.0180	0.0181
	2.50	0.0030	0.0208	0.0208	0.0001	0.0240	0.0240
$\mu_1 = 1, \mu_2 = 0$	0.15	0.9954	0.0290	0.0291	0.0017	0.0295	0.0295
	0.30	1.0018	0.0271	0.0271	0.0023	0.0234	0.0234
	0.50	1.0011	0.0260	0.0260	-0.0033	0.0279	0.0280
	0.70	1.0007	0.0247	0.0247	0.0009	0.0233	0.0233
	0.90	0.9987	0.0222	0.0222	-0.0015	0.0241	0.0241
	1.00	1.0009	0.0250	0.0250	0.0018	0.0212	0.0212
	1.50	0.9984	0.0222	0.0222	0.0022	0.0225	0.0226
	2.00	0.9991	0.0239	0.0239	-0.0005	0.0250	0.0250
	2.50	0.9973	0.0225	0.0225	0.0014	0.0231	0.0231

APPENDIX D

AVERAGES OF THE *NED(a)* BASED POPULATION VARIANCES ESTIMATES

Table D.1: Averages of the $NED(a)$ based population variances estimates. Sample sizes are (n_1, n_2)=(10,10), (30,30) and (50,50). Number of replications is 5000.

$N(0,1)$

	$\mu_1 = 0, \mu_2 = 0$			$\mu_1 = 1, \mu_2 = 0$		
a	(10, 10)	(30,30)	(50,50)	(10,10)	(30,30)	(50,50)
0.15	0.9631	1.0160	1.0184	0.9620	1.0171	1.0183
0.30	0.9126	0.9912	1.0003	0.9131	0.9912	1.0003
0.50	0.8482	0.9627	0.9803	0.8484	0.9627	0.9803
0.70	0.7988	0.9403	0.9644	0.7988	0.9403	0.9645
0.90	0.7599	0.9217	0.9515	0.7601	0.9217	0.9516
1.00	0.7444	0.9134	0.9458	0.7444	0.9135	0.9458
1.50	0.6871	0.8805	0.9225	0.6872	0.8805	0.9226
2.00	0.6505	0.8569	0.9048	0.6505	0.8570	0.9048
2.50	0.6272	0.8391	0.8909	0.6273	0.8390	0.8909

Table D.1: *(continued)*

$0.9N(0,1) + 0.1t(1)$

	$\mu_1 = 0, \mu_2 = 0$			$\mu_1 = 1, \mu_2 = 0$		
a	(10, 10)	(30,30)	(50,50)	(10,10)	(30,30)	(50,50)
0.15	4.0958	5.2891	5.8598	4.0906	5.2811	5.8485
0.30	3.0027	3.5358	3.8670	3.0060	3.5412	3.8751
0.50	2.3695	2.7090	2.9626	2.3715	2.7098	2.9639
0.70	2.0583	2.3255	2.5380	2.0570	2.3259	2.5391
0.90	1.8609	2.0995	2.2969	1.8627	2.0979	2.2968
1.00	1.7938	2.0148	2.2119	1.7928	2.0162	2.2120
1.50	1.5956	1.7780	1.9501	1.5981	1.7773	1.9504
2.00	1.5058	1.6569	1.8100	1.5045	1.6575	1.8101
2.50	1.4550	1.5835	1.7187	1.4562	1.5841	1.7190

Table D.1: *(continued)*

$0.9N(0,1) + 0.1[N(0,1)/U(0,1)]$

	$\mu_1 = 0, \mu_2 = 0$			$\mu_1 = 1, \mu_2 = 0$		
a	(10, 10)	(30,30)	(50,50)	(10,10)	(30,30)	(50,50)
0.15	1.1467	1.2590	1.2957	1.1470	1.2709	1.2882
0.30	1.0426	1.1455	1.1725	1.0263	1.1479	1.1702
0.50	0.9449	1.0808	1.1095	0.9354	1.0785	1.1151
0.70	0.8727	1.0436	1.0738	0.8763	1.0451	1.0781
0.90	0.8390	1.0142	1.0506	0.8368	1.0113	1.0550
1.00	0.8143	0.9940	1.0476	0.8101	1.0054	1.0443
1.50	0.7481	0.9577	1.0124	0.7479	0.9590	1.0121
2.00	0.7130	0.9357	0.9926	0.7125	0.9288	0.9874
2.50	0.6943	0.9127	0.9777	0.6965	0.9129	0.9759

APPENDIX E

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF TWO POPULATION VARIANCES

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$.

(n_1, n_2)=(40,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.233	0.046	0.237	0.191
	BF	0.043	0.043	0.042	0.055
	HDBF	0.047	0.051	0.047	0.086
(1:2)	F	0.552	0.561	0.546	0.550
	BF	0.346	0.463	0.321	0.353
	HDBF	0.362	0.495	0.337	0.408
(1:4)	F	0.921	0.990	0.913	0.942
	BF	0.874	0.968	0.854	0.879
	HDBF	0.880	0.973	0.862	0.897
(1:8)	F	0.997	1.000	0.993	0.998
	BF	0.996	1.000	0.992	0.997
	HDBF	0.996	1.000	0.993	0.997

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2)=(10,10)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.154	0.055	0.152	0.142
	BF	0.038	0.035	0.036	0.078
	HDBF	0.054	0.054	0.053	0.090
(1:2)	F	0.256	0.151	0.258	0.238
	BF	0.083	0.100	0.087	0.140
	HDBF	0.111	0.144	0.115	0.156
(1:4)	F	0.502	0.510	0.501	0.509
	BF	0.232	0.315	0.235	0.331
	HDBF	0.298	0.405	0.289	0.343
(1:8)	F	0.762	0.834	0.761	0.768
	BF	0.489	0.593	0.450	0.555
	HDBF	0.551	0.681	0.519	0.554

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2)=(10,20)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0:1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.181	0.051	0.163	0.151
	BF	0.040	0.044	0.040	0.067
	HDBF	0.049	0.057	0.053	0.084
(1:2)	F	0.304	0.190	0.314	0.295
	BF	0.077	0.113	0.082	0.121
	HDBF	0.107	0.159	0.111	0.163
(1:4)	F	0.619	0.603	0.604	0.608
	BF	0.277	0.399	0.251	0.320
	HDBF	0.338	0.487	0.306	0.392
(1:8)	F	0.845	0.937	0.862	0.872
	BF	0.556	0.748	0.537	0.610
	HDBF	0.618	0.823	0.596	0.664

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2)=(10,20)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0:1)$	$t(4)$	$\chi^2(4)$
(8:1)	F	0.840	0.931	0.845	0.859
	BF	0.754	0.839	0.738	0.776
	HDBF	0.766	0.864	0.752	0.764
(4:1)	F	0.586	0.664	0.588	0.599
	BF	0.445	0.543	0.433	0.480
	HDBF	0.467	0.584	0.454	0.467
(2:1)	F	0.280	0.216	0.307	0.278
	BF	0.152	0.175	0.166	0.198
	HDBF	0.167	0.203	0.177	0.195

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2)=(20,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.224	0.052	0.220	0.182
	BF	0.043	0.041	0.045	0.056
	HDBF	0.049	0.051	0.052	0.083
(1:2)	F	0.463	0.376	0.467	0.447
	BF	0.200	0.295	0.181	0.213
	HDBF	0.217	0.322	0.200	0.274
(1:4)	F	0.834	0.934	0.835	0.849
	BF	0.658	0.843	0.624	0.665
	HDBF	0.680	0.865	0.644	0.716
(1:8)	F	0.977	1.000	0.977	0.988
	BF	0.938	0.996	0.922	0.948
	HDBF	0.944	0.997	0.927	0.958

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.1: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2)=(20,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0:1)$	$t(4)$	$\chi^2(4)$
(8:1)	F	0.979	0.999	0.974	0.984
	BF	0.969	0.995	0.966	0.966
	HDBF	0.973	0.996	0.968	0.970
(4:1)	F	0.818	0.927	0.825	0.843
	BF	0.753	0.865	0.735	0.752
	HDBF	0.764	0.882	0.752	0.766
(2:1)	F	0.433	0.410	0.444	0.421
	BF	0.273	0.340	0.265	0.272
	HDBF	0.290	0.370	0.280	0.308

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$.

(n_1, n_2)=(40,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0:1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.117	0.010	0.124	0.085
	BF	0.007	0.009	0.007	0.009
	HDBF	0.008	0.011	0.009	0.022
(1:2)	F	0.403	0.316	0.397	0.387
	BF	0.136	0.219	0.133	0.151
	HDBF	0.150	0.249	0.148	0.206
(1:4)	F	0.838	0.954	0.841	0.874
	BF	0.666	0.862	0.631	0.679
	HDBF	0.688	0.884	0.655	0.729
(1:8)	F	0.989	1.000	0.984	0.994
	BF	0.970	0.999	0.954	0.967
	HDBF	0.972	0.999	0.958	0.972

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2)=(10,10)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.053	0.011	0.061	0.050
	BF	0.005	0.007	0.006	0.025
	HDBF	0.009	0.012	0.009	0.026
(1:2)	F	0.122	0.047	0.124	0.108
	BF	0.014	0.020	0.018	0.046
	HDBF	0.026	0.039	0.028	0.054
(1:4)	F	0.306	0.239	0.304	0.306
	BF	0.058	0.095	0.062	0.134
	HDBF	0.099	0.154	0.096	0.150
(1:8)	F	0.587	0.618	0.581	0.582
	BF	0.172	0.246	0.150	0.259
	HDBF	0.252	0.363	0.214	0.296

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2)=(10,20)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.074	0.010	0.071	0.056
	BF	0.007	0.007	0.005	0.018
	HDBF	0.009	0.012	0.009	0.023
(1:2)	F	0.149	0.053	0.155	0.135
	BF	0.011	0.017	0.011	0.028
	HDBF	0.019	0.033	0.018	0.050
(1:4)	F	0.400	0.308	0.383	0.386
	BF	0.058	0.110	0.049	0.096
	HDBF	0.094	0.173	0.077	0.148
(1:8)	F	0.697	0.750	0.708	0.715
	BF	0.179	0.326	0.163	0.255
	HDBF	0.252	0.445	0.224	0.342

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2)=(10,20)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(8:1)	F	0.729	0.828	0.734	0.747
	BF	0.476	0.614	0.465	0.564
	HDBF	0.523	0.683	0.506	0.549
(4:1)	F	0.423	0.439	0.435	0.433
	BF	0.200	0.293	0.207	0.274
	HDBF	0.235	0.352	0.236	0.269
(2:1)	F	0.161	0.073	0.173	0.155
	BF	0.047	0.051	0.054	0.087
	HDBF	0.060	0.075	0.067	0.087

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2)=(20,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1)	F	0.111	0.013	0.105	0.077
	BF	0.008	0.009	0.006	0.011
	HDBF	0.010	0.011	0.008	0.023
(1:2)	F	0.289	0.161	0.291	0.265
	BF	0.044	0.086	0.040	0.063
	HDBF	0.053	0.111	0.052	0.105
(1:4)	F	0.707	0.767	0.707	0.706
	BF	0.299	0.540	0.275	0.334
	HDBF	0.331	0.594	0.307	0.430
(1:8)	F	0.940	0.995	0.945	0.962
	BF	0.713	0.939	0.673	0.723
	HDBF	0.741	0.956	0.699	0.789

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table E.2: Empirical levels and powers: Tests for equality of two population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2)=(20,40)

$(\sigma_1^2 : \sigma_2^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(8:1)	F	0.948	0.996	0.943	0.960
	BF	0.886	0.971	0.868	0.886
	HDBF	0.896	0.977	0.881	0.895
(4:1)	F	0.699	0.820	0.710	0.723
	BF	0.508	0.686	0.498	0.526
	HDBF	0.539	0.724	0.530	0.564
(2:1)	F	0.294	0.211	0.307	0.272
	BF	0.111	0.151	0.106	0.117
	HDBF	0.127	0.182	0.123	0.151

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

APPENDIX F

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF THREE POPULATION VARIANCES

Table F.1: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.05$.

(n_1, n_2)=(40,40,40)

$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.333	0.054	0.345	0.258
	BF	0.041	0.047	0.041	0.058
	HDBF	0.047	0.060	0.047	0.095
(1:1:2)	BART	0.664	0.605	0.666	0.635
	BF	0.367	0.512	0.343	0.387
	HDBF	0.388	0.546	0.363	0.455
(1:1:4)	BART	0.962	0.994	0.962	0.978
	BF	0.928	0.984	0.899	0.932
	HDBF	0.935	0.987	0.907	0.945
(1:2:4)	BART	0.916	0.978	0.916	0.928
	BF	0.795	0.934	0.766	0.805
	HDBF	0.814	0.942	0.780	0.846

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(10,10,10)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.221	0.051	0.213	0.189
	BF	0.037	0.034	0.039	0.091
	HDBF	0.057	0.061	0.057	0.100
(1:1:2)	BART	0.315	0.159	0.327	0.313
	BF	0.083	0.113	0.083	0.162
	HDBF	0.118	0.158	0.115	0.168
(1:1:4)	BART	0.584	0.546	0.589	0.576
	BF	0.281	0.365	0.276	0.373
	HDBF	0.342	0.455	0.328	0.369
(1:2:4)	BART	0.489	0.389	0.491	0.490
	BF	0.178	0.235	0.168	0.282
	HDBF	0.232	0.323	0.220	0.293

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(10,10,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.246	0.053	0.250	0.206
	BF	0.040	0.042	0.052	0.077
	HDBF	0.046	0.057	0.059	0.092
(1:1:2)	BART	0.466	0.269	0.465	0.434
	BF	0.093	0.170	0.089	0.126
	HDBF	0.121	0.228	0.113	0.203
(1:1:4)	BART	0.829	0.841	0.833	0.826
	BF	0.403	0.660	0.386	0.459
	HDBF	0.460	0.741	0.445	0.573
(1:2:4)	BART	0.718	0.612	0.692	0.694
	BF	0.237	0.415	0.214	0.281
	HDBF	0.293	0.506	0.263	0.389

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(10,10,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.627	0.702	0.643	0.655
	BF	0.490	0.617	0.484	0.538
	HDBF	0.504	0.640	0.497	0.523
(4:1:1)	BART	0.650	0.692	0.672	0.647
	BF	0.495	0.606	0.494	0.521
	HDBF	0.504	0.633	0.511	0.515
(2:1:1)	BART	0.340	0.215	0.355	0.335
	BF	0.164	0.185	0.171	0.209
	HDBF	0.175	0.211	0.179	0.211

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.307	0.047	0.293	0.248
	BF	0.043	0.041	0.043	0.055
	HDBF	0.052	0.055	0.053	0.097
(1:1:2)	BART	0.593	0.447	0.583	0.555
	BF	0.223	0.332	0.213	0.251
	HDBF	0.247	0.378	0.234	0.324
(1:1:4)	BART	0.927	0.973	0.929	0.932
	BF	0.764	0.920	0.735	0.781
	HDBF	0.786	0.937	0.755	0.822
(1:2:4)	BART	0.834	0.880	0.846	0.842
	BF	0.538	0.760	0.527	0.574
	HDBF	0.563	0.802	0.559	0.647

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.805	0.892	0.814	0.827
	BF	0.634	0.796	0.622	0.670
	HDBF	0.666	0.826	0.646	0.704
(4:1:1)	BART	0.851	0.910	0.858	0.859
	BF	0.708	0.846	0.686	0.735
	HDBF	0.735	0.872	0.707	0.759
(2:1:1)	BART	0.495	0.387	0.504	0.483
	BF	0.225	0.307	0.212	0.255
	HDBF	0.249	0.347	0.230	0.299

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.282	0.046	0.263	0.226
	BF	0.046	0.044	0.039	0.062
	HDBF	0.054	0.059	0.046	0.095
(1:1:2)	BART	0.544	0.376	0.534	0.498
	BF	0.159	0.272	0.149	0.195
	HDBF	0.184	0.326	0.172	0.270
(1:1:4)	BART	0.894	0.946	0.899	0.895
	BF	0.643	0.848	0.616	0.669
	HDBF	0.680	0.884	0.648	0.733
(1:2:4)	BART	0.746	0.669	0.736	0.714
	BF	0.312	0.490	0.277	0.342
	HDBF	0.361	0.563	0.319	0.445

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.1: Empirical levels and powers: Tests equality of three population variances with $\alpha = 0.05$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.675	0.727	0.684	0.685
	BF	0.496	0.633	0.497	0.543
	HDBF	0.515	0.668	0.512	0.552
(4:1:1)	BART	0.669	0.707	0.684	0.666
	BF	0.520	0.641	0.506	0.547
	HDBF	0.529	0.663	0.513	0.549
(2:1:1)	BART	0.398	0.229	0.394	0.357
	BF	0.169	0.208	0.172	0.203
	HDBF	0.175	0.222	0.177	0.215

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$.

(n_1, n_2)=(40,40,40)

(σ_1^2, σ_2^2)		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.177	0.011	0.211	0.128
	BF	0.007	0.009	0.007	0.010
	HDBF	0.009	0.013	0.009	0.028
(1:1:2)	BART	0.505	0.377	0.501	0.475
	BF	0.165	0.281	0.161	0.191
	HDBF	0.182	0.317	0.175	0.262
(1:1:4)	BART	0.921	0.979	0.928	0.940
	BF	0.785	0.940	0.757	0.808
	HDBF	0.800	0.950	0.771	0.844
(1:2:4)	BART	0.828	0.916	0.840	0.851
	BF	0.551	0.801	0.528	0.574
	HDBF	0.578	0.833	0.553	0.659

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(10,10,10)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.093	0.011	0.094	0.076
	BF	0.007	0.006	0.005	0.027
	HDBF	0.011	0.012	0.012	0.034
(1:1:2)	BART	0.169	0.051	0.166	0.157
	BF	0.018	0.028	0.022	0.061
	HDBF	0.032	0.055	0.035	0.066
(1:1:4)	BART	0.387	0.318	0.393	0.394
	BF	0.103	0.154	0.095	0.183
	HDBF	0.151	0.237	0.138	0.203
(1:2:4)	BART	0.293	0.174	0.304	0.297
	BF	0.048	0.069	0.050	0.117
	HDBF	0.087	0.127	0.079	0.135

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(10,10,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.114	0.008	0.121	0.088
	BF	0.008	0.008	0.013	0.021
	HDBF	0.012	0.012	0.015	0.025
(1:1:2)	BART	0.258	0.100	0.256	0.235
	BF	0.012	0.033	0.013	0.032
	HDBF	0.021	0.060	0.020	0.067
(1:1:4)	BART	0.656	0.598	0.658	0.649
	BF	0.127	0.305	0.113	0.169
	HDBF	0.168	0.395	0.155	0.284
(1:2:4)	BART	0.491	0.315	0.480	0.469
	BF	0.052	0.129	0.040	0.075
	HDBF	0.076	0.190	0.060	0.148

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(10,10,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.483	0.502	0.491	0.501
	BF	0.282	0.399	0.274	0.346
	HDBF	0.303	0.442	0.297	0.336
(4:1:1)	BART	0.495	0.509	0.516	0.492
	BF	0.303	0.418	0.304	0.352
	HDBF	0.325	0.460	0.324	0.350
(2:1:1)	BART	0.212	0.087	0.222	0.190
	BF	0.062	0.079	0.066	0.094
	HDBF	0.068	0.093	0.075	0.093

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.163	0.013	0.160	0.114
	BF	0.008	0.007	0.006	0.014
	HDBF	0.011	0.012	0.009	0.028
(1:1:2)	BART	0.400	0.218	0.409	0.373
	BF	0.069	0.131	0.067	0.085
	HDBF	0.084	0.162	0.083	0.148
(1:1:4)	BART	0.845	0.905	0.848	0.854
	BF	0.479	0.742	0.447	0.515
	HDBF	0.520	0.792	0.483	0.602
(1:2:4)	BART	0.702	0.693	0.709	0.700
	BF	0.234	0.456	0.231	0.289
	HDBF	0.277	0.530	0.266	0.390

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.691	0.750	0.701	0.715
	BF	0.387	0.584	0.371	0.431
	HDBF	0.423	0.644	0.410	0.490
(4:1:1)	BART	0.739	0.805	0.749	0.756
	BF	0.491	0.672	0.461	0.534
	HDBF	0.526	0.720	0.496	0.580
(2:1:1)	BART	0.340	0.192	0.343	0.319
	BF	0.085	0.138	0.080	0.117
	HDBF	0.101	0.173	0.095	0.158

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(1:1:1)	BART	0.138	0.010	0.140	0.101
	BF	0.009	0.006	0.009	0.016
	HDBF	0.013	0.009	0.011	0.025
(1:1:2)	BART	0.342	0.167	0.347	0.305
	BF	0.035	0.085	0.033	0.060
	HDBF	0.047	0.119	0.044	0.109
(1:1:4)	BART	0.785	0.814	0.779	0.783
	BF	0.315	0.575	0.289	0.355
	HDBF	0.362	0.653	0.335	0.463
(1:2:4)	BART	0.552	0.395	0.534	0.514
	BF	0.091	0.199	0.077	0.119
	HDBF	0.119	0.260	0.097	0.194

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

Table F.2: Empirical levels and powers: Tests for equality of three population variances with $\alpha = 0.01$. *(continued)*

(n_1, n_2, n_3)=(20,20,40)

$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2)$		Mixed*	$N(0,1)$	$t(4)$	$\chi^2(4)$
(4:2:1)	BART	0.535	0.526	0.544	0.529
	BF	0.277	0.399	0.292	0.347
	HDBF	0.301	0.449	0.309	0.353
(4:1:1)	BART	0.520	0.538	0.525	0.515
	BF	0.336	0.461	0.326	0.371
	HDBF	0.347	0.488	0.332	0.371
(2:1:1)	BART	0.242	0.100	0.248	0.213
	BF	0.071	0.097	0.072	0.100
	HDBF	0.074	0.111	0.075	0.105

$*$: Mixed $= 0.9N(0.1) + 0.1N(0.8)$.

APPENDIX G

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF THREE POPULATION MEANS

Table G.1: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.05$.

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(11,11,11)				
(0,0,0)	ANOVA	0.047	0.048	0.045	0.047	0.052
	BF	0.047	0.044	0.041	0.044	0.051
	HDBF	0.052	0.052	0.083	0.054	0.063
	TanBF	0.056	0.058	0.051	0.060	0.058
(-1.70,0.54,1.89)	ANOVA	0.527	0.544	0.560	0.590	0.517
	BF	0.515	0.532	0.545	0.568	0.506
	HDBF	0.521	0.560	0.633	0.634	0.508
	TanBF	0.507	0.551	0.562	0.671	0.409
(2.60,-0.80,-2.90)	ANOVA	0.900	0.892	0.895	0.897	0.899
	BF	0.893	0.880	0.867	0.890	0.894
	HDBF	0.896	0.893	0.894	0.928	0.893
	TanBF	0.870	0.901	0.890	0.944	0.794
(2.50,1.67,-5.83)	ANOVA	0.999	0.997	0.992	0.990	1.000
	BF	0.999	0.995	0.985	0.988	1.000
	HDBF	0.999	0.997	0.990	0.997	1.000
	TanBF	0.999	0.997	0.987	0.997	0.998

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2q)$.

Table G.1: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.05$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(71,8,16)				
(0,0,0)	ANOVA	0.046	0.040	0.067	0.044	0.050
	BF	0.058	0.051	0.074	0.054	0.063
	HDBF	0.062	0.056	0.120	0.065	0.068
	TanBF	0.069	0.069	0.093	0.070	0.068
(-1.70,0.54,1.89)	ANOVA	0.187	0.187	0.276	0.219	0.167
	BF	0.217	0.221	0.298	0.245	0.195
	HDBF	0.224	0.234	0.373	0.281	0.199
	TanBF	0.227	0.240	0.350	0.300	0.169
(2.60,-0.80,-2.90)	ANOVA	0.371	0.397	0.366	0.430	0.348
	BF	0.424	0.441	0.426	0.470	0.403
	HDBF	0.431	0.462	0.481	0.519	0.403
	TanBF	0.424	0.472	0.435	0.544	0.327
(2.50,1.67,-5.83)	ANOVA	0.811	0.397	0.855	0.826	0.819
	BF	0.858	0.441	0.879	0.855	0.871
	HDBF	0.860	0.462	0.911	0.901	0.871
	TanBF	0.835	0.472	0.899	0.920	0.761

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2q)$.

Table G.1: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.05$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(12,56,16)				
(0,0,0)	ANOVA	0.067	0.065	0.075	0.059	0.070
	BF	0.063	0.053	0.063	0.047	0.063
	HDBF	0.068	0.059	0.108	0.058	0.069
	TanBF	0.069	0.068	0.090	0.062	0.066
(-1.70,0.54,1.89)	ANOVA	0.237	0.260	0.261	0.285	0.231
	BF	0.208	0.227	0.215	0.255	0.205
	HDBF	0.219	0.249	0.302	0.305	0.210
	TanBF	0.217	0.260	0.225	0.319	0.170
(2.60,-0.80,-2.90)	ANOVA	0.545	0.564	0.590	0.593	0.535
	BF	0.493	0.516	0.540	0.548	0.489
	HDBF	0.501	0.539	0.609	0.609	0.492
	TanBF	0.487	0.542	0.594	0.645	0.375
(2.50,1.67,-5.83)	ANOVA	0.931	0.926	0.928	0.920	0.944
	BF	0.909	0.900	0.893	0.894	0.921
	HDBF	0.910	0.912	0.930	0.932	0.920
	TanBF	0.881	0.909	0.910	0.946	0.789

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.2: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.01$.

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(11,11,11)				
(0,0,0)	ANOVA	0.007	0.008	0.008	0.010	0.052
	BF	0.007	0.009	0.006	0.007	0.051
	HDBF	0.009	0.011	0.016	0.011	0.063
	TanBF	0.012	0.013	0.010	0.013	0.058
(-1.70,0.54,1.89)	ANOVA	0.278	0.294	0.331	0.348	0.517
	BF	0.261	0.266	0.280	0.325	0.506
	HDBF	0.271	0.290	0.374	0.384	0.508
	TanBF	0.254	0.285	0.276	0.405	0.409
(2.60,-0.80,-2.90)	ANOVA	0.716	0.718	0.741	0.753	0.899
	BF	0.690	0.690	0.696	0.723	0.894
	HDBF	0.697	0.714	0.754	0.789	0.893
	TanBF	0.637	0.709	0.722	0.799	0.794
(2.50,1.67,-5.83)	ANOVA	0.992	0.985	0.974	0.974	1.000
	BF	0.989	0.976	0.954	0.964	1.000
	HDBF	0.989	0.982	0.965	0.983	1.000
	TanBF	0.975	0.979	0.957	0.987	0.998

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.2: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.01$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(71,8,16)				
(0,0,0)	ANOVA	0.012	0.010	0.026	0.013	0.050
	BF	0.015	0.013	0.026	0.015	0.063
	HDBF	0.017	0.014	0.055	0.020	0.068
	TanBF	0.017	0.017	0.035	0.019	0.068
(-1.70,0.54,1.89)	ANOVA	0.082	0.072	0.158	0.088	0.167
	BF	0.096	0.087	0.166	0.102	0.195
	HDBF	0.100	0.098	0.240	0.124	0.199
	TanBF	0.098	0.098	0.210	0.129	0.169
(2.60,-0.80,-2.90)	ANOVA	0.194	0.201	0.136	0.227	0.348
	BF	0.221	0.238	0.170	0.250	0.403
	HDBF	0.227	0.253	0.220	0.300	0.403
	TanBF	0.217	0.255	0.168	0.313	0.327
(2.50,1.67,-5.83)	ANOVA	0.527	0.201	0.617	0.613	0.819
	BF	0.585	0.238	0.660	0.649	0.871
	HDBF	0.594	0.253	0.743	0.722	0.871
	TanBF	0.555	0.255	0.688	0.742	0.761

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.2: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(12,10,8) and $\alpha = 0.01$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(12,56,16)				
(0,0,0)	ANOVA	0.020	0.016	0.023	0.012	0.070
	BF	0.016	0.011	0.019	0.009	0.063
	HDBF	0.018	0.014	0.047	0.014	0.069
	TanBF	0.022	0.018	0.033	0.012	0.066
(-1.70,0.54,1.89)	ANOVA	0.082	0.091	0.090	0.109	0.231
	BF	0.062	0.067	0.057	0.084	0.205
	HDBF	0.071	0.078	0.103	0.114	0.210
	TanBF	0.071	0.081	0.063	0.118	0.170
(2.60,-0.80,-2.90)	ANOVA	0.250	0.278	0.339	0.321	0.535
	BF	0.197	0.227	0.298	0.261	0.489
	HDBF	0.207	0.245	0.369	0.316	0.492
	TanBF	0.197	0.251	0.339	0.334	0.375
(2.50,1.67,-5.83)	ANOVA	0.734	0.751	0.763	0.775	0.944
	BF	0.651	0.671	0.675	0.706	0.921
	HDBF	0.660	0.694	0.756	0.774	0.920
	TanBF	0.598	0.668	0.689	0.781	0.789

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2q)$.

Table G.3: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.05$.

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(11,11,11)				
(0,0,0)	ANOVA	0.053	0.049	0.045	0.047	0.051
	BF	0.053	0.047	0.045	0.046	0.052
	HDBF	0.055	0.053	0.101	0.055	0.053
	TanBF	0.056	0.056	0.052	0.057	0.042
(-1.70,0.54,1.89)	ANOVA	0.860	0.865	0.862	0.869	0.865
	BF	0.858	0.865	0.870	0.866	0.865
	HDBF	0.861	0.891	0.913	0.919	0.864
	TanBF	0.847	0.890	0.884	0.933	0.739
(2.60,-0.80,-2.90)	ANOVA	0.998	0.997	0.995	0.988	0.999
	BF	0.997	0.996	0.991	0.987	0.999
	HDBF	0.998	0.997	0.994	0.999	0.999
	TanBF	0.995	0.998	0.995	1.000	0.991
(2.50,1.67,-5.83)	ANOVA	1.000	1.000	1.000	1.000	1.000
	BF	1.000	1.000	1.000	0.999	1.000
	HDBF	1.000	1.000	1.000	1.000	1.000
	TanBF	1.000	1.000	1.000	1.000	1.000

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2q)$.

Table G.3: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.05$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(71,8,16)				
(0,0,0)	ANOVA	0.044	0.041	0.053	0.042	0.048
	BF	0.061	0.056	0.067	0.059	0.064
	HDBF	0.064	0.060	0.128	0.071	0.065
	TanBF	0.066	0.063	0.090	0.075	0.053
(-1.70,0.54,1.89)	ANOVA	0.346	0.361	0.385	0.387	0.324
	BF	0.399	0.416	0.429	0.439	0.388
	HDBF	0.404	0.438	0.543	0.500	0.389
	TanBF	0.393	0.442	0.512	0.529	0.302
(2.60,-0.80,-2.90)	ANOVA	0.670	0.682	0.733	0.706	0.663
	BF	0.731	0.737	0.781	0.751	0.726
	HDBF	0.734	0.762	0.810	0.822	0.727
	TanBF	0.715	0.763	0.788	0.844	0.612
(2.50,1.67,-5.83)	ANOVA	0.996	0.990	0.995	0.981	0.997
	BF	0.999	0.995	0.996	0.987	0.999
	HDBF	0.999	0.998	0.999	0.999	0.999
	TanBF	0.997	0.997	0.998	0.999	0.992

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.3: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.05$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(12,56,16)				
(0,0,0)	ANOVA	0.058	0.063	0.078	0.059	0.064
	BF	0.050	0.057	0.074	0.051	0.058
	HDBF	0.052	0.063	0.136	0.064	0.058
	TanBF	0.057	0.063	0.092	0.071	0.049
(-1.70,0.54,1.89)	ANOVA	0.508	0.522	0.524	0.553	0.486
	BF	0.480	0.493	0.494	0.527	0.458
	HDBF	0.488	0.530	0.618	0.612	0.457
	TanBF	0.466	0.527	0.496	0.643	0.313
(2.60,-0.80,-2.90)	ANOVA	0.907	0.906	0.884	0.904	0.910
	BF	0.895	0.890	0.861	0.892	0.899
	HDBF	0.898	0.915	0.919	0.948	0.899
	TanBF	0.871	0.917	0.913	0.959	0.762
(2.50,1.67,-5.83)	ANOVA	0.999	1.000	0.999	0.995	1.000
	BF	0.999	0.999	0.998	0.993	1.000
	HDBF	0.999	1.000	0.999	0.999	1.000
	TanBF	0.999	1.000	0.999	1.000	0.997

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2q)$.

Table G.4: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.01$.

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(11,11,11)				
(0,0,0)	ANOVA	0.009	0.008	0.010	0.011	0.012
	BF	0.009	0.008	0.009	0.009	0.012
	HDBF	0.010	0.010	0.026	0.012	0.012
	TanBF	0.011	0.011	0.012	0.013	0.011
(-1.70,0.54,1.89)	ANOVA	0.669	0.683	0.689	0.708	0.664
	BF	0.660	0.678	0.688	0.699	0.656
	HDBF	0.665	0.715	0.786	0.795	0.657
	TanBF	0.631	0.709	0.710	0.820	0.470
(2.60,-0.80,-2.90)	ANOVA	0.984	0.982	0.974	0.970	0.989
	BF	0.983	0.977	0.964	0.965	0.988
	HDBF	0.984	0.986	0.980	0.992	0.988
	TanBF	0.975	0.987	0.978	0.994	0.941
(2.50,1.67,-5.83)	ANOVA	1.000	1.000	1.000	0.999	1.000
	BF	1.000	1.000	0.999	0.998	1.000
	HDBF	1.000	1.000	0.999	1.000	1.000
	TanBF	1.000	1.000	1.000	1.000	1.000

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.4: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.01$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(71,8,16)				
(0,0,0)	ANOVA	0.012	0.008	0.022	0.010	0.014
	BF	0.016	0.016	0.025	0.015	0.018
	HDBF	0.017	0.017	0.058	0.018	0.018
	TanBF	0.020	0.016	0.036	0.022	0.017
(-1.70,0.54,1.89)	ANOVA	0.172	0.185	0.239	0.205	0.158
	BF	0.219	0.228	0.270	0.249	0.206
	HDBF	0.222	0.247	0.393	0.299	0.206
	TanBF	0.213	0.245	0.343	0.318	0.136
(2.60,-0.80,-2.90)	ANOVA	0.453	0.474	0.454	0.505	0.437
	BF	0.523	0.538	0.544	0.564	0.502
	HDBF	0.529	0.568	0.614	0.656	0.502
	TanBF	0.500	0.563	0.543	0.674	0.374
(2.50,1.67,-5.83)	ANOVA	0.956	0.949	0.968	0.935	0.967
	BF	0.980	0.971	0.978	0.954	0.984
	HDBF	0.980	0.981	0.991	0.991	0.984
	TanBF	0.966	0.977	0.988	0.991	0.924

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

Table G.4: Empirical levels and powers: Tests for equality of three population means for (n_1, n_2, n_3)=(24,20,16) and $\alpha = 0.01$. *(continued)*

(μ_1, μ_2, μ_3)		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2, \sigma_2^2, \sigma_3^2)$=(12,56,16)				
(0,0,0)	ANOVA	0.018	0.015	0.026	0.016	0.020
	BF	0.017	0.013	0.022	0.012	0.017
	HDBF	0.018	0.016	0.060	0.019	0.019
	TanBF	0.018	0.018	0.038	0.018	0.016
(-1.70,0.54,1.89)	ANOVA	0.224	0.237	0.237	0.276	0.198
	BF	0.199	0.207	0.197	0.245	0.168
	HDBF	0.208	0.231	0.314	0.312	0.170
	TanBF	0.193	0.228	0.197	0.338	0.100
(2.60,-0.80,-2.90)	ANOVA	0.710	0.706	0.692	0.729	0.691
	BF	0.664	0.666	0.651	0.692	0.646
	HDBF	0.670	0.705	0.762	0.792	0.647
	TanBF	0.620	0.700	0.732	0.820	0.418
(2.50,1.67,-5.83)	ANOVA	0.997	0.990	0.991	0.979	0.998
	BF	0.993	0.985	0.985	0.970	0.996
	HDBF	0.994	0.992	0.993	0.996	0.996
	TanBF	0.986	0.992	0.991	0.996	0.954

*: Mix $= pN(q\Delta, V_i) + qN(-p\Delta, r^2 V_i)$, where p=0.1, Δ=2, r^2=0.25 and $V_i = (\sigma_i^2 - pq\Delta^2)/(p + r^2 q)$.

APPENDIX H

EMPIRICAL LEVELS AND POWERS OF THE TESTS FOR EQUALITY OF FOUR POPULATION MEANS

Table H.1: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.05$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)						
(0,0,0,0)	ANOVA	0.050	0.024	0.045	0.047	0.050
	B-F	0.048	0.017	0.040	0.043	0.049
	HDBF	0.054	0.059	0.090	0.054	0.059
	TanBF	0.055	0.046	0.044	0.057	0.044
(0.5,0,0,0)	ANOVA	0.182	0.061	0.062	0.120	0.478
	B-F	0.177	0.048	0.056	0.113	0.475
	HDBF	0.188	0.155	0.118	0.141	0.479
	TanBF	0.183	0.127	0.065	0.145	0.371
(0,0,0,0.7)	ANOVA	0.329	0.087	0.092	0.194	0.800
	B-F	0.321	0.072	0.084	0.186	0.798
	HDBF	0.331	0.241	0.150	0.224	0.800
	TanBF	0.322	0.203	0.093	0.248	0.678

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.1: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)						
(0,0,0,0)	ANOVA	0.101	0.054	0.124	0.100	0.109
	B-F	0.082	0.037	0.108	0.078	0.089
	HDBF	0.087	0.092	0.163	0.092	0.093
	TanBF	0.088	0.072	0.144	0.088	0.091
(0.5,0,0,0)	ANOVA	0.131	0.066	0.105	0.106	0.169
	B-F	0.106	0.045	0.088	0.083	0.140
	HDBF	0.112	0.103	0.141	0.095	0.142
	TanBF	0.112	0.086	0.113	0.098	0.124
(0,0,0,0.7)	ANOVA	0.140	0.062	0.140	0.109	0.197
	B-F	0.110	0.043	0.121	0.084	0.152
	HDBF	0.119	0.112	0.178	0.097	0.155
	TanBF	0.121	0.092	0.156	0.102	0.137

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.1: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)						
(0,0,0,0)	ANOVA	0.066	0.038	0.080	0.058	0.072
	B-F	0.056	0.025	0.067	0.048	0.066
	HDBF	0.061	0.067	0.130	0.064	0.073
	TanBF	0.066	0.051	0.093	0.063	0.064
(0.5,0,0,0)	ANOVA	0.081	0.037	0.081	0.065	0.102
	B-F	0.072	0.027	0.071	0.053	0.090
	HDBF	0.079	0.078	0.127	0.068	0.097
	TanBF	0.080	0.062	0.084	0.072	0.082
(0,0,0,0.7)	ANOVA	0.106	0.047	0.071	0.089	0.169
	B-F	0.096	0.031	0.061	0.076	0.154
	HDBF	0.102	0.093	0.114	0.088	0.158
	TanBF	0.099	0.078	0.075	0.097	0.127

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.2: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)						
(0,0,0,0)	ANOVA	0.050	0.035	0.053	0.048	0.053
	B-F	0.048	0.030	0.048	0.042	0.051
	HDBF	0.059	0.064	0.108	0.053	0.055
	TanBF	0.052	0.062	0.055	0.061	0.045
(0.5,0,0,0)	ANOVA	0.199	0.067	0.076	0.134	0.534
	B-F	0.197	0.051	0.063	0.124	0.526
	HDBF	0.206	0.183	0.134	0.153	0.527
	TanBF	0.206	0.155	0.078	0.167	0.391
(0,0,0,0.7)	ANOVA	0.548	0.131	0.116	0.331	0.971
	B-F	0.540	0.120	0.124	0.322	0.968
	HDBF	0.551	0.413	0.224	0.379	0.969
	TanBF	0.525	0.373	0.158	0.414	0.914

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.2: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)						
(0,0,0,0)	ANOVA	0.183	0.123	0.189	0.165	0.185
	B-F	0.084	0.038	0.104	0.066	0.087
	HDBF	0.091	0.089	0.162	0.078	0.090
	TanBF	0.088	0.079	0.137	0.074	0.085
(0.5,0,0,0)	ANOVA	0.203	0.138	0.179	0.184	0.260
	B-F	0.094	0.044	0.092	0.083	0.130
	HDBF	0.099	0.109	0.146	0.099	0.133
	TanBF	0.096	0.091	0.113	0.095	0.114
(0,0,0,0.7)	ANOVA	0.277	0.160	0.210	0.235	0.534
	B-F	0.123	0.062	0.125	0.101	0.225
	HDBF	0.130	0.130	0.180	0.118	0.229
	TanBF	0.128	0.108	0.163	0.114	0.181

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.2: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:16)						
(0,0,0,0)	ANOVA	0.049	0.021	0.067	0.041	0.052
	B-F	0.082	0.048	0.098	0.078	0.087
	HDBF	0.085	0.089	0.168	0.090	0.089
	TanBF	0.093	0.085	0.134	0.095	0.077
(0.5,0,0,0)	ANOVA	0.059	0.030	0.069	0.049	0.080
	B-F	0.104	0.057	0.101	0.082	0.137
	HDBF	0.106	0.105	0.175	0.098	0.139
	TanBF	0.104	0.102	0.145	0.102	0.109
(0,0,0,0.7)	ANOVA	0.121	0.048	0.049	0.084	0.238
	B-F	0.172	0.081	0.086	0.125	0.323
	HDBF	0.176	0.160	0.140	0.144	0.323
	TanBF	0.168	0.155	0.100	0.157	0.253

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.2: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
	$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)					
(0,0,0,0)	ANOVA	0.036	0.019	0.043	0.032	0.118
	B-F	0.062	0.035	0.063	0.054	0.062
	HDBF	0.066	0.073	0.133	0.066	0.065
	TanBF	0.066	0.071	0.081	0.070	0.058
(0.5,0,0,0)	ANOVA	0.042	0.025	0.050	0.039	0.189
	B-F	0.066	0.039	0.073	0.064	0.106
	HDBF	0.071	0.087	0.147	0.075	0.111
	TanBF	0.066	0.078	0.101	0.078	0.093
(0,0,0,0.7)	ANOVA	0.086	0.036	0.037	0.061	0.322
	B-F	0.124	0.054	0.060	0.091	0.187
	HDBF	0.130	0.119	0.124	0.106	0.191
	TanBF	0.127	0.114	0.069	0.123	0.136

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.2: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:9:4:1)						
(0,0,0,0)	ANOVA	0.118	0.074	0.123	0.115	0.041
	B-F	0.064	0.030	0.072	0.054	0.067
	HDBF	0.067	0.071	0.140	0.066	0.069
	TanBF	0.066	0.060	0.094	0.070	0.055
(0.5,0,0,0)	ANOVA	0.138	0.089	0.111	0.122	0.058
	B-F	0.078	0.034	0.057	0.059	0.098
	HDBF	0.084	0.085	0.122	0.074	0.100
	TanBF	0.082	0.068	0.070	0.078	0.073
(0,0,0,0.7)	ANOVA	0.182	0.091	0.142	0.141	0.181
	B-F	0.096	0.034	0.080	0.069	0.243
	HDBF	0.101	0.104	0.155	0.091	0.244
	TanBF	0.102	0.089	0.115	0.099	0.180

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.3: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.05$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)						
(0,0,0,0)	ANOVA	0.047	0.037	0.050	0.050	0.047
	B-F	0.047	0.036	0.049	0.049	0.047
	HDBF	0.055	0.071	0.179	0.060	0.049
	TanBF	0.051	0.078	0.061	0.067	0.032
(0.5,0,0,0)	ANOVA	0.709	0.145	0.155	0.417	0.996
	B-F	0.709	0.144	0.155	0.417	0.996
	HDBF	0.723	0.645	0.344	0.539	0.996
	TanBF	0.686	0.561	0.189	0.552	0.984
(0,0,0,0.7)	ANOVA	0.959	0.256	0.276	0.721	1.000
	B-F	0.959	0.254	0.276	0.719	1.000
	HDBF	0.962	0.917	0.490	0.841	1.000
	TanBF	0.954	0.848	0.333	0.845	1.000

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.3: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)				
(0,0,0,0)	ANOVA	0.098	0.083	0.105	0.093	0.097
	B-F	0.093	0.077	0.100	0.089	0.090
	HDBF	0.100	0.111	0.245	0.104	0.092
	TanBF	0.094	0.105	0.168	0.103	0.069
(0.5,0,0,0)	ANOVA	0.201	0.104	0.096	0.157	0.419
	B-F	0.191	0.098	0.090	0.151	0.408
	HDBF	0.202	0.200	0.195	0.176	0.408
	TanBF	0.191	0.183	0.110	0.181	0.315
(0,0,0,0.7)	ANOVA	0.300	0.119	0.133	0.175	0.926
	B-F	0.280	0.110	0.126	0.166	0.910
	HDBF	0.302	0.267	0.306	0.214	0.916
	TanBF	0.269	0.228	0.227	0.223	0.701

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.3: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.05$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)				
(0,0,0,0)	ANOVA	0.067	0.058	0.063	0.064	0.071
	B-F	0.066	0.056	0.062	0.062	0.069
	HDBF	0.074	0.078	0.208	0.075	0.071
	TanBF	0.069	0.088	0.099	0.074	0.048
(0.5,0,0,0)	ANOVA	0.149	0.072	0.065	0.109	0.318
	B-F	0.146	0.070	0.063	0.106	0.314
	HDBF	0.153	0.151	0.187	0.130	0.317
	TanBF	0.142	0.141	0.075	0.144	0.227
(0,0,0,0.7)	ANOVA	0.202	0.077	0.093	0.124	0.670
	B-F	0.198	0.074	0.090	0.120	0.661
	HDBF	0.218	0.190	0.301	0.164	0.664
	TanBF	0.198	0.177	0.173	0.170	0.428

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.4: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.01$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)						
(0,0,0,0)	ANOVA	0.009	0.003	0.010	0.007	0.012
	B-F	0.008	0.002	0.008	0.007	0.011
	HDBF	0.009	0.013	0.018	0.009	0.020
	TanBF	0.011	0.005	0.010	0.010	0.012
(0.5,0,0,0)	ANOVA	0.060	0.012	0.015	0.033	0.237
	B-F	0.055	0.009	0.011	0.029	0.233
	HDBF	0.060	0.047	0.032	0.042	0.239
	TanBF	0.060	0.032	0.015	0.043	0.157
(0,0,0,0,0.7)	ANOVA	0.140	0.021	0.022	0.064	0.566
	B-F	0.133	0.015	0.019	0.057	0.559
	HDBF	0.147	0.096	0.043	0.083	0.562
	TanBF	0.130	0.059	0.021	0.092	0.405

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.4: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
		$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)				
(0,0,0,0)	ANOVA	0.046	0.013	0.067	0.041	0.054
	B-F	0.028	0.006	0.053	0.023	0.036
	HDBF	0.030	0.033	0.098	0.030	0.040
	TanBF	0.032	0.024	0.079	0.032	0.039
(0.5,0,0,0)	ANOVA	0.066	0.019	0.051	0.042	0.094
	B-F	0.043	0.012	0.040	0.025	0.064
	HDBF	0.048	0.042	0.078	0.033	0.068
	TanBF	0.046	0.033	0.056	0.035	0.057
(0,0,0,0,0.7)	ANOVA	0.065	0.014	0.078	0.046	0.095
	B-F	0.040	0.007	0.065	0.030	0.061
	HDBF	0.045	0.039	0.111	0.040	0.065
	TanBF	0.042	0.027	0.092	0.038	0.057

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.4: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,11,11,11) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
	$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)					
(0,0,0,0)	ANOVA	0.020	0.005	0.025	0.015	0.027
	B-F	0.014	0.002	0.019	0.012	0.020
	HDBF	0.018	0.017	0.051	0.018	0.026
	TanBF	0.016	0.013	0.025	0.018	0.020
(0.5,0,0,0)	ANOVA	0.027	0.007	0.026	0.015	0.038
	B-F	0.022	0.004	0.019	0.010	0.031
	HDBF	0.026	0.023	0.048	0.017	0.035
	TanBF	0.026	0.017	0.024	0.018	0.029
(0,0,0,0,0.7)	ANOVA	0.041	0.009	0.024	0.027	0.076
	B-F	0.034	0.005	0.017	0.021	0.065
	HDBF	0.039	0.031	0.043	0.027	0.070
	TanBF	0.036	0.019	0.022	0.027	0.050

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.5: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
	$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)					
(0,0,0,0)	ANOVA	0.009	0.004	0.011	0.007	0.011
	B-F	0.009	0.002	0.009	0.005	0.011
	HDBF	0.013	0.011	0.028	0.008	0.014
	TanBF	0.013	0.010	0.010	0.012	0.011
(0.5,0,0,0)	ANOVA	0.068	0.010	0.021	0.035	0.289
	B-F	0.066	0.007	0.011	0.031	0.279
	HDBF	0.071	0.056	0.038	0.042	0.282
	TanBF	0.067	0.044	0.018	0.051	0.174
(0,0,0,0,0.7)	ANOVA	0.301	0.035	0.036	0.142	0.897
	B-F	0.294	0.031	0.039	0.132	0.890
	HDBF	0.304	0.202	0.091	0.171	0.891
	TanBF	0.279	0.164	0.051	0.197	0.733

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.5: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)						
(0,0,0,0)	ANOVA	0.105	0.042	0.114	0.084	0.107
	B-F	0.029	0.007	0.055	0.021	0.033
	HDBF	0.033	0.032	0.100	0.029	0.037
	TanBF	0.032	0.027	0.068	0.026	0.037
(0.5,0,0,0)	ANOVA	0.114	0.048	0.102	0.104	0.162
	B-F	0.040	0.007	0.043	0.032	0.061
	HDBF	0.045	0.036	0.085	0.041	0.064
	TanBF	0.042	0.026	0.058	0.040	0.053
(0,0,0,0,0.7)	ANOVA	0.150	0.058	0.133	0.123	0.295
	B-F	0.045	0.016	0.066	0.033	0.083
	HDBF	0.051	0.050	0.115	0.044	0.088
	TanBF	0.048	0.040	0.096	0.041	0.071

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.5: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:16)						
(0,0,0,0)	ANOVA	0.016	0.003	0.035	0.010	0.019
	B-F	0.032	0.007	0.050	0.022	0.033
	HDBF	0.032	0.033	0.100	0.025	0.033
	TanBF	0.033	0.030	0.076	0.031	0.029
(0.5,0,0,0)	ANOVA	0.021	0.007	0.031	0.015	0.028
	B-F	0.036	0.013	0.047	0.027	0.054
	HDBF	0.039	0.040	0.101	0.038	0.055
	TanBF	0.038	0.037	0.078	0.038	0.042
(0,0,0,0,0.7)	ANOVA	0.049	0.013	0.018	0.033	0.130
	B-F	0.084	0.023	0.032	0.054	0.185
	HDBF	0.087	0.081	0.070	0.065	0.185
	TanBF	0.080	0.064	0.044	0.066	0.130

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.5: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)						
(0,0,0,0)	ANOVA	0.010	0.003	0.010	0.007	0.044
	B-F	0.018	0.005	0.016	0.012	0.020
	HDBF	0.021	0.025	0.053	0.018	0.023
	TanBF	0.018	0.017	0.025	0.021	0.019
(0.5,0,0,0)	ANOVA	0.009	0.003	0.016	0.007	0.086
	B-F	0.019	0.005	0.023	0.014	0.040
	HDBF	0.020	0.025	0.064	0.019	0.044
	TanBF	0.019	0.021	0.029	0.024	0.034
(0,0,0,0,0.7)	ANOVA	0.028	0.005	0.008	0.014	0.142
	B-F	0.047	0.010	0.017	0.029	0.058
	HDBF	0.052	0.045	0.045	0.034	0.061
	TanBF	0.048	0.038	0.018	0.042	0.042

$*$: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.5: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(11,16,16,21) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:9:4:1)						
(0,0,0,0)	ANOVA	0.043	0.017	0.047	0.038	0.010
	B-F	0.017	0.002	0.023	0.011	0.018
	HDBF	0.020	0.019	0.061	0.018	0.019
	TanBF	0.019	0.014	0.034	0.019	0.015
(0.5,0,0,0)	ANOVA	0.062	0.023	0.041	0.040	0.015
	B-F	0.024	0.005	0.014	0.016	0.028
	HDBF	0.027	0.026	0.047	0.022	0.030
	TanBF	0.027	0.019	0.021	0.020	0.022
(0,0,0,0,0.7)	ANOVA	0.070	0.022	0.054	0.047	0.078
	B-F	0.026	0.006	0.026	0.016	0.120
	HDBF	0.030	0.030	0.071	0.025	0.120
	TanBF	0.028	0.023	0.043	0.027	0.076

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.6: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.01$.

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
	$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:1:1:1)					
(0,0,0,0)	ANOVA	0.010	0.006	0.010	0.010	0.011
	B-F	0.010	0.005	0.010	0.010	0.011
	HDBF	0.013	0.016	0.066	0.015	0.012
	TanBF	0.010	0.019	0.012	0.017	0.006
(0.5,0,0,0)	ANOVA	0.475	0.045	0.051	0.205	0.982
	B-F	0.474	0.044	0.050	0.204	0.982
	HDBF	0.491	0.422	0.177	0.304	0.982
	TanBF	0.456	0.337	0.065	0.320	0.930
(0,0,0,0,0.7)	ANOVA	0.866	0.102	0.114	0.499	1.000
	B-F	0.865	0.100	0.114	0.498	1.000
	HDBF	0.875	0.776	0.302	0.665	1.000
	TanBF	0.846	0.664	0.148	0.679	1.000

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.6: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(16:1:1:1)						
(0,0,0,0)	ANOVA	0.045	0.027	0.051	0.040	0.041
	B-F	0.041	0.022	0.048	0.036	0.037
	HDBF	0.044	0.049	0.160	0.043	0.039
	TanBF	0.041	0.046	0.097	0.045	0.027
(0.5,0,0,0)	ANOVA	0.110	0.040	0.038	0.079	0.283
	B-F	0.100	0.034	0.033	0.070	0.267
	HDBF	0.106	0.110	0.112	0.097	0.268
	TanBF	0.097	0.092	0.056	0.095	0.186
(0,0,0,0,0.7)	ANOVA	0.126	0.035	0.072	0.077	0.653
	B-F	0.113	0.028	0.064	0.067	0.605
	HDBF	0.123	0.121	0.210	0.091	0.611
	TanBF	0.108	0.094	0.139	0.093	0.332

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

Table H.6: Empirical levels and powers: Tests for equality of four population means for (n_1, n_2, n_3, n_4)=(50,50,50,50) and $\alpha = 0.01$. *(continued)*

$(\mu_1, \mu_2, \mu_3, \mu_4)$		$N(0,1)$	Mix*	$\chi^2(3)$	$t(4)$	$U(-1,1)$
$(\sigma_1^2 : \sigma_2^2 : \sigma_3^2 : \sigma_4^2)$=(1:4:9:16)						
(0,0,0,0)	ANOVA	0.021	0.013	0.019	0.020	0.023
	B-F	0.020	0.012	0.017	0.019	0.022
	HDBF	0.024	0.026	0.101	0.023	0.022
	TanBF	0.021	0.026	0.031	0.024	0.012
(0.5,0,0,0)	ANOVA	0.065	0.021	0.018	0.041	0.171
	B-F	0.064	0.020	0.016	0.039	0.166
	HDBF	0.067	0.062	0.086	0.055	0.168
	TanBF	0.061	0.053	0.022	0.057	0.099
(0,0,0,0,0.7)	ANOVA	0.060	0.019	0.032	0.034	0.331
	B-F	0.057	0.017	0.031	0.033	0.318
	HDBF	0.068	0.065	0.162	0.046	0.322
	TanBF	0.062	0.059	0.069	0.054	0.156

*: Mix $= 0.9N(0,1) + 0.1N(0,64)$.

APPENDIX I

C PROGRAM 1: EQUALITY OF TWO MEANS WITH EQUAL VARIANCES

```
/*****************************************************************/
/*                                                               */
/*   Testing for equality of two population means when           */
/*   population variances are unequal                            */
/*                                                               */
/*****************************************************************/


#include <stdio.h>
#include <iostream.h>
#include <math.h>
#include <float.h>
#include <imsls.h>
#include <time.h>

#define N 5000
#define PI M_PI
#define ERROR 0.00001
#define ND 5
#define SIZE1 40
#define SIZE2 40

#define SIZE   ((SIZE1) + (SIZE2))
// #define BAND   0.5
#define NEH 100
#define SAM 5

FILE *fout;

void printHead( char s[] )
{
   time_t lt;

   lt = time(NULL);
   fprintf(fout,ctime(&lt));
   fprintf(fout,"One-Sided Test\n");

   fprintf(fout,"\n");
   fprintf(fout,"*****************************************\n");
   fprintf(fout,"\t\t%15s \n",s);
   fprintf(fout,"*****************************************\n");

}



/*********************************************************/
```

```
/*      Normal Random Number generation                    */
/*********************************************************/


void randomNormal(double r1[],double r2[], double avg, int m1, int m2)
{

   int i;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   for(i = 0; i < m1; i++)
    r1[i] = r1[i] + avg;

   seed = imsls_random_seed_get();

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

   seed=imsls_random_seed_get();

}



/*********************************************************/
/*      t(df) Random Number generation                   */
/*********************************************************/



void random_t(double r1[],double r2[],double avg,int m1, int m2)
{

   int i, df=2;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_student_t(m1,df,IMSLS_RETURN_USER,r1,0);


   for(i = 0; i < m1; i++)
    r1[i] = r1[i] + avg;

   seed = imsls_random_seed_get();
```

```
   imsls_random_seed_set(seed);
   imsls_d_random_student_t(m2,df,IMSLS_RETURN_USER,r2,0);

   seed=imsls_random_seed_get();

}



/*********************************************/
/*           Normal Mixture with t           */
/*********************************************/

void random43(double r1[],double r2[], double avg, int m1, int m2)
{

   int i, j;
   static int seed = 123475;
   double *trandom, *prob;

   imsls_random_option(5);

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   trandom = imsls_d_random_student_t(m1,1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   prob = imsls_d_random_uniform(m1,0);

   for(i=0; i < m1;i++)
   {
    if( prob[i] > 0.9 )
         r1[i] = r1[i] + avg;
        else
         r1[i] = trandom[i] + avg;
   }


      /* Second Set */

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
```

```
   imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   trandom=imsls_d_random_student_t(m2,1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   prob=imsls_d_random_uniform(m2,0);

   for(j=0; j < m2;j++)
   {
    if( prob[j] > 0.9 )
         r2[j] = r2[j];
        else
         r2[j] = trandom[j];
   }


seed = imsls_random_seed_get();
}


/*************************************************/
/*      Normal Mixture with Uniform              */
/*************************************************/

void random44(double random1[],double random2[], double avg, int m1,
           int m2)
{
   int i, j, k;
   static int seed = 123475;
   double *urandom, *prob;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,random1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   urandom=imsls_d_random_uniform(m1,0);

   for( k = 0; k < m1; k++){
    if( urandom[k] < DBL_EPSILON )
         urandom[k] = DBL_EPSILON;
         else
          ;
   }
```

```
    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    prob=imsls_d_random_uniform(m1,0);

    for(j=0; j < m1;j++)
    {
     if( prob[j] > 0.9 )
          random1[j] = random1[j]/urandom[j] + avg;
         else
          random1[j] = random1[j] + avg;
    }

    /* Second set */

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_normal(m2,IMSLS_RETURN_USER,random2,0);

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    urandom=imsls_d_random_uniform(m2,0);

    for( k = 0; k < m2; k++){
     if( urandom[k] < DBL_EPSILON )
          urandom[k] = DBL_EPSILON;
         else
           ;
    }

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    prob=imsls_d_random_uniform(m2,0);

    for(i=0; i < m2;i++)
    {
     if( prob[i] > 0.9 )
          random2[i] = random2[i]/urandom[i];
          else
          random2[i] = random2[i];
    }

seed = imsls_random_seed_get();
}

main()
{
```

```
double alpha005 = 0.05, alpha001 = 0.01;
double random_1[SIZE1]={0}, random_2[SIZE2]={0};
double HD_level,   HD_rej005,   HD_rej001;
double NED_level[ND], NED_rej005[ND], NED_rej001[ND];
double LD_level,   LD_rej005,   LD_rej001;
double TIKU_level, TIKU_rej005, TIKU_rej001;
double STUD_level, STUD_rej005, STUD_rej001;

void estimate(double [], double [], int, int);
double hd(double [], double [], int, int);
double ned(double [], double [], double, int, int);
double ld(double [], double [], int, int);
double tiku(double [], double [], int, int);
double student(double [], double [], int, int);

void (*f[4])(double [], double [], double, int ,int) = {
      randomNormal,random_t,random44, random43};

double diff[3]={0.0,1.0,2.5};
double a[ND] = {0.15,0.3,0.5,1.0,2};
int n1[SAM] = {8,20,8,40,40};
int n2[SAM] = {8,20,20,10,40};

double times, second;
int minute, sampleNumber;
int d,i,j,k, sam, choice;


fout = fopen("eq2mean.txt","a");
for( sam = 0; sam < SAM; sam++){

 for( choice = 0; choice < 4; choice++){
   switch( choice )
   {
    case 0:
      printHead("  Normal(0,1) ");
       break;
     case 1:
       printHead("     t(2)     ");
       break;
     case 2:
       printHead("0.9N(0,1)+0.1N(0,1)/U(0,1)");
       break;
     case 3:
      printHead("0.9N(0,1)+0.1t(1)");
       break;
     default:
```

```
        fprintf(fout,"No distribution Selected\n");
         break;
      }


      for( d = 0; d < 3; d++)
      {
         fprintf(fout,"----n1 = %d, n2 = %d  d = %2.1f  ---\n",
                                          n1[sam],n2[sam],diff[d]);

     sampleNumber = 0;
     HD_rej005 = 0;   HD_rej001=0;
     LD_rej005 = 0;   LD_rej001=0;
     TIKU_rej005 = 0;       TIKU_rej001=0;
     STUD_rej005 = 0;       STUD_rej001=0;

         for( i = 0; i < ND; i++)
         {
            NED_rej005[i] = 0;
            NED_rej001[i]=0;
         }

       while( sampleNumber >=0 && sampleNumber < N)
{

          (*f[choice])( random_1, random_2, diff[d], n1[sam], n2[sam]);

         estimate(random_1, random_2, n1[sam], n2[sam]);
         HD_level = hd(random_1, random_2, n1[sam], n2[sam]);

          for( i = 0; i < ND ; i++)
           NED_level[i] = ned(random_1, random_2, a[i], n1[sam], n2[sam]);

         if( (HD_level < 0)||(NED_level[0] < 0)||(NED_level[1] < 0) ||
              (NED_level[2] < 0)||(NED_level[3] < 0) || (NED_level[4]<0))
        ;
         else
          {
            if(HD_level < alpha001)
     HD_rej001 += 1;
            if(HD_level < alpha005)
     HD_rej005  += 1;

             for( k = 0; k < ND; k++)
             {
     if(NED_level[k] < alpha001)
       NED_rej001[k] += 1;
```

```
        if(NED_level[k] < alpha005)
          NED_rej005[k]  += 1;
               }

              LD_level = ld(random_1, random_2, n1[sam], n2[sam]);

                 if(LD_level < alpha001)
          LD_rej001 += 1;
        if(LD_level < alpha005)
          LD_rej005  += 1;

              TIKU_level = tiku(random_1, random_2, n1[sam] ,n2[sam]);

        if(TIKU_level < alpha001)
          TIKU_rej001 += 1;
        if(TIKU_level < alpha005)
          TIKU_rej005  += 1;

              STUD_level = student(random_1, random_2, n1[sam], n2[sam]);

if(STUD_level < alpha001)
    STUD_rej001 += 1;
        if(STUD_level < alpha005)
          STUD_rej005  += 1;

               sampleNumber++;
//             printf("%d %d\n", sampleNumber,choice);

            }  /*  else */
         } /* while */

      fprintf(fout,"Test Statistc   \talpha = 0.05 \talpha = 0.01\n");
      fprintf(fout,"_______________________________"
                                   "_____________________\n\n");
      fprintf(fout,"  HD            \t% f  \t% f\n", HD_rej005/N,
             HD_rej001/N);

      for( j = 0; j < ND; j++)
        fprintf(fout,"  NED(%3.2f)     \t% f  \t% f\n",a[j],NED_rej005[j]/N,
                                                    NED_rej001[j]/N);
      fprintf(fout,"  LD(MLE)       \t% f  \t% f\n",
                                          LD_rej005/N, LD_rej001/N);
      fprintf(fout,"  TIKU          \t% f  \t% f\n",
                                        TIKU_rej005/N, TIKU_rej001/N);
      fprintf(fout,"  t             \t% f  \t% f\n",
                                        STUD_rej005/N, STUD_rej001/N);
```

```
   }      /* d   */
 }  /* choice */
} /* SAM */
  times = imsls_ctime();
  minute = times/60;
  second = times - 60*minute;
  fprintf(fout,"\n\nCPU TIME USED is %d min %4.2f seconds\n", minute,
          second);
  return 0;
}



/*************************************************************************/
/* Calculation of Initial values and sample density estimate             */
/*************************************************************************/


double fstar1[SIZE1], fstar2[SIZE2];
double mu1hat, mu2hat, sigmahat, h1, h2;

void estimate(double r1[], double r2[], int m1, int m2)
{
   double *Median1, *Median2, *Sigma;
   double  sum, expo1[SIZE1]={0}, r12[SIZE], expo2[SIZE2]={0};
   double BAND1=0.5, BAND2=0.5;
   int i,j,k,l, m;

/* Calling of initail Values for MU_1, Mu_2, and Sigma */

   if( m1 < 30) BAND1=0.9;
   if( m2 < 30) BAND2=0.9;

   Median1 = imsls_d_simple_statistics(m1,1,r1,IMSLS_MEDIAN_AND_SCALE,0);
   Median2 = imsls_d_simple_statistics(m2,1,r2,IMSLS_MEDIAN_AND_SCALE,0);
   mu1hat =  Median1[14];
   mu2hat =  Median2[14];
   h1 = BAND1 * Median1[15];
   h2 = BAND2 * Median2[15];

   for(i=0; i < m1; i++)
      r12[i] = fabs(r1[i]-mu1hat);

   for(j=0; j < m2;j++)
      r12[m1+j] = fabs(r2[j]-mu2hat);

   m = m1+m2;
```

```
   Sigma =  imsls_d_simple_statistics(m,1,r12,IMSLS_MEDIAN_AND_SCALE,0);
   sigmahat =  1.48*Sigma[14];


/* Calling of Sample Density Estiamte */


   for(l = 0; l < m1; l++)
   {
    sum = 0.0;
    for(j = 0; j < m1; j++)
      {
       expo1[j] = ( (r1[j]-r1[l])*(r1[j]-r1[l]) )/(2.0*h1*h1);
              sum = sum + exp(-expo1[j]);
      }
     fstar1[l] = (double) sum/(m1*(sqrt(2.0*PI)*h1));

   }


   for( k = 0; k < m2; k++)
   {
    sum = 0.0;
    for(l = 0; l < m2; l++)
      {
       expo2[l] = ( (r2[l]-r2[k])*(r2[l]-r2[k]) )/(2.0*h2*h2);
              sum = sum + exp(-expo2[l]);
      }
     fstar2[k] = (double) sum/(m2*(sqrt(2.0*PI)*h2));
   }
}



/*****************************************************************/
/*                    Calculation of STATISTICS                  */
/*****************************************************************/


double ned(double r1[], double r2[], double b, int m1, int m2)
{
   double mstar1[SIZE1], mstar2[SIZE2];
   double mu1NED, mu2NED, sigmaNED;
   double raf1[SIZE1], raf2[SIZE2], weight1[SIZE1], weight2[SIZE2];
   double stopPoint;
   double bunmo,expon, delta1[SIZE1], delta2[SIZE2];
```

```
   double numerator, numer1, numer2, denominator1, denominator2;
   double NED;
   double ned_dist, mu1Hat, mu2Hat, sigmaHat;

   int i,j,k,l;
   int converge = 0;

   mu1Hat = mu1hat;
   mu2Hat = mu2hat;
   sigmaHat = sigmahat;


/*************************************************************/
/*                  Calculation of Estimators                */
/*************************************************************/



do
{

/*  First Model Smoothing here  */


   bunmo = 2.0*(sigmaHat*sigmaHat + h1*h1);

   for(i = 0; i < m1; i++)
   {
     expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/bunmo;
      mstar1[i] = exp( -expon )/sqrt(PI*bunmo);
      if(mstar1[i] <= DBL_EPSILON)
        weight1[i] = 0.0;
      else
      {
        delta1[i] = fstar1[i]/mstar1[i] - 1.0;
        raf1[i] = ( b+1.0 - (b+1.0+b*delta1[i])*exp( -(b*delta1[i]) ) )
                  /(b*b);
        weight1[i] = (raf1[i]+1)/(delta1[i]+1);
      }
        weight1[i] = weight1[i] < 0.0 ? 0.0:weight1[i];
   }



/*Second Model smoothing here  */
```

```
   bunmo = 2.0*(sigmaHat*sigmaHat + h2*h2);

   for(k = 0; k < m2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/bunmo;
      mstar2[k] = exp( -expon )/sqrt(PI*bunmo);
      if(mstar2[k] <= DBL_EPSILON)
    weight2[k] = 0.0;
      else
      {
       delta2[k] = fstar2[k]/mstar2[k] - 1.0;
      raf2[k] =( b+1.0 - (b+1.0+b*delta2[k])*exp(-(b*delta2[k])))
                /(b*b);
        weight2[k] = (raf2[k]+1)/(delta2[k]+1);
       }
       weight2[k] = weight2[k] < 0.0 ? 0.0:weight2[k];
   }



/*  Estiamtors of Mu1 Mu2 and Sigma  */

/*******************HERE IS Mu1 *******************/


    numerator = 0.0;
    denominator1 = 0.0;


    for(l = 0; l < m1; l++)
    {
        numerator = numerator + weight1[l]*r1[l];
          denominator1 = denominator1 + weight1[l];
    }

    mu1NED = numerator/denominator1;



/******************* HERE IS Mu2 ********************/


    numerator = 0.0;
    denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
```

```
        numerator = numerator + weight2[j]*r2[j];
          denominator2 = denominator2 + weight2[j];
    }

    mu2NED = numerator/denominator2;


/************** HERE IS POOLED SIGMA ********************/


numer1 = 0.0;
    numer2 = 0.0;

for(l = 0; l < m1; l++)
       numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
        for( k = 0; k< m2; k++)
          numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

    sigmaNED = sqrt((numer1 + numer2)/(denominator1 + denominator2));


   stopPoint = sqrt((mu1NED-mu1Hat)*(mu1NED-mu1Hat) +
                        (mu2NED-mu2Hat)*(mu2NED-mu2Hat)+
                               (sigmaNED-sigmaHat)*(sigmaNED-sigmaHat));

    mu1Hat = mu1NED;
  mu2Hat = mu2NED;
    sigmaHat = sigmaNED;

   converge++;
   if( converge == NEH )
   {
      printf("NED(%3.2f) FAIL TO CONVERGE\n", b);
      return(-1);
   }

}while(stopPoint > ERROR);

/* ============ REWEIGHTING ========================== */

        for(i=0;i<m1;i++)
        weight1[i] = sqrt(weight1[i]);

        for(j=0; j<m2;j++)
        weight2[j] = sqrt(weight2[j]);

    numerator = 0.0;
```

```
    denominator1 = 0.0;

    for(l = 0; l < m1; l++)
    {
       numerator = numerator + weight1[l]*r1[l];
         denominator1 = denominator1 + weight1[l];
    }

    mu1NED = numerator/denominator1;



    numerator = 0.0;
    denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
       numerator = numerator + weight2[j]*r2[j];
         denominator2 = denominator2 + weight2[j];
    }

    mu2NED = numerator/denominator2;



numer1 = 0.0;
    numer2 = 0.0;

for(l = 0; l < m1; l++)
     numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
        for( k = 0; k< m2; k++)
          numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

    sigmaNED = sqrt((numer1+numer2)/(denominator1+denominator2-2));


/* ============  END of REWEIGHTING ============================ */

NED = (mu1NED - mu2NED) / ( sigmaNED*sqrt(1.0/m1 + 1.0/m2) );

//    ned_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(NED),m1+m2-2) );
ned_dist = 1.0 - imsls_d_t_cdf(NED, m1+m2-2);

    return(ned_dist);
}
```

```
/*************************************************************/
/*       Calculation of Hellinger Disparity                  */
/*************************************************************/



double hd(double r1[], double r2[], int m1, int m2)
{
   double mstar1[SIZE1], mstar2[SIZE2];
   double mu1HD, mu2HD, sigmaHD;
//   double raf1[SIZE1], raf2[SIZE2],delta1[SIZE1], delta2[SIZE2];
   double stopPoint;
   double bunmo,expon,weight1[SIZE1], weight2[SIZE2];
   double numerator, numer1, numer2, denominator1, denominator2;
   double HD; /* TEST STATISTICS */
   double hd_dist, mu1Hat, mu2Hat, sigmaHat;
   int i, j, k, l;
   int converge = 0;

   mu1Hat = mu1hat;
   mu2Hat = mu2hat;
   sigmaHat = sigmahat;


do
{

/*  First Model Smoothing here  */


   bunmo = 2.0*(sigmaHat*sigmaHat + h1*h1);

   for(i = 0; i < m1; i++)
   {
      expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/bunmo;
      mstar1[i] = exp( -expon )/sqrt(PI*bunmo);
      if(fstar1[i] <= DBL_EPSILON)
            weight1[i] = 0.0;

      else
      {
//       delta1[i] = fstar1[i]/mstar1[i] - 1.0 ;
//       raf1[i] = 2.0*( sqrt(delta1[i]+1)-1 );
    weight1[i] = 2.0 * sqrt(mstar1[i]/fstar1[i])
       - mstar1[i]/fstar1[i];
         weight1[i] = weight1[i] < 0.0 ? 0.0:weight1[i];
```

```
      }
   }



/*Second Model smoothing here  */


   bunmo = 2.0*(sigmaHat*sigmaHat + h2*h2);

   for(k = 0; k < m2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/bunmo;
      mstar2[k] = exp( -expon )/sqrt(PI*bunmo);
      if(fstar2[k] <= DBL_EPSILON)
         weight2[k] = 0.0;

      else
      {
//      delta2[k] = fstar2[k]/mstar2[k] - 1.0;
//      raf2[k] = 2*( sqrt(delta2[k]+1)-1 );
        weight2[k] = 2.0 * sqrt(mstar2[k]/fstar2[k])
       - mstar2[k]/fstar2[k];
        weight2[k] = weight2[k] < 0.0 ? 0.0:weight2[k];
     }

   }


   /*********************************************************/
   /*              Estiamtors of Mu1 Mu2 Sigma              */
   /*********************************************************/


/*******************HERE IS Mu1*******************/

        numerator = 0.0;
        denominator1 = 0.0;

    for(l = 0; l < m1; l++)
  {
         numerator = numerator + weight1[l]*r1[l];
          denominator1 = denominator1 + weight1[l];
   }

    mu1HD = numerator/denominator1;
```

```
/******************** HERE IS Mu2  **********************/


   numerator = 0.0;
   denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
           numerator = numerator + weight2[j]*r2[j];
       denominator2 = denominator2 + weight2[j];
    }

    mu2HD = numerator/denominator2;


/************ HERE IS POOLED SIGMA **********************/


           numer1 = 0.0;
           numer2 = 0.0;

for(l = 0; l < m1; l++)
           numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
        for(k = 0; k < m2; k++)
         numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

   sigmaHD = sqrt((numer1 + numer2)/(denominator1 + denominator2));



   stopPoint = sqrt((mu1HD-mu1Hat)*(mu1HD-mu1Hat)+
                            (mu2HD-mu2Hat)*(mu2HD-mu2Hat)+
                              (sigmaHD-sigmaHat)*(sigmaHD-sigmaHat));

     mu1Hat = mu1HD;
    mu2Hat = mu2HD;
    sigmaHat = sigmaHD;

   converge++;
   if( converge == NEH )
   {
      printf("HD FAIL TO CONVERGE\n");
      return(-1);
   }
 }while(stopPoint > ERROR);
```

```
 /* ====================== REWEIGHTING ====================== */

  for( i = 0; i < m1; i++)
weight1[i] = sqrt(weight1[i]);

  for( j = 0; j < m2; j++)
weight2[j] = sqrt(weight2[j]);


        numerator = 0.0;
        denominator1 = 0.0;

    for(l = 0; l < m1; l++)
     {
             numerator = numerator + weight1[l]*r1[l];
             denominator1 = denominator1 + weight1[l];
      }

    mu1HD = numerator/denominator1;


      numerator = 0.0;
      denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
           numerator = numerator + weight2[j]*r2[j];
       denominator2 = denominator2 + weight2[j];
   }

    mu2HD = numerator/denominator2;


           numer1 = 0.0;
           numer2 = 0.0;

          for(l = 0; l < m1; l++)
             numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
          for(k = 0; k < m2; k++)
            numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

      sigmaHD = sqrt((numer1 + numer2)/(denominator1 + denominator2-2));


 /* ================= END of REWEIGHTING =============== */
```

```
    HD = (mu1HD - mu2HD) / ( sigmaHD*sqrt( 1.0/m1 + 1.0/m2) );
//  hd_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(HD),m1+m2-2) );
    hd_dist = 1.0 - imsls_d_t_cdf(HD,m1+m2-2);

    return(hd_dist);

}


/*****************************************************************/
/*        Calculation of Likelihood Disparity(MLE)               */
/*****************************************************************/



double ld(double r1[], double r2[], int m1, int m2)
{
   double mu1LD, mu2LD, sigmaLD;
   double sum1 = 0.0, sum2 = 0.0, varsum =0.0;
   double LD;
   double ld_dist;

   int i, j, k, l;

   for( i = 0; i < m1; i++)
    sum1 += r1[i];

   mu1LD = sum1 / m1;

   for( j = 0; j < m2; j++)
    sum2 += r2[j];

   mu2LD = sum2 / m2;

   for( k = 0; k < m1; k++)
    varsum += (r1[k] - mu1LD)*(r1[k] - mu1LD);

   for( l = 0; l < m2; l++)
    varsum += (r2[l] - mu2LD)*(r2[l] - mu2LD);

   sigmaLD = varsum / (m1+m2);

   LD = (mu1LD - mu2LD) / (sqrt( sigmaLD*( 1.0/m1 + 1.0/m2 )) );
//   ld_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(LD), m1+m2-2));
     ld_dist = 1.0 - imsls_d_t_cdf(LD, m1+m2-2);
```

```
    return(ld_dist);

}


/*****************************************************************/
/* Calculation of Tiku's Modified MLE Estimate and Test          */
/*****************************************************************/

double tiku(double rand1[], double rand2[], int k1, int k2)
{

   void bubble(double *, const int);

   double A1, A2, B1, B2, C1, C2, D1, D2, K1, K2, m1, m2;
   double alpha1, beta1, alpha2, beta2;
   double mu1Tiku, mu2Tiku;
   double sigma1Tiku, sigma2Tiku, sigma_sqTiku;
   double temp, tmp;
   double tiku_dist, TIKU; /* test statistics */
   double ran1[SIZE1], ran2[SIZE2];
   int n1, n2, r11, r12, r21, r22, i, j, k,l;


   bubble(rand1, k1);
   bubble(rand2, k2);

 switch(k1)
   {
      case 8:
              alpha1 = 0.6968;
               beta1 = 0.8481;
               break;
      case 10:
              alpha1 = 0.6737;
              beta1 = 0.8610;
              break;
       case 20:
        alpha1 = 0.6864;
              beta1 = 0.8401;
              break;
       default:
        alpha1 = 0.6902;
              beta1 = 0.8309;
      break;
   }
```

```
 switch(k2)
{
   case 8:
           alpha2 = 0.6968;
           beta2 = 0.8481;
           break;
   case 10:
           alpha2 = 0.6737;
           beta2 = 0.8610;
           break;
   case 20:
   alpha2 = 0.6864;
           beta2 = 0.8401;
           break;
  default:
     alpha2 = 0.6902;
           beta2 = 0.8309;
   break;
}

for( k = 0; k < k1; k++)
    ran1[k+1] = rand1[k];
for( j = 0; j < k2; j++)
    ran2[j+1] = rand2[j];

 n1 = k1;
 r11 = (int) floor(0.5+0.1*n1);
 r12 = (int) floor(0.5+0.1*n1);
 A1 = n1 - r11 -r12;
 m1 = n1 - r11 - r12 + r11*beta1 + r12*beta2;
 D1 = ( r12*alpha1 - r11*alpha2 )/ m1;

 tmp = 0.0;
 for( i = r11+1; i <= n1-r12; i++)
  tmp += ran1[i];

 K1 = ( tmp + r11*beta1*ran1[r11+1] + r12*beta2*ran1[n1-r12] )/m1;
 B1 = r12*alpha2*(ran1[n1-r12] - K1) - r11*alpha1*(ran1[r11+1] - K1);

 temp = 0.0;
 for( l = r11+1; l <= n1-r12; l++)
  temp += ran1[l]*ran1[l];

 C1 = temp + r11*beta1*ran1[r11+1]*ran1[r11+1]
           + r12*beta2*ran1[n1-r12]*ran1[n1-r12] - m1*K1*K1;

 sigma1Tiku = ( B1 + sqrt(B1*B1 + 4.0*A1*C1) )/( 2.0*sqrt( A1*(A1-1) ) );
```

```
   mu1Tiku = K1 + D1*sigma1Tiku;


   n2 = k2;
   r21 = (int) floor(0.5+0.1*n2);
   r22 = (int) floor(0.5+0.1*n2);
   A2 = n2 - r21 -r22;
   m2 = n2 - r21 - r22 + r21*beta1 + r22*beta2;
   D2 = ( r22*alpha1 - r21*alpha2 )/ m2;

   tmp = 0.0;
   for( k = r21+1; k <= n2-r22; k++)
    tmp += ran2[k];

   K2 = ( tmp + r21*beta1*ran2[r21+1] + r22*beta2*ran2[n2-r22] )/m2;
   B2 = r22*alpha2*(ran2[n2-r22] - K2) - r21*alpha1*(ran2[r21+1] - K2);

   temp = 0.0;
   for( j = r21+1; j <= n2-r22; j++)
    temp += ran2[j]*ran2[j];

   C2 = temp + r21*beta1*ran2[r21+1]*ran2[r21+1]
             + r22*beta2*ran2[n2-r22]*ran2[n2-r22] - m2*K2*K2;

   sigma2Tiku = ( B2 + sqrt(B2*B2 + 4.0*A2*C2) )/( 2.0*sqrt( A2*(A2-1) ) );
   mu2Tiku = K2 + D2*sigma2Tiku;


   sigma_sqTiku = ( (A1 - 1)*sigma1Tiku*sigma1Tiku
                  + (A2 - 1)*sigma2Tiku*sigma2Tiku ) /(A1 + A2 -2);

   TIKU = ( mu1Tiku - mu2Tiku ) / sqrt( sigma_sqTiku * ( 1.0/m1 + 1.0/m2) );
//   tiku_dist = 2.0*(1.0- imsls_d_t_cdf(fabs(TIKU),A1+A2-2));
   tiku_dist = 1.0 - imsls_d_t_cdf(TIKU, A1+A2-2);

   return(tiku_dist);

}


void bubble(double *array, const int size)
{
   int pass, j;

   void swap(double *, double *);

   for(pass = 1 ; pass < size; pass++)
```

```
    for(j = 0; j < size -1; j++)
       if(array[j] > array[j+1])
          swap(&array[j], &array[j+1]);
}


void swap(double *Ptr1, double *Ptr2)
{
   double temp;

   temp = *Ptr1;
   *Ptr1 = *Ptr2;
   *Ptr2 = temp;
}



/*******************************************************/
/*        Calculation of Student t statistics          */
/*******************************************************/



double student(double r1[], double r2[], int m1, int m2)
{
   double *student1, *student2;
   double mean1, mean2, sv1, sv2, Sp_sq, t_dist, student_t;


   student1 = imsls_d_simple_statistics(m1,1,r1,0);
   student2 = imsls_d_simple_statistics(m2,1,r2,0);

   mean1 = student1[0];
   mean2 = student2[0];
   sv1 = student1[1];
   sv2 = student2[1];

   Sp_sq = (double)  ( (m1-1)*sv1+(m2-1)*sv2 ) / ( m1+m2 - 2 );

   student_t = ( mean1 - mean2 ) /  sqrt( Sp_sq*(1.0/m1 + 1.0/m2) );
//   t_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(student_t), m1+m2-2));
   t_dist = 1.0 - imsls_d_t_cdf(student_t, m1+m2-2);

   return(t_dist);
}
```

```
/*************************************************************/
/*          END OF CODE  -- GOOD LUCK!!!                     */
/*************************************************************/
```

APPENDIX J

C PROGRAM 2: EQUALITY OF TWO MEANS WITH UNEQUAL VARIANCES

```
/*********************************************************************/
/*                                                                   */
/* THIS IS THE PROGRAM FOR TESTING EQUALITY OF TWO MEANS WHEN        */
/*  VARIANCES ARE UNEQUAL                                            */
/*                                                                   */
/*********************************************************************/

#include <stdio.h>
#include <math.h>
#include <float.h>
#include <imsls.h>
#include <stdlib.h>

#define N     5000
#define PI M_PI
#define ERROR  0.00001
#define SIZE1   70
#define SIZE2  70
#define NEH 100
#define SAM 7
#define STD 3



void bubble(double *, const int);

FILE *fout;


double MEDIAN(double r[], const int k)
{
  int check, q;
  double med;

  bubble(r,k);

  check = k%2;

  if( check == 0){
   q = k/2;
   med = (r[q]+r[q+1])/2.0;
  }
  else{
   q = (k+1)/2;
   med =  r[q];
  }
```

```
  return(med);
}

void printHead( char s[] )
{

//   fprintf(fout,"One-Sided Test\n");
   fprintf(fout,"Two-Sided Test\n");

fprintf(fout,"\n");
   fprintf(fout,"*****************************************************\n");
   fprintf(fout,"\t\t%21s \n",s);
   fprintf(fout,"*****************************************************\n");

}

/*************************************************************************/
/*                              t(4)                                     */
/*************************************************************************/

void randomStudent(double random1[], double random2[], double avg,
         double SD, int n1, int n2)
{
   int i;
   static int seed = 123475;
   imsls_random_seed_set(seed);
   imsls_d_random_student_t(n1,4,IMSLS_RETURN_USER,random1,0);

   for(i = 0; i < n1; i++)
    random1[i] = random1[i]/sqrt(2.0) + avg;

   seed = imsls_random_seed_get();

   imsls_random_seed_set(seed);
   imsls_d_random_student_t(n2,5,IMSLS_RETURN_USER,random2,0);

   for(i = 0; i < n2; i++)
    random2[i] = random2[i]*SD/sqrt(2.0);

   seed=imsls_random_seed_get();

}

/*************************************************************************/
/*                            N(0,1)                                     */
/*************************************************************************/
```

```
void randomNormal(double random1[],double random2[], double avg,
         double SD, int n1, int n2)
{

   int i;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(n1,IMSLS_RETURN_USER,random1,0);

   for(i = 0; i < n1; i++)
    random1[i] = random1[i] + avg;

   seed = imsls_random_seed_get();

   imsls_random_seed_set(seed);
   imsls_d_random_normal(n2,IMSLS_RETURN_USER,random2,0);

   for(i = 0; i < n2; i++)
    random2[i] = random2[i]*SD;

   seed=imsls_random_seed_get();

}

/*************************************************************************/
/*                        0.9N(0,1) + 0.1N(0,64)                         */
/*************************************************************************/

void randomMixed(double random1[],double random2[], double avg, double SD,
                 int n1, int n2)
{
   double r1[SIZE1], u1[SIZE1];
   double r2[SIZE2], u2[SIZE2];
   int i,j;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(n1,IMSLS_RETURN_USER,r1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   imsls_d_random_uniform(n1,IMSLS_RETURN_USER,u1,0);
```

```
   for(i = 0; i < n1; i++){
    if( u1[i] > 0.9 )
    random1[i] = 8.0*r1[i]/sqrt(0.72) + avg;
      else
       random1[i] = r1[i]/sqrt(0.72) + avg;
   }

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   imsls_d_random_normal(n2,IMSLS_RETURN_USER,r2,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   imsls_d_random_uniform(n2,IMSLS_RETURN_USER,u2,0);


   for(j = 0; j < n2; j++){
    if( u2[j] > 0.9 )
      random2[j] = 8.0*r2[j]*SD/sqrt(0.72);
        else
         random2[j] = r2[j]*SD/sqrt(0.72);
   }



   seed=imsls_random_seed_get();

}


/*************************************************************************/
/*                      0.9N(0,1) + 0.1[N(0,1)/U(0,1)]                   */
/*************************************************************************/

void randomCon(double random1[],double random2[], double avg, double SD,
                  int n1, int n2)
{
   double r1[SIZE1], u1[SIZE1], t1[SIZE1];
   double r2[SIZE2], u2[SIZE2], t2[SIZE2];
   int i,j;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(n1,IMSLS_RETURN_USER,r1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
```

```
    imsls_d_random_uniform(n1,IMSLS_RETURN_USER,u1,0);

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_uniform(n1,IMSLS_RETURN_USER,t1,0);



    for(i = 0; i < n1; i++){
     if( u1[i] > 0.9 )
       random1[i] = (r1[i]/t1[i])/sqrt(2.1) + avg;
         else
          random1[i] = r1[i]/sqrt(2.1) + avg;
    }

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_normal(n2,IMSLS_RETURN_USER,r2,0);

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_uniform(n2,IMSLS_RETURN_USER,u2,0);

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_uniform(n2,IMSLS_RETURN_USER,t2,0);


    for(j = 0; j < n2; j++){
     if( u2[j] > 0.9 )
       random2[j] = (r2[j]/t2[j])*SD/sqrt(2.1);
         else
          random2[j] = r2[j]*SD/sqrt(2.1);
    }



    seed=imsls_random_seed_get();

}


/*************************************************************************/
/*                      Uniform(-1,1)                                    */
/*************************************************************************/

void randomUnif(double random1[],double random2[], double avg, double SD,
```

```
                int n1, int n2)
{
   double  u1[SIZE1], u2[SIZE2];
   int i,j;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_uniform(n1,IMSLS_RETURN_USER,u1,0);

   for(i = 0; i < n1; i++)
    random1[i] = (2.0*u1[i]-1)*sqrt(3.0) + avg;

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   imsls_d_random_uniform(n2,IMSLS_RETURN_USER,u2,0);


   for(j = 0; j < n2; j++)
    random2[j] = (2.0*u2[j]-1)*SD*sqrt(3.0);

   seed=imsls_random_seed_get();

}

/*************************************************************************/
/*                        main program                                   */
/*************************************************************************/

main()
{

   double random_1[SIZE1]={0}, random_2[SIZE2]={0};
   double HD_level,   HD_rej10, HD_rej05,   HD_rej01;
   double NED_level, NED_rej10, NED_rej05, NED_rej01;
   double W_level,   W_rej10, W_rej05,   W_rej01;
   double TIKU_level, TIKU_rej10, TIKU_rej05, TIKU_rej01;

   void estimate(double [], double [], int, int);
   double hd(double [], double [], int, int);
   double ned(double [], double [], int, int);
   double Welch(double [], double [], int, int);
   double tiku(double [], double [], int, int);

  void (*f[5])(double [], double [],double, double, int, int) = {
         randomMixed,randomCon,randomNormal,randomStudent, randomUnif};

   double diff[3];
```

```
   double sd[STD] = {1.0,2,4.0};
   int m1[SAM] = {10,10,20,30,40,10,70};
   int m2[SAM] = {10,20,10,30,10,40,70};

   int sampleNumber, n1, n2;
   int d,o,i,choice;


   fout = fopen("un2mean.txt","a");
for( choice = 3; choice < 5; choice++)   ////
{

switch( choice )
   {
      case 0:
        printHead("0.9N(0,1)+0.1N(0,64)");
        diff[0]=0.0; diff[1]=1.0; diff[2]=2.5;
       break;
      case 1:
       printHead("0.9N(0,1)+0.1[N(0,1)/U(0,1)]");
        diff[0]=0.0; diff[1]=1.0; diff[2]=2.5;
        break;
      case 2:
     printHead("      Normal        ");
        diff[0]=0.0; diff[1]=1.0; diff[2]=2.5;
        break;
      case 3:
        printHead("        t(5)          ");
        diff[0]=0.0; diff[1]=1.0; diff[2]=2.5;
        break;
      case 4:
       printHead("    Uniform(-1,1)   ");
        diff[0]=0.0; diff[1]=0.5; diff[2]=1.0;
        break;
      default:
       fprintf(fout,"No distribution Selected\n");
       break;
   }


   for(i = 0; i < SAM; i++){   /* sample size */
      n1 = m1[i];
      n2 = m2[i];
      for( o = 0; o < STD; o++){  /* sd */

    for( d = 0; d < 3; d++){  /* mean difference */
```

```
        fprintf(fout,"\n** n1 = %d, n2 = %d  d = %2.1f sd1:sd2 = 1:%3.2f"
                "**\n", n1, n2, diff[d], sd[o]);

    sampleNumber = 0;

    HD_rej05 = 0;      HD_rej01=0;   HD_rej10=0;
    W_rej05 = 0;      W_rej01=0;    W_rej10=0;
    TIKU_rej05 = 0;   TIKU_rej01=0; TIKU_rej10=0;
               NED_rej05 = 0;    NED_rej01=0;  NED_rej10=0;


    while( sampleNumber >=0 && sampleNumber < N)
{
        (*f[choice])( random_1, random_2, diff[d], sd[o], n1, n2);

       estimate(random_1, random_2, n1, n2);
       HD_level = hd(random_1, random_2, n1, n2);
        NED_level = ned(random_1, random_2, n1, n2);


       if( (HD_level < 0)||(NED_level < 0))
            ;
       else
       {

       if(HD_level < 0.01)
       HD_rej01 += 1;
       if(HD_level < 0.05)
       HD_rej05  += 1;
if(HD_level < 0.10)
       HD_rej10  += 1;


           if(NED_level < 0.01)
         NED_rej01 += 1;
         if(NED_level < 0.05)
         NED_rej05 += 1;
         if(NED_level < 0.10)
         NED_rej10 += 1;


       W_level = Welch(random_1, random_2, n1, n2);

       if(W_level < 0.01)
       W_rej01 += 1;
       if(W_level < 0.05)
       W_rej05  += 1;
```

```
        if(W_level < 0.10)
        W_rej10  += 1;

        TIKU_level = tiku(random_1, random_2, n1, n2);

        if(TIKU_level < 0.01)
        TIKU_rej01 += 1;
        if(TIKU_level < 0.05)
        TIKU_rej05  += 1;
        if(TIKU_level < 0.10)
        TIKU_rej10  += 1;



         sampleNumber++;
//         printf("%d\n",  sampleNumber);

        }
          /*  else */
     }     /* while */

   fprintf(fout,"Test Statistc   \t\t 10% \t\t 5%\t\t 1%\n");
   fprintf(fout,"_____________________________________________________"
          "__\n\n");

   fprintf(fout,"  HD            \t% f \t% f \t% f\n",
                          HD_rej10/N, HD_rej05/N, HD_rej01/N);
   fprintf(fout,"  NED           \t% f \t% f \t% f\n",
                          NED_rej10/N, NED_rej05/N, NED_rej01/N);
   fprintf(fout,"  Welch         \t% f \t% f \t% f\n",
                          W_rej10/N, W_rej05/N, W_rej01/N);
   fprintf(fout,"  TIKU          \t% f \t% f \t% f\n",
                          TIKU_rej10/N, TIKU_rej05/N, TIKU_rej01/N);
   } /* for diff */
  } /* for sd */
 } /* for sample sizes */

 } /* choice */
  return 0;
}

/*************************************************************************/
/* Calculation of Initial values and sample density estimates          */
/*************************************************************************/


double fstar_1[SIZE1], fstar_2[SIZE2];
```

```
double mu1hat, mu2hat, sigma1hat, sigma2hat, h1, h2;

void estimate(double r1[], double r2[], int n1, int n2)
{
   double  sum, tmp1[SIZE1]={0}, expo1[SIZE1]={0};
   double  tmp2[SIZE2]={0}, expo2[SIZE2]={0}, BAND1, BAND2;
   double *Median1, *Median2;
//   double  t1[SIZE1], t2[SIZE2], MAD;
   int j,k,l;

   if( n1 < 30 ){
    BAND1 = 0.9;
   }
   else{
    BAND1 = 0.5;
   }

   if( n2 < 30){
    BAND2 = 0.9;
   }
   else {
    BAND2 = 0.5;
   }


/* Calling of initail Values for MU_1, Mu_2, and Sigma */


   Median1 = imsls_d_simple_statistics(n1,1,r1,IMSLS_MEDIAN_AND_SCALE,0);
   Median2 = imsls_d_simple_statistics(n2,1,r2,IMSLS_MEDIAN_AND_SCALE,0);
   mu1hat =  Median1[14];
   mu2hat =  Median2[14];
   sigma1hat = Median1[16];
   sigma2hat = Median2[16];
   h1 = BAND1 * Median1[15];
   h2 = BAND2 * Median2[15];
/*
   mu1hat = MEDIAN(r1, n1);
   mu2hat = MEDIAN(r2, n2);

   for(i=0; i < n1; i++)
    t1[i] = fabs(r1[i]-mu1hat);

   MAD = MEDIAN(t1, n1);
   sigma1hat = 1.48258*MAD;
   h1 = BAND*MAD;
```

```
   for(i=0; i < n2; i++)
    t2[i] = fabs(r2[i]-mu2hat);

   MAD = MEDIAN(t2, n2);
   sigma2hat = 1.48258*MAD;
   h2 = BAND*MAD;

*/
/* Calling of Sample Density Estiamte */


   for(l = 0; l < n1; l++)
   {
    sum = 0.0;
    for(j = 0; j < n1; j++)
      {
       expo1[j] = ( (r1[j]-r1[l])*(r1[j]-r1[l]) )/(2.0*h1*h1);
        tmp1[j] = exp(-expo1[j])/(sqrt(2.0*PI)*h1);
        sum = sum + tmp1[j];
      }
     fstar_1[l] = (double) sum/n1;

   }


   for( k = 0; k < n2; k++)
   {
    sum = 0.0;
    for(l = 0; l < n2; l++)
      {
       expo2[l] = ( (r2[l]-r2[k])*(r2[l]-r2[k]) )/(2.0*h2*h2);
        tmp2[l] = exp(-expo2[l])/(sqrt(2.0*PI)*h2);
        sum = sum + tmp2[l];
      }
     fstar_2[k] = (double) sum/n2;

   }


}




/*********************************************************************/
/*                    Calculation of STATISTICS                      */
/*********************************************************************/
```

```
double ned(double r1[], double r2[], int n1, int n2)
{
   double mstar_1[SIZE1], mstar_2[SIZE2];
   double mu1NED, mu2NED, sigma1NED, sigma2NED, s1NED, s2NED;
   double raf1[SIZE1], raf2[SIZE2], weight1[SIZE1], weight2[SIZE2];
   double stopPoint, dof, c;
   double bunmo,expon, delta1[SIZE1], delta2[SIZE2];
   double numerator, numer1, numer2, den1, den2;
   double NED, ned_dist, mu1Hat, mu2Hat, s1Hat, s2Hat;

   int i,j,k,l;
   int converge = 0;

   mu1Hat = mu1hat;
   mu2Hat = mu2hat;
   s1Hat = sigma1hat;
   s2Hat = sigma2hat;


/*************************************************************************/
/*                        Calculation of Estimators                      */
/*************************************************************************/

do
{

/*  First Model Smoothing here  */


   bunmo = 2.0*(s1Hat*s1Hat + h1*h1);

   for(i = 0; i < n1; i++)
   {
      expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/bunmo;
      mstar_1[i] = exp( -expon )/sqrt(PI*bunmo);
      if(mstar_1[i] <= DBL_EPSILON)
    weight1[i] = 0.0;
      else
      {
        delta1[i] = fstar_1[i]/mstar_1[i] - 1.0;
         raf1[i] =2.0 - (2.0+delta1[i])*exp(-delta1[i]);
         weight1[i] = (raf1[i]+1)/(delta1[i]+1);
      }

      if(weight1[i] <= 0.0 )
```

```
        weight1[i]=0.0;
      else if( weight1[i] >= 1.0)
       weight1[i] = 1.0;
      else
       ;

  }


/*Second Model smoothing here  */


   bunmo = 2.0*(s2Hat*s2Hat + h2*h2);

   for(k = 0; k < n2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/(2.0*bunmo);
      mstar_2[k] = exp( -expon )/sqrt(2.0*PI*bunmo);

      if(mstar_2[k] <= DBL_EPSILON)
    weight2[k] = 0.0;
      else
      {
       delta2[k] = fstar_2[k]/mstar_2[k] - 1.0;
      raf2[k] = 2.0 - (2.0+delta2[k])*exp(-delta2[k]);
        weight2[k] = (raf2[k]+1)/(delta2[k]+1);
       }
      if(weight2[k] <= 0.0 )
       weight2[k]=0.0;
      else if( weight2[k] >= 1.0)
       weight2[k] = 1.0;
      else
       ;
   }


         /*  Estiamtors of Mu1 Mu2 Sigma  */

/*******************HERE IS Mu1 *******************/


    numerator = 0.0;
    den1 = 0.0;


    for(l = 0; l < n1; l++)
    {
```

```
        numerator = numerator + weight1[l]*r1[l];
          den1 = den1 + weight1[l];
    }

    mu1NED = numerator/den1;



/******************* HERE IS Mu2 ********************/


    numerator = 0.0;
    den2 = 0.0;

    for(j = 0; j < n2; j++)
    {
       numerator = numerator + weight2[j]*r2[j];
         den2 = den2 + weight2[j];
    }

    mu2NED = numerator/den2;


/************** HERE ARE SIGMAs ********************/


numer1 = 0.0;
    numer2 = 0.0;

for(l = 0; l < n1; l++)
      numer1 = numer1 + weight1[l]*(r1[l]-mu1NED)*(r1[l]-mu1NED);
        for( k = 0; k< n2; k++)
         numer2 = numer2 + weight2[k]*(r2[k]-mu2NED)*(r2[k]-mu2NED);

    s1NED = sqrt(numer1 / den1);
        s2NED = sqrt(numer2 / den2);



    stopPoint=(mu1NED-mu1Hat)*(mu1NED-mu1Hat)+(mu2NED-mu2Hat)*
                   (mu2NED-mu2Hat)+(s1NED-s1Hat)*(s1NED-s1Hat)+
                    (s2NED-s2Hat)*(s2NED-s2Hat);

    mu1Hat = mu1NED;
   mu2Hat = mu2NED;
    s1Hat = s1NED;
```

```
        s2Hat = s2NED;

   converge++;
   if( converge == NEH )
      return(-1);

}while(sqrt(stopPoint) > ERROR);


   /* Reweighting Estiamtors of Mu1 Mu2 Sigma1 and Sigma2  */


    numerator = 0.0;
    den1 = 0.0;


    for(l = 0; l < n1; l++)
    {
       numerator = numerator + sqrt(weight1[l])*r1[l];
         den1 = den1 + sqrt(weight1[l]);
    }

    mu1NED = numerator/den1;



    numerator = 0.0;
    den2 = 0.0;

    for(j = 0; j < n2; j++)
    {
       numerator = numerator + sqrt(weight2[j])*r2[j];
         den2 = den2 + sqrt(weight2[j]);
    }

    mu2NED = numerator/den2;



numer1 = 0.0;
    numer2 = 0.0;
```

```
for(l = 0; l < n1; l++)
      numer1 = numer1+sqrt(weight1[l])*(r1[l]-mu1NED)*(r1[l]-mu1NED);
        for( k = 0; k< n2; k++)
         numer2 = numer2+sqrt(weight2[k])*(r2[k]-mu2NED)*(r2[k]-mu2NED);

    s1NED = sqrt(numer1 /den1);
        s2NED = sqrt(numer2 / den2);


        sigma1NED = (den1*s1NED*s1NED)/(den1-1.0);
        sigma2NED = (den2*s2NED*s2NED)/(den2-1.0);

//   NED = (mu1NED - mu2NED) / sqrt(sigma1NED/den1 + sigma2NED/den2);
//        c = ( sigma1NED/den1 ) / ( sigma1NED/den1 + sigma2NED/den2 );

        NED = (mu1NED - mu2NED) / sqrt(sigma1NED/n1 + sigma2NED/n2);
        c = ( sigma1NED/n1 ) / ( sigma1NED/n1 + sigma2NED/n2 );
        dof = (double) ((n1-1)*(n2-1) )/( (n2-1)*c*c+(n1-1)*(1-c)*(1-c));
        if(dof<=1.0)
        dof=1.0;

    ned_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(NED),dof) );
//        ned_dist = 1.0 - imsls_d_t_cdf(NED, SIZE-2);

return(ned_dist);
}



/*************************************************************************/
/*                    Calculation of Hellinger Disparity                 */
/*************************************************************************/



double hd(double r1[], double r2[], int n1, int n2)
{
   double mstar_1[SIZE1], mstar_2[SIZE2];
   double mu1HD, mu2HD, s1HD, s2HD;
//   double raf1[SIZE1], raf2[SIZE2], delta1[SIZE1], delta2[SIZE2];
   double stopPoint, dof, c;
   double bunmo,expon, weight1[SIZE1], weight2[SIZE2];
   double numerator, numer1, numer2, den1, den2;
   double HD; /* TEST STATISTICS */
   double hd_dist, mu1Hat, mu2Hat, s1Hat, s2Hat;
   int i, j, k, l;
   int converge = 0;
```

```
   mu1Hat = mu1hat;
   mu2Hat = mu2hat;
   s1Hat = sigma1hat;
   s2Hat = sigma2hat;


do
{

/*  First Model Smoothing here  */


   bunmo = 2.0*(s1Hat*s1Hat + h1*h1);

   for(i = 0; i < n1; i++)
   {
      expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/(bunmo);
      mstar_1[i] = exp( -expon )/sqrt(PI*bunmo);

      if(fstar_1[i] <= DBL_EPSILON)
   weight1[i] = 0.0;
      else
      {
        /*
        delta1[i] = fstar_1[i]/mstar_1[i] - 1.0 ;
         raf1[i] = 2.0*( sqrt(delta1[i]+1)-1 );
         weight1[i] = (raf1[i]+1)/(delta1[i]+1);
        */
        weight1[i] = 2.0*sqrt(mstar_1[i]/fstar_1[i])-mstar_1[i]/fstar_1[i];
        if(weight1[i] <= 0.0)
          weight1[i]=0.0;
      }
   }



/*Second Model smoothing here  */


   bunmo = 2.0*(s2Hat*s2Hat + h2*h2);

   for(k = 0; k < n2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/bunmo;
      mstar_2[k] = exp( -expon )/sqrt(PI*bunmo);
```

```
      if(fstar_2[k] <= DBL_EPSILON)
    weight2[k] = 0.0;
      else
      {
        /*
        delta2[k] = fstar_2[k]/mstar_2[k] - 1.0;
       raf2[k] = 2*( sqrt(delta2[k]+1)-1 );
         weight2[k] = (raf2[k]+1)/(delta2[k]+1);
        */
        weight2[k] = 2.0*sqrt(mstar_2[k]/fstar_2[k])-mstar_2[k]/fstar_2[k];
        if(weight2[k] <= 0.0)
          weight2[k]=0.0;
      }
   }



   /*********************************************************/
   /*              Estiamtors of Mu1 Mu2 Sigma              */
   /*********************************************************/


/*******************HERE IS Mu1 *******************/

    numerator = 0.0;
        den1 = 0.0;

    for(l = 0; l < n1; l++)
        {
          numerator = numerator + weight1[l]*r1[l];
        den1 = den1 + weight1[l];
   }

    mu1HD = numerator/den1;


/******************** HERE IS Mu2 **********************/


numerator = 0.0;
den2 = 0.0;

    for(j = 0; j < n2; j++)
    {
       numerator = numerator + weight2[j]*r2[j];
      den2 = den2 + weight2[j];
```

```
    }

     mu2HD = numerator/den2;


/************ HERE ARE SIGMAs **********************/


 numer1 = 0.0;
    numer2 = 0.0;

 for(l = 0; l < n1; l++)
           numer1 = numer1 + weight1[l]*(r1[l]-mu1HD)*(r1[l]-mu1HD);
         for(k = 0; k < n2; k++)
        numer2 = numer2 + weight2[k]*(r2[k]-mu2HD)*(r2[k]-mu2HD);


     s1HD = sqrt(numer1 / den1);
         s2HD = sqrt(numer2 / den2);


     stopPoint = (mu1HD-mu1Hat)*(mu1HD-mu1Hat)+(mu2HD-mu2Hat)*
                       (mu2HD-mu2Hat)+(s1HD-s1Hat)*(s1HD-s1Hat)+
                         (s2HD-s2Hat)*(s2HD-s2Hat);

      mu1Hat = mu1HD;
      mu2Hat = mu2HD;
     s2Hat = s2HD;
         s1Hat = s1HD;

         converge++;

         if( converge == NEH )
             return(-1);

}while(sqrt(stopPoint) > ERROR);

    /* Reweighting */


    numerator = 0.0;
   den1 = 0.0;

    for(l = 0; l < n1; l++)
  {
       numerator = numerator + sqrt(weight1[l])*r1[l];
        den1 = den1 + sqrt(weight1[l]);
```

```
   }

    mu1HD = numerator/den1;


        numerator = 0.0;
den2 = 0.0;

    for(j = 0; j < n2; j++)
    {
       numerator = numerator + sqrt(weight2[j])*r2[j];
      den2 = den2 + sqrt(weight2[j]);
    }

    mu2HD = numerator/den2;

numer1 = 0.0;
   numer2 = 0.0;

for(l = 0; l < n1; l++)
          numer1 = numer1 + sqrt(weight1[l])*(r1[l]-mu1HD)*(r1[l]-mu1HD);
        for(k = 0; k < n2; k++)
      numer2 = numer2 + sqrt(weight2[k])*(r2[k]-mu2HD)*(r2[k]-mu2HD);

    s1HD = numer1 / (den1-1.0);    /* estimates of sigma^2 */
        s2HD = numer2 / (den2-1.0);

  HD = (mu1HD - mu2HD) /  ( sqrt(s1HD/n1 + s2HD/n2) );
//   HD = (mu1HD - mu2HD) / (sqrt(sigma1HD/den1 + sigma2HD/den2));
//        c = (sigma1HD/den1) / (sigma1HD/den1 + sigma2HD/den2);
        c = (s1HD/n1) / (s1HD/n1 + s2HD/n2);
        dof = (double) ((n1-1)*(n2-1))/((n2-1)*c*c+(n1-1)*(1-c)*(1-c));
        if(dof <= 1.0)
          dof=1.0;


   hd_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(HD),dof) );
//    hd_dist = 1.0 - imsls_d_t_cdf(HD,dof);

    return(hd_dist);
}


/*************************************************************************/
/*              Calculation of Likelihood Disparity(MLE)                 */
/*************************************************************************/
```

```
double Welch(double r1[], double r2[], int n1, int n2)
{
   double mu1W, mu2W, sigma1W, sigma2W;
   double sum1 = 0.0, sum2 = 0.0, varsum1 =0.0, varsum2=0.0;
   double W, w_dist, dof, c;
   int i, j, k, l;

   for( i = 0; i < n1; i++)
    sum1 += r1[i];

   mu1W = sum1 / n1;

   for( j = 0; j < n2; j++)
    sum2 += r2[j];

   mu2W = sum2 / n2;

   for( k = 0; k < n1; k++)
    varsum1 += (r1[k] - mu1W)*(r1[k] - mu1W);

   for( l = 0; l < n2; l++)
    varsum2 += (r2[l] - mu2W)*(r2[l] - mu2W);

   sigma1W = varsum1 / (n1-1);
   sigma2W = varsum2 / (n2-1);

   W = (mu1W - mu2W) / (sqrt( sigma1W/n1 + sigma2W/n2) );


  c = ( sigma1W/n1 ) / ( sigma1W/n1 + sigma2W/n2 );
  dof = (double) ( (n1-1)*(n2-1) )/( (n2-1)*c*c + (n1-1)*(1-c)*(1-c) );
  if(dof<=1.00){
   dof=1.0;
   }

   w_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(W), dof));
// w_dist = 1.0 - imsls_d_t_cdf(W, dof);

    return(w_dist);

}


/*************************************************************************/
/* Calculation of Tiku's Modified MLE Estimate and Test                  */
```

```
/*************************************************************************/


double tiku(double rand1[], double rand2[], int k1, int k2)
{

   double A1, A2, B1, B2, C1, C2, D1, D2, K1, K2, m1, m2;
   double alpha1, beta1, alpha2, beta2;
   double mu1Tiku, mu2Tiku;
   double sigma1Tiku, sigma2Tiku;
   double temp, tmp, c;
   double tiku_dist, TIKU; /* test statistics */
   double ran1[SIZE1], ran2[SIZE2], dof;
   int n1, n2, r11, r12, r21, r22, i, j, k,l;


   bubble(rand1, k1);
   bubble(rand2, k2);

       /* re-indecies to keep the subscripts in the formulat */

   switch(k1)
   {
      case 10:
         alpha1 = 0.6737;
         beta1 = 0.8610;
         break;
      case 20:
       alpha1 = 0.6864;
         beta1 = 0.8401;
         break;
      case 30:
         alpha1 = 0.6880;
         beta1 = 0.8365;
         break;
      default:
       alpha1 = 0.6902;
         beta1 = 0.8309;
       break;
   }

   switch(k2)
   {
      case 10:
         alpha2 = 0.6737;
         beta2 = 0.8610;
```

```
        break;
    case 20:
     alpha2 = 0.6864;
        beta2 = 0.8401;
        break;
    case 30:
        alpha2 = 0.6880;
        beta2 = 0.8365;
        break;
    default:
     alpha2 = 0.6902;
        beta2 = 0.8309;
     break;
 }

for( k = 0; k < k1; k++)
    ran1[k+1] = rand1[k];
for( j = 0; j < k2; j++)
    ran2[j+1] = rand2[j];

 n1 = k1;
 r11 = (int) floor(0.5+0.1*n1);
 r12 = (int) floor(0.5+0.1*n1);
 A1 = n1 - r11 -r12;
 m1 = n1 - r11 - r12 + r11*beta1 + r12*beta2;
 D1 = ( r12*alpha1 - r11*alpha2 )/ m1;

 tmp = 0.0;
 for( i = r11+1; i <= n1-r12; i++)
  tmp += ran1[i];

 K1 = ( tmp + r11*beta1*ran1[r11+1] + r12*beta2*ran1[n1-r12] )/m1;
 B1 = r12*alpha2*(ran1[n1-r12] - K1) - r11*alpha1*(ran1[r11+1] - K1);

 temp = 0.0;
 for( l = r11+1; l <= n1-r12; l++)
  temp += ran1[l]*ran1[l];

 C1 = temp + r11*beta1*ran1[r11+1]*ran1[r11+1]
           + r12*beta2*ran1[n1-r12]*ran1[n1-r12] - m1*K1*K1;

 sigma1Tiku = (B1 + sqrt(B1*B1 + 4.0*A1*C1))/(2.0*sqrt( A1*(A1-1)));
 mu1Tiku = K1 + D1*sigma1Tiku;


 n2 = k2;
 r21 = (int) floor(0.5+0.1*n2);
```

```
   r22 = (int) floor(0.5+0.1*n2);
   A2 = n2 - r21 -r22;
   m2 = n2 - r21 - r22 + r21*beta1 + r22*beta2;
   D2 = ( r22*alpha1 - r21*alpha2 )/ m2;

   tmp = 0.0;
   for( k = r21+1; k <= n2-r22; k++)
    tmp += ran2[k];

   K2 = ( tmp + r21*beta1*ran2[r21+1] + r22*beta2*ran2[n2-r22] )/m2;
   B2 = r22*alpha2*(ran2[n2-r22] - K2) - r21*alpha1*(ran2[r21+1] - K2);

   temp = 0.0;
   for( j = r21+1; j <= n2-r22; j++)
    temp += ran2[j]*ran2[j];

   C2 = temp + r21*beta1*ran2[r21+1]*ran2[r21+1]
             + r22*beta2*ran2[n2-r22]*ran2[n2-r22] - m2*K2*K2;

   sigma2Tiku = (B2 + sqrt(B2*B2+4.0*A2*C2))/(2.0*sqrt(A2*(A2-1)));
   mu2Tiku = K2 + D2*sigma2Tiku;

   sigma1Tiku = sigma1Tiku*sigma1Tiku;
   sigma2Tiku = sigma2Tiku*sigma2Tiku;

   c = (sigma1Tiku/m2)/(sigma1Tiku/m1 + sigma2Tiku/m2);
   dof = (double) ( (A1-1)*(A2-1) )/( (A2-1)*c*c + (A1-1)*(1-c)*(1-c) );
   if(dof <= 1.0 ){
   dof=1.0;
   }


   TIKU = ( mu1Tiku - mu2Tiku ) / sqrt( sigma1Tiku/m1 + sigma2Tiku/m2 );
   tiku_dist = 2.0*(1.0- imsls_d_t_cdf(fabs(TIKU),dof));
//   tiku_dist = 1.0 - imsls_d_t_cdf(TIKU, dof);


   return(tiku_dist);

}


void bubble(double *array, const int size)
{
   int pass, j;

   void swap(double *, double *);
```

```
   for(pass = 1 ; pass < size; pass++)
     for(j = 0; j < size -1; j++)
       if(array[j] > array[j+1])
           swap(&array[j], &array[j+1]);
}


void swap(double *Ptr1, double *Ptr2)
{
   double temp;

   temp = *Ptr1;
   *Ptr1 = *Ptr2;
   *Ptr2 = temp;
}

/*************************************************************************/
/*          END OF CODE  -- GOOD LUCK!!!                                 */
/*************************************************************************/
```

APPENDIX K

C PROGRAM 3: EQUALITY OF TWO VARIANCES

```
   ]

/*************************************************************/
/*                                                           */
/*        Program for Testing Equality of Variances          */
/*        When Group Size is  >= 3                           */
/*        Call F-ratio for two populations                   */
/*                                                           */
/*************************************************************/


#include <imsls.h>
#include <stdio.h>
#include <float.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>

#define PI M_PI
#define ERROR 0.000001
#define NEH 100

/* Change the following number as needed */

#define GROUP   2          /* GROUP SIZE */
#define ITER 5000
#define SIZE 40        /* Largest sample Size */
#define BAND 0.5

FILE *fout;

void bubble(double *array, const int size);
double MEDIAN(double [], const int );
double MEAN(double [], const int);
double VAR(double [], const int);

main()
{
   double hd(int [], double [][SIZE]);
   double bart(int [], double [][SIZE]);
//   double LEVEN(int [], double [][SIZE]);
   double Fratio(int [], double [][SIZE]);

   int SAM, STD, n[GROUP];
   double std[GROUP];

   double unif[GROUP][SIZE];
```

```
double x[GROUP][SIZE];
static int seed;
double leven, lev90, lev95, lev99;
int j, k, rep, POP, UE;
double hdbf, hdbf90, hdbf95, hdbf99;
double Bart, b90, b95, b99;
char *dist[26]={"0.9N(0,1)+0.1N(0,8)","N(0,1)","t(4)","Chisq(4)"};


fout = fopen("fratio.txt","a");

for(POP = 0; POP < 4; POP++){

  switch(POP){
 case 0:
           dist="        N(0,1)      ";
            break;
     case 1:
           dist="0.9N(0,1)+0.1N(0,64)";
            break;
     case 2:
           dist="         t(1)       ";
            break;
     case 3:
           dist="         Chisq(1)   ";
            break;
     case 4:
           dist="         U(0,1)     ";
            break;
     default:
           printf("NO DISTRIBUTION DEFINED!!!!\n");
           break;
}


 for(SAM = 0; SAM < 4;SAM++){

   seed = 123475;

   switch(SAM){
case 0:
          n[0]=40; n[1]=40;
              break;
       case 1:
          n[0]=10; n[1]=10;
              break;
       case 2:
```

```
            n[0]=10; n[1]=20;
                break;
          case 3:
            n[0]=20; n[1]=40;
                break;
          default:
            break;
     }

    if(n[0] == n[2])
UE = 4;
    else
UE = 7;

      for(STD = 0; STD < UE; STD++){
        rep = 0;
        lev90=0; lev95=0; lev99=0;
        hdbf90=0; hdbf95=0; hdbf99=0;
        b90=0; b95=0; b99=0;


        switch(STD){
   case 0:
              std[0]=1; std[1]=1;
                  break;
           case 1:
              std[0]=1; std[1]=sqrt(2);
                  break;
           case 2:
              std[0]=1; std[1]=2;
                  break;
           case 3:
              std[0]=1; std[1]=sqrt(8);
                  break;
           case 4:
              std[1]=1; std[0]=sqrt(8);
                  break;
           case 5:
              std[1]=1; std[0]=2;
                  break;
           case 6:
              std[1]=1; std[0]=sqrt(2);
                  break;
           default:
              break;
}
```

```
fprintf(fout,"\n*****************************************\n");
fprintf(fout,"%26s\n",dist[POP]);
fprintf(fout,"Number of Groups = %d\n",GROUP);
for(j = 0; j < GROUP; j++)
fprintf(fout,"Variance%d = % 3.2f\t n%d = %d\n",
         j+1, std[j]*std[j], j+1, n[j]);
fprintf(fout,"*****************************************\n");

while( (rep >= 0) && (rep < ITER))
{

               /*  Sampleing */

  for(j = 0; j < GROUP; j++){
   imsls_random_seed_set(seed);

  switch(POP){
  case 0:
     imsls_d_random_uniform(n[j],IMSLS_RETURN_USER,unif[j],0);
  imsls_d_random_normal(n[j],IMSLS_RETURN_USER,x[j],0);
          for(k = 0; k < n[j]; k++){
      if( unif[j][k] > 0.9)
     x[j][k] = sqrt(8.0)*x[j][k];
      }
              break;
      case 1:
              imsls_d_random_normal(n[j],IMSLS_RETURN_USER,x[j],0);
              break;
      case 2:
  imsls_d_random_student_t(n[j],4,IMSLS_RETURN_USER,x[j],0);
              break;
      case 3:
              imsls_d_random_chi_squared(n[j],4,IMSLS_RETURN_USER,
                                                        x[j],0);
              break;
      default:
     break;
 }

 for(k=0; k < n[j]; k++)
   x[j][k] = x[j][k]*std[j];

 seed = imsls_random_seed_get();

}
```

```
  hdbf = hd(n,x);

  if( hdbf < 0)
   ;
  else{

   if( hdbf < 0.10 )
    hdbf90++;
   if( hdbf < 0.05 )
    hdbf95++;
   if( hdbf < 0.01 )
    hdbf99++;

   Bart = bart(n,x);

   if(Bart < 0.10)
    b90++;
   if(Bart < 0.05)
    b95++;
   if(Bart < 0.01)
    b99++;

/*
   Bart = Fratio(n,x);


   if(Bart < 0.05)
    b90++;
   if(Bart < 0.025)
    b95++;
   if(Bart < 0.005)
    b99++;
*/
   leven = LEVEN(n, x);

   if( leven < 0.10 )
    lev90++;
   if( leven < 0.05 )
    lev95++;
   if( leven < 0.01 )
    lev99++;

  rep++;

  printf("%d\n", rep);
 } /* ELSE */
```

```
} /* while */


  fprintf(fout,"TEST\t alpha10\t alpha05\t alpha01\n\n");
  fprintf(fout,"----------------------------------------\n");
  fprintf(fout,"F\t %f\t %f\t %f\n",b90/ITER,b95/ITER,b99/ITER);
  fprintf(fout,"BF\t %f\t %f\t %f\n",lev90/ITER,lev95/ITER,lev99/ITER);
  fprintf(fout,"HDBF\t %f\t %f\t %f\n",hdbf90/ITER,hdbf95/ITER,
                                                    hdbf99/ITER);
} /* SAM */
} /* STD */
} /* POP */

}   /* main */

/*************************************************************/
/*                                                           */
/*        Modified Levene's Test with WLE & HD               */
/*                                                           */
/*************************************************************/

double hd( int m[], double x[][SIZE])
{
   double mstar[GROUP][SIZE], fstar[GROUP][SIZE];
   double muhat[GROUP], sdHat[GROUP], h[GROUP];
   double muHD[GROUP], sdHD[GROUP];
//   double raf[GROUP][SIZE], delta[GROUP][SIZE];
   double  weight[GROUP][SIZE];
   double sum, expo, muBunja, Bunmo, sigmaBunja, diff;
   double demon, weighttmp, w[GROUP]={0.0}, wsum;
   int i, j, k, converge=0;
   double z[GROUP][SIZE], zBarBar, zbar[GROUP], nsum;
   double Leven_den, Leven_num , levenHD, w50HD, zbarsum;
   double *stat;
//   double y[GROUP][SIZE];
/*
  for( i = 0; i < GROUP; i++)
    muhat[i] = MEDIAN(x[i],m[i]);

  for(i = 0; i < GROUP; i++){
    for(j = 0; j < m[i]; j++){
       y[i][j] = fabs(x[i][j] - muhat[i]);
    }
    sdHat[i] =1.48* MEDIAN(y[i],m[i]);
    h[i] = BAND * MEDIAN(y[i],m[i]);
   }
```

```
*/
 for( i = 0; i < GROUP; i++){
   stat = imsls_d_simple_statistics(m[i],1,x[i],IMSLS_MEDIAN_AND_SCALE,0);
   muhat[i] = stat[14];
   sdHat[i] = stat[16];
   h[i] = BAND*stat[15];
 }



   /* Estimated Sample Densties */

   for(i = 0; i < GROUP; i++){
      for(j = 0; j < m[i]; j++){
       sum = 0.0;
         for(k = 0; k < m[i]; k++){
            expo=(x[i][k]-x[i][j])*(x[i][k]-x[i][j])/(2.0*h[i]*h[i]);
            sum += exp(-expo);
         }

         fstar[i][j] = (double) sum / (m[i]*sqrt(2.0*PI)*h[i]);
      }
   }



   /*  Model Smoothing */

for(  i = 0; i < GROUP; i++){
   do{
       demon = 2.0*(sdHat[i]*sdHat[i]+h[i]*h[i]);
       for(k = 0; k < m[i]; k++){
          expo = (x[i][k] - muhat[i])*(x[i][k] - muhat[i])/demon;
          if(fstar[i][k] < DBL_EPSILON)
            weight[i][k] = 0;
          else{
            mstar[i][k] = exp(-expo)/(sqrt(demon*PI));
             /*
              delta[i][k] =  fstar[i][k]/mstar[i][k] - 1.0;
               raf[i][k] = 2.0*(sqrt(delta[i][k]+1.0)-1.0);
               weighttmp = (raf[i][k]+1)/(delta[i][k]+1);
             */
            weighttmp = 2.0*sqrt(mstar[i][k]/fstar[i][k])
                          - mstar[i][k]/fstar[i][k];
```

```
            if(weighttmp < 0.0)
                weight[i][k] = 0.0;
            else if(weighttmp > 1.0)
               weight[i][k] = 1.0;
            else
                weight[i][k] =weighttmp;
            }
      }

        muBunja = 0.0;
        Bunmo = 0.0;

       for(j = 0; j < m[i]; j++){
          muBunja  += weight[i][j]*x[i][j];
          Bunmo += weight[i][j];
         }

         muHD[i] = muBunja / Bunmo;

         sigmaBunja = 0.0;
         for(k=0; k < m[i]; k++)
           sigmaBunja=sigmaBunja+weight[i][k]*(x[i][k]-muHD[i])
                      *(x[i][k]-muHD[i]);

         sdHD[i] = sqrt(sigmaBunja / Bunmo);
         diff =  (muhat[i] - muHD[i])*(muhat[i] - muHD[i])
                 + (sdHat[i]-sdHD[i])*(sdHat[i]-sdHD[i]);

    muhat[i] = muHD[i];
        sdHat[i] = sdHD[i];

        converge++;
        if(converge == NEH)
       return(-1);

  }while(diff > ERROR);
}

nsum = 0;
wsum=0;
for(i = 0; i < GROUP; i++){
  nsum += m[i];
  for( j =0; j < m[i]; j++){
   z[i][j] = fabs(x[i][j] - muHD[i]);
       w[i] += weight[i][j];
  }
  wsum += w[i];
```

```
 }

  zbarsum = 0;
  for(k = 0; k < GROUP; k++){
      zbar[k] = MEAN(z[k], m[k]);
      zbarsum += m[k]*zbar[k];
  }

  zBarBar = zbarsum / nsum ;

  Leven_num = 0.0;
  Leven_den = 0.0;

  for( i = 0; i < GROUP; i++){
    Leven_num = Leven_num + m[i]*(zbar[i] - zBarBar)*(zbar[i]-zBarBar);
    for(j = 0; j < m[i]; j++){
     Leven_den = Leven_den + (z[i][j] - zbar[i])*(z[i][j]-zbar[i]);
    }
  }

  w50HD = (Leven_num/(GROUP-1) ) / ( Leven_den / ( nsum - GROUP) );

  levenHD = 1.0 - imsls_d_F_cdf(w50HD,GROUP-1,nsum-GROUP);

   return(levenHD);

}



void bubble(double *array, const int size)
{
   int pass, j;

   void swap(double *, double *);

   for(pass = 1 ; pass < size; pass++)
     for(j = 0; j < size -1; j++)
       if(array[j] > array[j+1])
          swap(&array[j], &array[j+1]);
}


void swap(double *Ptr1, double *Ptr2)
{
   double temp;
```

```
    temp = *Ptr1;
    *Ptr1 = *Ptr2;
    *Ptr2 = temp;
}


double MEDIAN(double r[], const int k)
{
  int check, q;
  double med;

  bubble(r,k);

  check = k%2;

  if( check == 0){
   q = k/2;
   med = (r[q]+r[q+1])/2.0;
  }
  else{
   q = (k+1)/2;
   med =  r[q];
  }

  return(med);
}


double MEAN(double r[], const int k)
{
   int i;
   double sum = 0, mean;

   for( i = 0; i < k; i++)
    sum += r[i];

   mean = sum / (double) k;

   return(mean);
}


double VAR(double r[], const int k)
{
   int i;
   double sum = 0, ssum = 0, var, num;
```

```
   for( i = 0; i < k; i++){
      sum += r[i];
      ssum += r[i]*r[i];
    }

   num = ssum - ((sum*sum)/(double) k);
   var = num /((double) k - 1);

   return(var);
}

/*************************************************************/
/*                                                           */
/*        Bartlett's Test with WLE & HD                      */
/*                                                           */
/*************************************************************/

double bart(int m[], double z[][SIZE])
{
   double var[GROUP];
   double term = 0, nsum = 0, varbar = 0, B = 0, C, Bart;
   int i, k;

   for(i = 0; i < GROUP; i++){
      var[i] = VAR(z[i],m[i]);
      varbar = varbar + (m[i] - 1.0)*var[i];
      term = term + 1.0/( (double)m[i] - 1.0 );
      nsum = nsum + (double)m[i] - 1.0;
   }

   varbar = varbar / nsum;
   C = 1.0 + (term - 1.0/nsum)/(3*(GROUP-1));

   for( k = 0; k < GROUP; k++)
    B = B + (m[k]-1)*log(var[k]/varbar);

   Bart = 1.0 - imsls_d_chi_squared_cdf(-(B/C), GROUP-1);

   return(Bart);

}

/*************************************************************/
/*                                                           */
/*        Modified Levene's Test with Median                 */
/*         ( Brown & Forsythe )                              */
/*************************************************************/
```

```
double LEVEN(int n[], double x[][SIZE])
{
   double xmed[GROUP], zBarBar, zbar[GROUP];
   double  z[GROUP][SIZE],nsum, zbarsum;
   double  Leven_num, Leven_den, w50, leven;
   int i,j,k;

   for(j = 0; j < GROUP; j++)
xmed[j] = MEDIAN(x[j],n[j]);



  nsum = 0;

  for(i = 0; i < GROUP; i++){
   for(j = 0; j < n[i]; j++){
   z[i][j] = fabs(x[i][j] - xmed[i]);
   }
   nsum += n[i];
  }

  zbarsum = 0;
  for(k = 0; k < GROUP; k++){
   zbar[k] = MEAN(z[k],n[k]);
        zbarsum += n[k]*zbar[k];
  }


  zBarBar = zbarsum/nsum;

  Leven_num = 0.0;
  Leven_den = 0.0;

  for( i = 0; i < GROUP; i++){
   Leven_num = Leven_num + n[i]*(zbar[i] - zBarBar)*(zbar[i]-zBarBar);
   for(j = 0; j < n[i]; j++){
   Leven_den = Leven_den+(z[i][j] zbar[i])*(z[i][j]-zbar[i]);
   }
  }

  w50 = (Leven_num/(GROUP-1) ) / ( Leven_den / ( nsum - GROUP) );

   leven = 1.0 - imsls_d_F_cdf(w50,GROUP-1,nsum-GROUP);

   return(leven);
```

```
}


double Fratio(int m[], double z[][SIZE])
{
   double var[GROUP], fratio, f;

   var[0] = VAR(z[0],m[0]);
   var[1] = VAR(z[1],m[1]);

   f = var[0]/var[1];

   fratio = min( imsls_d_F_cdf(f, m[0]-1,m[1]-1),
                      (1.0-imsls_d_F_cdf(f, m[0]-1,m[1]-1)) );
   return(fratio);
}

/************************** END OF CODE ********************************/
```

APPENDIX L

C PROGRAM 4: EQUALITY OF THREE VARIANCES

```
#include <imsls.h>
#include <stdio.h>
#include <float.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>

#define PI M_PI
#define ERROR 0.000001
#define NEH 100
/*************************************************************/
/*                                                           */
/*        Program for Testing Equality of Variances          */
/*        When Group Size is  >= 3                           */
/*        Call Bartlett for more than two population         */
/*************************************************************/

/* Change the following number as needed */

#define GROUP   3          /* GROUP SIZE */
#define ITER 5000
#define SIZE 40        /* Largest sample Size */
#define BAND 0.5

FILE *fout;

void bubble(double *array, const int size);
double MEDIAN(double [], const int );
double MEAN(double [], const int);
double VAR(double [], const int);

main()
{
   double hd(int [], double [][SIZE]);
   double bart(int [], double [][SIZE]);
   double LEVEN(int [], double [][SIZE]);
//   double Fratio(int [], double [][SIZE]);

   int SAM, STD, n[GROUP];
   double std[GROUP];

   double unif[GROUP][SIZE];
   double x[GROUP][SIZE];
   static int seed;
   double leven, lev90, lev95, lev99;
   int j, k, rep, POP, UE;
   double hdbf, hdbf90, hdbf95, hdbf99;
```

```
   double Bart, b90, b95, b99;
   char *dist[26]={"0.9N(0,1)+0.1N(0,8)","N(0,1)","t(4)","Chisq(4)"};


   fout = fopen("3bart.txt","a");

for(POP = 0; POP < 4; POP++){
   for(SAM = 0; SAM < 5;SAM++){

     seed = 123475;

              /*************** K >= 3 **********************/


     switch(SAM){
case 0:
           n[0]=40; n[1]=40; n[2]=40;
               break;
        case 1:
           n[0]=10; n[1]=10; n[2]=10;
               break;
        case 2:
           n[0]=10; n[1]=10; n[2]=40;
               break;
        case 3:
           n[0]=20; n[1]=20; n[2]=40;
               break;
        case 4:
           n[0]=10; n[1]=20; n[2]=40;
               break;
        default:
           break;
     }

/*
switch(SAM){
   case 0:
    n[0]=40; n[1]=40;
      break;
   case 1:
    n[0]=10; n[1]=10;
      break;
   case 2:
    n[0]=10; n[1]=20;
      break;
   case 3:
    n[0]=20; n[1]=40;
```

```
      break;
   default:
    break;
}
*/


   if(n[0] == n[2])
UE = 4;
   else
UE = 7;

   for(STD = 0; STD < UE; STD++){
     rep = 0;
     lev90=0; lev95=0; lev99=0;
     hdbf90=0; hdbf95=0; hdbf99=0;
     b90=0; b95=0; b99=0;

            /************** K >= 3 ***************************/


     switch(STD){
case 0:
         std[0]=1; std[1]=1; std[2]=1;
             break;
         case 1:
          std[0]=1; std[1]=1; std[2]=sqrt(2);
              break;
         case 2:
          std[0]=1; std[1]=1; std[2]=2;
              break;
         case 3:
          std[0]=1; std[1]=sqrt(2); std[2]=2;
              break;
         case 4:
         std[0]=2; std[1]=sqrt(2); std[2]=1;
              break;
         case 5:
         std[0]=2; std[1]=1; std[2]=1;
              break;
         case 6:
         std[0]=sqrt(2); std[1]=1; std[2]=1;
              break;
        default:
         break;
      }
```

```
/*

switch(STD){
case 0:
    std[0]=4; std[1]=4;
        break;
   case 1:
    std[0]=4; std[1]=sqrt(8);
        break;
   case 2:
    std[0]=4; std[1]=2;
        break;
   case 3:
    std[0]=4; std[1]=sqrt(2);
        break;

// ONLY for Unequal sample Sizse

   case 4:
    std[1]=4; std[0]=sqrt(8);
        break;
   case 5:
    std[1]=4; std[0]=2;
        break;
   case 6:
    std[1]=4; std[0]=sqrt(2);
        break;

   default:
    break;
}
*/

  fprintf(fout,"\n*****************************************\n");
  fprintf(fout,"%26s\n",dist[POP]);
  fprintf(fout,"Number of Groups = %d\n",GROUP);
  for(j = 0; j < GROUP; j++)
  fprintf(fout,"Variance%d = % 3.2f\t n%d = %d\n",
           j+1, std[j]*std[j], j+1, n[j]);
  fprintf(fout,"*****************************************\n");

while( (rep >= 0) && (rep < ITER))
{

  /*  Sampleing */

for(j = 0; j < GROUP; j++){
```

```
   imsls_random_seed_set(seed);

 switch(POP){
    case 0:
     imsls_d_random_uniform(n[j],IMSLS_RETURN_USER,unif[j],0);
  imsls_d_random_normal(n[j],IMSLS_RETURN_USER,x[j],0);
          for(k = 0; k < n[j]; k++){
       if( unif[j][k] > 0.9)
         x[j][k] = sqrt(8.0)*x[j][k];
      }
          break;
    case 1:
               imsls_d_random_normal(n[j],IMSLS_RETURN_USER,x[j],0);
          break;
    case 2:
  imsls_d_random_student_t(n[j],4,IMSLS_RETURN_USER,x[j],0);
          break;
    case 3:

               imsls_d_random_chi_squared(n[j],4,IMSLS_RETURN_USER,
                                                          x[j],0);
          break;
    default:
         break;
 }

 for(k=0; k < n[j]; k++)
   x[j][k] = x[j][k]*std[j];

 seed = imsls_random_seed_get();

}

hdbf = hd(n,x);

if( hdbf < 0)
 ;
else{

 if( hdbf < 0.10 )
  hdbf90++;
 if( hdbf < 0.05 )
  hdbf95++;
 if( hdbf < 0.01 )
  hdbf99++;

 Bart = bart(n,x);
```

```
    if(Bart < 0.10)
     b90++;
    if(Bart < 0.05)
     b95++;
    if(Bart < 0.01)
     b99++;

/*
    Bart = Fratio(n,x);


    if(Bart < 0.05)
     b90++;
    if(Bart < 0.025)
     b95++;
    if(Bart < 0.005)
     b99++;
*/
    leven = LEVEN(n, x);

    if( leven < 0.10 )
     lev90++;
    if( leven < 0.05 )
     lev95++;
    if( leven < 0.01 )
     lev99++;

  rep++;

  printf("%d\n", rep);
 } /* ELSE */


} /* while */


  fprintf(fout,"TEST\t alpha10\t alpha05\t alpha01\n\n");
  fprintf(fout,"----------------------------------------\n");
  fprintf(fout,"BART\t %f\t %f\t %f\n",b90/ITER,b95/ITER,b99/ITER);
  fprintf(fout,"BF\t %f\t %f\t %f\n",lev90/ITER,lev95/ITER,lev99/ITER);
  fprintf(fout,"HDBF\t %f\t %f\t %f\n",hdbf90/ITER,hdbf95/ITER,
                                                    hdbf99/ITER);
} /* SAM */
} /* STD */
} /* POP */
```

```
}   /* main */

/*********************************************************/
/*                                                       */
/*        Modified Levene's Test with WLE & HD           */
/*                                                       */
/*********************************************************/

double hd( int m[], double x[][SIZE])
{
   double mstar[GROUP][SIZE], fstar[GROUP][SIZE];
   double muhat[GROUP], sdHat[GROUP], h[GROUP];
   double muHD[GROUP], sdHD[GROUP];
//   double raf[GROUP][SIZE], delta[GROUP][SIZE];
   double  weight[GROUP][SIZE];
   double sum, expo, muBunja, Bunmo, sigmaBunja, diff;
   double demon, weighttmp, w[GROUP]={0.0}, wsum;
   int i, j, k, converge=0;
   double z[GROUP][SIZE], zBarBar, zbar[GROUP], nsum;
   double Leven_den, Leven_num , levenHD, w50HD, zbarsum;
   double *stat;
//   double y[GROUP][SIZE];
/*
for( i = 0; i < GROUP; i++)
    muhat[i] = MEDIAN(x[i],m[i]);

for(i = 0; i < GROUP; i++){
      for(j = 0; j < m[i]; j++){
           y[i][j] = fabs(x[i][j] - muhat[i]);
          }
      sdHat[i] =1.48* MEDIAN(y[i],m[i]);
          h[i] = BAND * MEDIAN(y[i],m[i]);
        }


*/
  for( i = 0; i < GROUP; i++){
      stat = imsls_d_simple_statistics(m[i],1,x[i],
                                          IMSLS_MEDIAN_AND_SCALE,0);
          muhat[i] = stat[14];
          sdHat[i] = stat[16];
          h[i] = BAND*stat[15];
        }



   /* Estimated Sample Densties */
```

```
   for(i = 0; i < GROUP; i++){
      for(j = 0; j < m[i]; j++){
       sum = 0.0;
         for(k = 0; k < m[i]; k++){
            expo=(x[i][k]-x[i][j])*(x[i][k]-x[i][j])/(2.0*h[i]*h[i]);
            sum += exp(-expo);
         }

         fstar[i][j] = (double) sum / (m[i]*sqrt(2.0*PI)*h[i]);
      }
   }



   /*  Model Smoothing */

for(  i = 0; i < GROUP; i++){
   do{
       demon = 2.0*(sdHat[i]*sdHat[i]+h[i]*h[i]);
       for(k = 0; k < m[i]; k++){
          expo = (x[i][k] - muhat[i])*(x[i][k] - muhat[i])/demon;
          if(fstar[i][k] < DBL_EPSILON)
            weight[i][k] = 0;
          else{
            mstar[i][k] = exp(-expo)/(sqrt(demon*PI));
              /*
            delta[i][k] =  fstar[i][k]/mstar[i][k] - 1.0;
                raf[i][k] = 2.0*(sqrt(delta[i][k]+1.0)-1.0);
                weighttmp = (raf[i][k]+1)/(delta[i][k]+1);
              */
            weighttmp = 2.0*sqrt(mstar[i][k]/fstar[i][k])
                          - mstar[i][k]/fstar[i][k];

            if(weighttmp < 0.0)
              weight[i][k] = 0.0;
            else if(weighttmp > 1.0)
              weight[i][k] = 1.0;
            else
              weight[i][k] =weighttmp;
            }
      }

         muBunja = 0.0;
         Bunmo = 0.0;

       for(j = 0; j < m[i]; j++){
```

```
            muBunja  += weight[i][j]*x[i][j];
            Bunmo += weight[i][j];
          }

          muHD[i] = muBunja / Bunmo;

          sigmaBunja = 0.0;
          for(k=0; k < m[i]; k++)
            sigmaBunja=sigmaBunja+weight[i][k]*(x[i][k]-muHD[i])*
                                                    (x[i][k]-muHD[i]);

          sdHD[i] = sqrt(sigmaBunja / Bunmo);
          diff =  (muhat[i] - muHD[i])*(muhat[i] - muHD[i])
                  + (sdHat[i]-sdHD[i])*(sdHat[i]-sdHD[i]);

     muhat[i] = muHD[i];
          sdHat[i] = sdHD[i];

          converge++;
          if(converge == NEH)
          return(-1);

    }while(diff > ERROR);
  }

  nsum = 0;
  wsum=0;
  for(i = 0; i < GROUP; i++){
    nsum += m[i];
    for( j =0; j < m[i]; j++){
      z[i][j] = fabs(x[i][j] - muHD[i]);
      w[i] += weight[i][j];
    }
    wsum += w[i];
  }

   zbarsum = 0;
   for(k = 0; k < GROUP; k++){
     zbar[k] = MEAN(z[k], m[k]);
     zbarsum += m[k]*zbar[k];
   }

   zBarBar = zbarsum / nsum ;

   Leven_num = 0.0;
   Leven_den = 0.0;
```

```
  for( i = 0; i < GROUP; i++){
    Leven_num = Leven_num + m[i]*(zbar[i] - zBarBar)*(zbar[i]-zBarBar);
    for(j = 0; j < m[i]; j++){
      Leven_den = Leven_den + (z[i][j] - zbar[i])*(z[i][j]-zbar[i]);
   }
  }

  w50HD = (Leven_num/(GROUP-1) ) / ( Leven_den / ( nsum - GROUP) );

  levenHD = 1.0 - imsls_d_F_cdf(w50HD,GROUP-1,nsum-GROUP);

   return(levenHD);

}



void bubble(double *array, const int size)
{
   int pass, j;

   void swap(double *, double *);

   for(pass = 1 ; pass < size; pass++)
    for(j = 0; j < size -1; j++)
       if(array[j] > array[j+1])
          swap(&array[j], &array[j+1]);
}


void swap(double *Ptr1, double *Ptr2)
{
   double temp;

   temp = *Ptr1;
   *Ptr1 = *Ptr2;
   *Ptr2 = temp;
}

double MEDIAN(double r[], const int k)
{
  int check, q;
  double med;

  bubble(r,k);

  check = k%2;
```

```
  if( check == 0){
   q = k/2;
   med = (r[q]+r[q+1])/2.0;
  }
  else{
   q = (k+1)/2;
   med =  r[q];
  }

  return(med);
}

double MEAN(double r[], const int k)
{
   int i;
   double sum = 0, mean;

   for( i = 0; i < k; i++)
    sum += r[i];

   mean = sum / (double) k;

   return(mean);
}

double VAR(double r[], const int k)
{
   int i;
   double sum = 0, ssum = 0, var, num;

   for( i = 0; i < k; i++){
      sum += r[i];
      ssum += r[i]*r[i];
   }

   num = ssum - ((sum*sum)/(double) k);
   var = num /((double) k - 1);

   return(var);
}

/*************************************************************/
/*                                                           */
/*        Bartlett's Test with WLE & HD                      */
/*                                                           */
/*************************************************************/
```

```
double bart(int m[], double z[][SIZE])
{
   double var[GROUP];
   double term = 0, nsum = 0, varbar = 0, B = 0, C, Bart;
   int i, k;

   for(i = 0; i < GROUP; i++){
      var[i] = VAR(z[i],m[i]);
      varbar = varbar + (m[i] - 1.0)*var[i];
      term = term + 1.0/( (double)m[i] - 1.0 );
      nsum = nsum + (double)m[i] - 1.0;
   }

   varbar = varbar / nsum;
   C = 1.0 + (term - 1.0/nsum)/(3*(GROUP-1));

   for( k = 0; k < GROUP; k++)
    B = B + (m[k]-1)*log(var[k]/varbar);

   Bart = 1.0 - imsls_d_chi_squared_cdf(-(B/C), GROUP-1);

   return(Bart);

}

/*************************************************************/
/*                                                           */
/*        Modified Levene's Test with Median                 */
/*         ( Brown & Forsythe )                              */
/*************************************************************/

double LEVEN(int n[], double x[][SIZE])
{
   double xmed[GROUP], zBarBar, zbar[GROUP];
   double  z[GROUP][SIZE],nsum, zbarsum;
   double  Leven_num, Leven_den, w50, leven;
   int i,j,k;

   for(j = 0; j < GROUP; j++)
xmed[j] = MEDIAN(x[j],n[j]);



  nsum = 0;

  for(i = 0; i < GROUP; i++){
```

```
    for(j = 0; j < n[i]; j++){
      z[i][j] = fabs(x[i][j] - xmed[i]);
    }
    nsum += n[i];
  }

  zbarsum = 0;
  for(k = 0; k < GROUP; k++){
   zbar[k] = MEAN(z[k],n[k]);
        zbarsum += n[k]*zbar[k];
  }


  zBarBar = zbarsum/nsum;

  Leven_num = 0.0;
  Leven_den = 0.0;

  for( i = 0; i < GROUP; i++){
   Leven_num = Leven_num+n[i]*(zbar[i]-zBarBar)*(zbar[i]-zBarBar);
    for(j = 0; j < n[i]; j++){
      Leven_den = Leven_den + (z[i][j] - zbar[i])*(z[i][j]-zbar[i]);
    }
  }

  w50 = (Leven_num/(GROUP-1) ) / ( Leven_den / ( nsum - GROUP) );

   leven = 1.0 - imsls_d_F_cdf(w50,GROUP-1,nsum-GROUP);

   return(leven);

}


double Fratio(int m[], double z[][SIZE])
{
    double var[GROUP], fratio, f;

    var[0] = VAR(z[0],m[0]);
    var[1] = VAR(z[1],m[1]);

    f = var[0]/var[1];

    fratio = min( imsls_d_F_cdf(f, m[0]-1,m[1]-1),
                      (1.0-imsls_d_F_cdf(f, m[0]-1,m[1]-1)) );
    return(fratio);
}
```

```
/************************* END OF CODE ******************************/
```

APPENDIX M

C PROGRAM 5: EQUALITY OF THREE MEANS

```
/*************************************************************************/
/*                                                                       */
/*    THIS IS THE PROGRAM FOR ANOVA                                      */
/*                                                                       */
/*************************************************************************/

#include <imsls.h>
#include <stdio.h>
#include <float.h>
#include <math.h>
#include <time.h>

#define PI M_PI
#define ERROR 0.00001
#define NEH 100

/* Change the following number as needed */

#define GROUP   3          /* GROUP SIZE */
#define ITER 5000

#define SIZE 100        /* Largest sample Size */
// #define BAND 0.5
#define  MEAN 4
#define SAM 2
#define STD 3


FILE *fout;
int rep;


double MEDIAN(double r[], const int k)
{
void bubble(double *, const int);

  int check, q;
  double med;

  bubble(r,k);

  check = k%2;

  if( check == 0){
    q = k/2;
    med = (r[q]+r[q+1])/2.0;
  }
```

```
  else{
    q = (k+1)/2;
    med =  r[q];
  }

  return(med);
}



void main()
{

   int nn[SAM][GROUP]={{12,10,8},
                       {24,20,16}};
   double sd[STD][GROUP] ={{sqrt(11), sqrt(11), sqrt(11)},
                           {sqrt(71), sqrt(8) ,   4     },
                           {sqrt(12), sqrt(56), sqrt(15)}};
/* Availrable size for TIKU  5, 8, 10, 12, 16, 20, 24, 30 , > 30 */
   double AVG[MEAN][GROUP] ={{   0,      0,      0},
                             {-1.7,  0.536,  1.888},
     { 2.6, -0.800, -2.900},
                             { 2.5,  1.667, -5.833}};

   double avg[GROUP], std[GROUP];
   double *statistics;
   double random[GROUP][SIZE];
   double xbar[GROUP], svar[GROUP], w[GROUP], v[GROUP];
   double wSum, sum1, sum2, term, xBar;
   double bf1, bf2, bf3, bff, bfnume, bfdeno;
   double aa[GROUP], fdeno;
   double f1, f2 , f3;
   int i, j, k, o, n[GROUP];
   static int seed = 123457;
   double times, second;
   int minute;
   double bf90 = 0, bf95 = 0, bf99 = 0;
   double f90 = 0, f95 = 0, f99 = 0;
   double HDBF90=0, HDBF95=0, HDBF99=0;
   double TikuBF90=0, TikuBF95=0, TikuBF99=0;
   double BF, F, HD, Tiku;
   double sampleTot, anovaDeno;
   time_t lt;
   double hdbf(int [], double [][SIZE]);
   double tan(int [], double [][SIZE]);
   double *prob, var;      /* Mixture */
   int POP, SD, sam;
```

```
    char *dist[26]={"Normal","Mixture","Chi(3)","t(4)","U(-1,1)"};

    fout = fopen("3means.txt","a");


    lt = time(NULL);
    fprintf(fout,ctime(&lt));
    fprintf(fout,"\n\n");
   for(sam=1; sam <SAM;sam++){      ////
     for(k=0; k < GROUP; k++)
     n[k] = nn[sam][k];

   for(POP = 2; POP <5; POP++){      ////

   for(SD = 0; SD < STD; SD++){
     for( k = 0; k < GROUP; k++)
     std[k] = sd[SD][k];
   for( o=0; o < MEAN; o++){

   bf90 = 0, bf95 = 0, bf99 = 0;
   f90 = 0, f95 = 0, f99 = 0;
   HDBF90=0, HDBF95=0, HDBF99=0;
   TikuBF90=0, TikuBF95=0, TikuBF99=0;

   for( k = 0; k < GROUP; k++)
     avg[k] = AVG[o][k];

    sampleTot = 0.0;
    for( i = 0; i < GROUP; i++)
      sampleTot += n[i];

    rep=0;
    while( rep >=0 && rep < ITER)
    {

     switch(POP){

case 0:   /*************** Normal ***************************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
  imsls_d_random_normal(n[j],IMSLS_RETURN_USER,
                                        random[j],0);
       for(k = 0; k < n[j]; k++){
            random[j][k] = random[j][k]*std[j] + avg[j];
                  }
               seed = imsls_random_seed_get();
            }
```

```
        break;

        case 1:  /********** Mixture Distribution*************/
            for(j = 0; j < GROUP; j++)
            {
       imsls_random_seed_set(seed);
       imsls_d_random_normal(n[j],IMSLS_RETURN_USER,
                                        random[j],0);
                 prob = imsls_d_random_uniform(n[j],0);
                 var = (std[j]*std[j]-0.36)/0.325;
                    for(k = 0; k < n[j]; k++){
                  if( prob[k] > 0.9)
                    random[j][k] = random[j][k]*sqrt(var)+1.8;
                else
                 random[j][k] = random[j][k]*sqrt(0.25*var)-0.2;

                       random[j][k] = random[j][k]+avg[j];
                    }
                 seed = imsls_random_seed_get();
        }
        break;

        case 2: /********* Chi-Squared(3) ****************************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
       imsls_d_random_chi_squared(n[j],3,IMSLS_RETURN_USER,
                        random[j],0);
       for(k = 0; k < n[j]; k++){
            random[j][k] = (random[j][k]-3.0)*std[j]/sqrt(6.0)
                                + avg[j];
                 }
               seed = imsls_random_seed_get();
        }

break;

        case 3: /********* t(3) ****************************/
for(j = 0; j < GROUP; j++){
         imsls_random_seed_set(seed);
       imsls_d_random_student_t(n[j],4,IMSLS_RETURN_USER,
                             random[j],0);
       for(k = 0; k < n[j]; k++){
            random[j][k] = random[j][k]*std[j]/sqrt(2.0)
                                + avg[j];
                 }
                 seed = imsls_random_seed_get();
               }
```

```
break;

        case 4: /********* U(-1,1) ******************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
                  imsls_d_random_uniform(n[j],IMSLS_RETURN_USER,random[j]
                                         ,0);
      for(k = 0; k < n[j]; k++){
           random[j][k] = sqrt(3)*(2*random[j][k]-1)*std[j]+
                                    avg[j];
                 }
                 seed = imsls_random_seed_get();
                }
break;
        default:
   fprintf(fout,"No Population Defined\n");
        break;
  } /* switch(POP) */



   HD=hdbf(n,random);

   if(HD < 0.0)
          ;
   else{   /* HERE IS THE BIGGGGGGGGGGGGGGGGGGGGGGGGG LOOP */

   if( HD > 0.90 )
    HDBF90++;
   if( HD > 0.95 )
    HDBF95++;
   if( HD > 0.99 )
    HDBF99++;

   Tiku = tan(n,random);

   if( Tiku > 0.90)
    TikuBF90++;
   if( Tiku > 0.95)
    TikuBF95++;
   if(Tiku > 0.99)
    TikuBF99++;

   wSum = 0.0;
   sum1 = 0.0;
   sum2 = 0.0;
```

```
    for(j = 0; j < GROUP; j++)
    {
       statistics = imsls_d_simple_statistics(n[j],1,random[j],0);
       xbar[j] = statistics[0];
       svar[j] = statistics[1];
       v[j] = n[j] - 1;
       w[j] = (double) n[j] / svar[j];
       sum1 = sum1 + n[j]*xbar[j];
       wSum = w[j] + wSum;
       sum2 = xbar[j]*w[j] + sum2;
    }
    xBar = sum1 / sampleTot;

    term = 0.0;
    bfnume = 0.0;
    bfdeno = 0.0;
    anovaDeno = 0.0;
    for(i = 0; i < GROUP; i++){
       term = term + (1.0 - w[i]/wSum)*(1.0 - w[i]/wSum)/v[i];
       bfnume = bfnume  + n[i]*(xbar[i] - xBar)*(xbar[i] - xBar);
       aa[i] = (1 - (double)n[i]/sampleTot)*svar[i];
       bfdeno = bfdeno + aa[i];
       anovaDeno  += v[i]*svar[i];

    }



  /********* Brown and Forsythe Test *****************/

  BF = bfnume / bfdeno;

  fdeno = 0.0;
  for(k = 0; k < GROUP; k++)
   fdeno = fdeno + (double)(aa[k]*aa[k])/(n[k]-1.0);

//  bff = ceil( bfdeno*bfdeno/fdeno );
bff = (double) (bfdeno*bfdeno/fdeno);
  bf1 = imsls_d_F_inverse_cdf(0.90,GROUP-1,bff);
  bf2 = imsls_d_F_inverse_cdf(0.95,GROUP-1,bff);
  bf3 = imsls_d_F_inverse_cdf(0.99,GROUP-1,bff);

   if( BF > bf1 )
    bf90++;
   if( BF> bf2 )
    bf95++;
   if( BF > bf3 )
```

```
    bf99++;

    /********** ANOVA F  test *********************************/

    F = (bfnume*(sampleTot-GROUP)) / (anovaDeno*(GROUP-1));

    f1 = imsls_d_F_inverse_cdf(0.90,GROUP-1,sampleTot-GROUP);
    f2 = imsls_d_F_inverse_cdf(0.95,GROUP-1,sampleTot-GROUP);
    f3 = imsls_d_F_inverse_cdf(0.99,GROUP-1,sampleTot-GROUP);

    if( F > f1 )
     f90++;
    if( F > f2 )
     f95++;
    if( F > f3 )
     f99++;

    rep++;
    printf("%d\n", rep);
    } /* THIS IT END OF THE BIGGGGGGGGGGGGGGGGGGGGGG LOOP */
   }

   fprintf(fout,"****************************************\n");
   fprintf(fout,"%26s\n",dist[POP]);
   fprintf(fout,"Number of Groups = %d\n",GROUP);
   for(j = 0; j < GROUP; j++)
   fprintf(fout,"Mu%d = % 4.2f\t Variance%d = % 3.2f\t n%d = %d\n",
            j+1, avg[j], j+1, std[j]*std[j], j+1, n[j]);
   fprintf(fout,"****************************************\n");
   fprintf(fout, "\n");
   fprintf(fout,"TEST\t alpha10\t alpha05\t alpha01\n\n");
   fprintf(fout,"---------------------------------------\n");
   fprintf(fout,"ANOVA\t %f\t %f\t %f\n",f90/ITER, f95/ITER, f99/ITER);
   fprintf(fout,"B-F\t %f\t %f\t %f\n",bf90/ITER, bf95/ITER, bf99/ITER);
   fprintf(fout,"HDBF\t %f\t %f\t %f\n",HDBF90/ITER, HDBF95/ITER,
                HDBF99/ITER);
   fprintf(fout,"TanBF\t %f\t %f\t %f\n",TikuBF90/ITER, TikuBF95/ITER,
               TikuBF99/ITER);
} /* MEAN */
} /* SD */
} /*POP */
} /* SAM */


   times = imsls_ctime();
   minute = times/60;
   second = times - 60*minute;
```

```
  fprintf(fout,"\n\nCPU TIME USED is %d min %4.2f seconds\n", minute,
         second);

}

double hdbf( int m[], double x[][SIZE])
{
   double mstar[GROUP][SIZE], fstar[GROUP][SIZE], z[GROUP][SIZE];
   double muhat[GROUP], sdHat[GROUP], h[GROUP], correct[GROUP], c[GROUP];
   double muHD[GROUP], sdHD[GROUP], muHDall, numerator, denominator;
//   double raf[GROUP][SIZE], delta[GROUP][SIZE], weighttmp[GROUP][SIZE];
   double  weight[GROUP][SIZE], BFfdeno;
   double sum, expo, muBunja, Bunmo, sigmaBunja, diff;
   double hdBF, HDBF, demon;
   double wTot, allMu, wSum[GROUP]={0.0}, BFf;
   double MED, BAND;
   double mSum;
//   double *desc;
   int i, j, k, converge=0;


/*
   for(i = 0; i < GROUP; i++){
      desc=imsls_d_simple_statistics(m[i],1,x[i],IMSLS_MEDIAN_AND_SCALE,
                                     0);
      muhat[i] = desc[14];
      sdHat[i] = desc[16];
      h[i] = BAND * desc[15];
   }

*/


   for( i = 0; i < GROUP; i++){
      muhat[i] = MEDIAN(x[i],m[i]);
      for( j = 0; j < m[i]; j++){
        z[i][j] = fabs(x[i][j] - muhat[i]);
      }
      MED = MEDIAN(z[i],m[i]);
      sdHat[i] = 1.48258*MED;

      if(m[i]<30)
       BAND=0.9;
      else
       BAND=0.5;

      h[i] = BAND*MED;
```

```
   }

   /* Estimated Sample Densties */

   for(i = 0; i < GROUP; i++){
      for(j = 0; j < m[i]; j++){
       sum = 0.0;
         for(k = 0; k < m[i]; k++){
           expo=( (x[i][k]-x[i][j])* (x[i][k]-x[i][j]))/
                (2.0* (h[i])* (h[i]));
           sum += exp(-expo);
         }

         fstar[i][j] = (double) sum / (m[i]*sqrt(2.0*PI)*h[i]);
      }
    }



   /*  Model Smoothing */

  for(  i = 0; i < GROUP; i++){
   do{
       demon = 2.0*(sdHat[i]*sdHat[i]+h[i]*h[i]);
       for(k = 0; k < m[i]; k++)
       {
          expo =  (x[i][k] - muhat[i])*(x[i][k] - muhat[i])/demon ;
   mstar[i][k] = exp(-expo)/(sqrt(demon*PI));

          if((fstar[i][k] <= DBL_EPSILON))
             weight[i][k] = 0.0;
          else
          {
//           delta[i][k] =  fstar[i][k]/mstar[i][k] - 1.0;
//           raf[i][k] =  2.0 - (2.0+delta[i][k])*exp(-delta[i][k]);
//           raf[i][k] = 2.0*(sqrt(delta[i][k]+1.0)-1.0);
//           raf[i][k] = (raftmp < -1.0) ? -1.0:raftmp;
//           weighttmp[i][k] = (raf[i][k]+1)/(delta[i][k]+1);

            weight[i][k]=2.0*sqrt(mstar[i][k]/fstar[i][k])
                         -(mstar[i][k]/fstar[i][k]);
//           weight[i][k] = (raf[i][k]+1)/(delta[i][k]+1);
/*
            if(weighttmp < 0.0)
               weight[i][k] = 0.0;
            else if(weighttmp > 1.0)
               weight[i][k] = 1.0;
```

```
            else
               weight[i][k] =weighttmp;
*/
            }

            if(weight[i][k] <= 0.0)
                weight[i][k] = 0.0;
            else if(weight[i][k] >= 1.0)
                weight[i][k] = 1.0;
            else
                        ;



// fprintf(fout,"% f\t % f\t % f\t % f\t % f\t % f\t % f\n",x[i][k],
//      mstar[i][k],fstar[i][k],delta[i][k],raf[i][k],
//           weight[i][k],weighttmp[i][k]);

      }    /* do */


        muBunja = 0.0;
        sigmaBunja = 0.0;
        Bunmo = 0.0;

       for(j = 0; j < m[i]; j++){
          muBunja  += weight[i][j]*x[i][j];
          Bunmo += weight[i][j];
         }

         muHD[i] = muBunja / Bunmo;


        for(j = 0; j < m[i]; j++){
            sigmaBunja+=weight[i][j]*((x[i][j]-muHD[i])*(x[i][j]-muHD[i]));
         }

         sdHD[i] = sqrt(sigmaBunja / Bunmo);

         diff =  (muhat[i] - muHD[i])*(muhat[i] - muHD[i])
                 + (sdHat[i]-sdHD[i])*(sdHat[i]-sdHD[i]);


    muhat[i] = muHD[i];
        sdHat[i] = sdHD[i];

      converge++;
```

```
      if(converge == NEH)
    return(-1);

  }while(diff > ERROR);
 }

 /*************** Reweighting ************************/
/*
   for( k = 0; k < GROUP; k++){
   for( j = 0; j < m[k]; j++){
   weight[k][j] = sqrt(weight[k][j]);
   }
 }

 for( i = 0; i < GROUP; i++){

   muBunja = 0.0;
   Bunmo = 0.0;

   for(j = 0; j < m[i]; j++){
      muBunja  += weight[i][j]*x[i][j];
      Bunmo += weight[i][j];
   }

   if(Bunmo <= 0)
    muHD[i] = 0;
   else
    muHD[i] = muBunja / Bunmo;


   sigmaBunja = 0.0;
   for(j = 0; j < m[i]; j++){
     sigmaBunja+=weight[i][j]*(x[i][j]-muHD[i])*(x[i][j]-muHD[i]);
   }
   if( Bunmo <= 0 || sigmaBunja <= 0)
    sdHD[i] = DBL_EPSILON;
   else
        sdHD[i] = sqrt(sigmaBunja / Bunmo);
 }
*/
 /*************** End of Reweighting ***********************/

  allMu = 0.0;
  wTot = 0.0;
  mSum = 0.0;

  for( i = 0; i < GROUP; i++){
```

```
    for( j = 0; j < m[i]; j++){
    allMu += weight[i][j]*x[i][j];
        wSum[i] += weight[i][j];   /* flag */
   }
   wTot += wSum[i];
   sdHD[i] = sqrt( wSum[i]*sdHD[i]*sdHD[i] / (wSum[i]-1.0) );
   mSum += m[i];
  }

  for(k = 0; k < GROUP; k++){
    correct[k] = (1.0 - (m[k] / mSum))*(sdHD[k]*sdHD[k]);
//   correct[k] = (1.0 - (wSum[k] / wTot))*(sdHD[k]*sdHD[k]);
   }

  if( wTot <= 0)
    muHDall = 0.0;
  else
    muHDall = allMu / wTot;

  numerator = 0.0;
  denominator  = 0.0;

  for(i = 0; i < GROUP; i++){
    numerator += m[i]*(muHD[i] - muHDall)* (muHD[i] - muHDall) ;
//   numerator += wSum[i]*(muHD[i] - muHDall)* (muHD[i] - muHDall) ;
    denominator += correct[i];
  }

 BFfdeno = 0.0;
 for(i = 0; i < GROUP; i++){
   c[i] = correct[i] / denominator;
   BFfdeno += (c[i]*c[i])/(m[i]- 1.0);
 }


   BFf = (double) (1.0/BFfdeno);
//  BFf = (ceil) (1.0/BFfdeno);

   HDBF = (numerator)/(denominator);
   hdBF = imsls_d_F_cdf(HDBF,GROUP-1, BFf);

   return(hdBF);
}

/*********************************************************************/
/* Calculation of Tiku's Modified MLE Estimate and Test              */
/*********************************************************************/
```

```
double tan(int n[], double x[][SIZE])
{

   void bubble(double *, const int);

   double A[GROUP]={0}, B, C, D, K, m[GROUP],  M1, M2, d, Tan, tan;
   double alpha1, beta1, alpha2, beta2, alpha, beta;
   double muTiku[GROUP], sigmaTiku[GROUP], grandmu;
   double temp, tmp, ran[SIZE+1], dof;
   int choice, mTan;
   int n1, r1, r2, i, j,  k,l;

   mTan=0;
   grandmu = 0.0;
   for(i = 0; i < GROUP; i++)
   {
     choice = n[i];
     switch(choice)
     {
    case 5:
            alpha = 0.7540;
            beta = 0.8108;
            break;
       case 8:
            alpha = 0.6968;
            beta = 0.8481;
            break;
        case 10:
            alpha = 0.6737;
            beta = 0.8610;
            break;
       case 12:
            alpha = 0.6564;
            beta = 0.8700;
           break;
       case 16:
            alpha = 0.7048;
            beta = 0.8274;
           break;
       case 20:
           alpha = 0.6864;
            beta = 0.8401;
            break;
       case 24:
```

```
            alpha = 0.6724;
             beta = 0.8489;
             break;
        case 30:
             alpha = 0.6880;
             beta = 0.8365;
             break;
        default:
            alpha = 0.6902;
             beta = 0.8309;
            break;
}

    /* Keep the subscripts the same as in the formula */

alpha1 = alpha;  beta1 = beta ;
alpha2 = alpha;  beta2 = beta ;

bubble(&x[i][0],n[i]);

for( k = 0; k < n[i]; k++)
   ran[k+1] = x[i][k];

n1 = n[i]; ;
r1 = (int) floor(0.5+0.1*n1);
r2 = (int) floor(0.5+0.1*n1);
A[i] = n1 - r1 -r2;
m[i] = n1 - r1 - r2 + r1*beta1 + r2*beta2;
D = ( r2*alpha2 - r1*alpha1 )/ m[i];

tmp = 0.0;
for( j = r1+1; j <= n1-r2; j++)
 tmp += ran[j];

K = ( tmp + r1*beta1*ran[r1+1] + r2*beta2*ran[n1-r2] )/m[i];
B = r2*alpha2*(ran[n1-r2] - K) - r1*alpha1*(ran[r1+1] - K);

temp = 0.0;
for( l = r1+1; l <= n1-r2; l++)
 temp += ran[l]*ran[l];

C = temp + r1*beta1*ran[r1+1]*ran[r1+1]
          + r2*beta2*ran[n1-r2]*ran[n1-r2] - m[i]*K*K;

sigmaTiku[i] = (B+sqrt(B*B+4.0*A[i]*C))/(2.0*sqrt(A[i]*(A[i]-1)));
muTiku[i] = K + D*sigmaTiku[i];
grandmu +=  m[i]*muTiku[i];
```

```
    mTan += m[i];

  }

   grandmu = grandmu/(double) mTan;

   M1 = 0.0;
   M2 = 0.0;
   d = 0.0;
   for(j = 0; j < GROUP; j++){
   M1 += m[j]*(muTiku[j] - grandmu)*(muTiku[j] - grandmu);
   M2 += (1.0 - (double)m[j]/mTan)*sigmaTiku[j]*sigmaTiku[j];
   d +=  (1.0 - (double)m[j]/mTan)*(1.0-(double)m[j]/mTan)
                    *pow(sigmaTiku[j], 4)/(A[j]-1);
   }

   dof = (double) ((M2*M2)/d);
//  dof = ceil((M2*M2)/d);

   Tan = M1/M2;

   tan = imsls_d_F_cdf(Tan, GROUP-1, dof);
   return(tan);

}


void bubble(double *array, const int size)
{
    int pass, j;

    void swap(double *, double *);

    for(pass = 1 ; pass < size; pass++)
      for(j = 0; j < size -1; j++)
        if(array[j] > array[j+1])
           swap(&array[j], &array[j+1]);
}


void swap(double *Ptr1, double *Ptr2)
{
    double temp;

    temp = *Ptr1;
    *Ptr1 = *Ptr2;
    *Ptr2 = temp;
```

APPENDIX N

C PROGRAM 6: EQUALITY OF FOUR MEANS

```
/*************************************************************************/
/*                                                                       */
/*    THIS IS THE PROGRAM FOR ANOVA  (4 Populations)                     */
/*    ( Code differ from tanF)                                           */
/*************************************************************************/

#include <imsls.h>
#include <stdio.h>
#include <float.h>
#include <math.h>
#include <time.h>

#define PI M_PI
#define ERROR 0.00001
#define NEH 100

/* Change the following number as needed */

#define GROUP  4          /* GROUP SIZE */
#define ITER 5000

#define SIZE 100        /* Largest sample Size */
// #define BAND 0.5
#define MEAN 3
// #define SAM 3
#define STD 3


FILE *fout;
int rep;


double MEDIAN(double r[], const int k)
{
void bubble(double *, const int);

  int check, q;
  double med;

  bubble(r,k);

  check = k%2;

  if( check == 0){
   q = k/2;
   med = (r[q]+r[q+1])/2.0;
  }
```

```
  else{
   q = (k+1)/2;
   med =  r[q];
  }

  return(med);
}



void main()
{
/*
   int nn[SAM][GROUP]={{11,11,11,11},
                       {11,16,16,21},
                       {50,50,50,50}};
*/
int n[GROUP]={50,50,50,50};
/* Equal sample Size */
   double sd[STD][GROUP] ={{1,1,1,1},
       {4,1,1,1},
                           {4,3,2,1}};

/*     Unequa sample sizes */
/*
   double sd[STD][GROUP] ={{1,1,1,1},
       {4,1,1,1},
                           {1,1,1,4},
                           {4,3,2,1},
                           {1,2,3,4}};
*/
/* Availrable size for TIKU  5, 8, 10, 12, 16, 20, 24, 30 , > 30 */
   double AVG[MEAN][GROUP] ={{0,0,0,0},
                             {0.5,0,0,0},
                             {0,0,0,0.7}};


   double avg[GROUP], std[GROUP];
   double *statistics;
   double random[GROUP][SIZE];
   double xbar[GROUP], svar[GROUP], w[GROUP], v[GROUP];
   double wSum, sum1, sum2, term, xBar;
   double bf1, bf2, bf3, bff, bfnume, bfdeno;
   double aa[GROUP], fdeno;
   double f1, f2 , f3;
   int i, j, k, o;
   static int seed = 123457;
```

```
 double times, second;
 int minute;
 double bf90 = 0, bf95 = 0, bf99 = 0;
 double f90 = 0, f95 = 0, f99 = 0;
 double HDBF90=0, HDBF95=0, HDBF99=0;
 double TikuBF90=0, TikuBF95=0, TikuBF99=0;
 double BF, F, HD, Tiku;
 double sampleTot, anovaDeno;
 time_t lt;
 double hdbf(int [], double [][SIZE]);
 double tan(int [], double [][SIZE]);
 double *prob;      /* Mixture */
 int POP, SD;
 char *dist[26]={"Normal","Mixture","Chi(3)","t(4)","U(-1,1)"};

 fout = fopen("4means2.txt","a");


 lt = time(NULL);
 fprintf(fout,ctime(&lt));
 fprintf(fout,"\n\n");

for(POP = 2; POP <5; POP++){      ////

  for(SD = 0; SD < STD; SD++){

 for( k = 0; k < GROUP; k++){
     std[k] = sd[SD][k];
  }

  for( o=0; o < MEAN; o++){

   bf90 = 0, bf95 = 0, bf99 = 0;
   f90 = 0, f95 = 0, f99 = 0;
   HDBF90=0, HDBF95=0, HDBF99=0;
   TikuBF90=0, TikuBF95=0, TikuBF99=0;

for( k = 0; k < GROUP; k++){
    avg[k] = AVG[o][k];
  }

  sampleTot = 0.0;
  for( i = 0; i < GROUP; i++)
  sampleTot += n[i];

  rep=0;
  while( rep >=0 && rep < ITER)
```

```
    {

      switch(POP){

case 0:   /*************** Normal ***************************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
  imsls_d_random_normal(n[j],IMSLS_RETURN_USER,random[j],0);
        for(k = 0; k < n[j]; k++){
       random[j][k] = random[j][k]*std[j] + avg[j];
                   }
                seed = imsls_random_seed_get();
             }
                break;

        case 1: /**********  Mixture Distribution*************/
           for(j = 0; j < GROUP; j++)
           {
     imsls_random_seed_set(seed);
     imsls_d_random_normal(n[j],IMSLS_RETURN_USER,random[j],0);
               prob = imsls_d_random_uniform(n[j],0);
                for(k = 0; k < n[j]; k++){
            if( prob[k] > 0.9)
                random[j][k] = 8.0*random[j][k]*std[j]+avg[j];
          else
             random[j][k] = random[j][k]*std[j]+avg[j];
              }
             seed = imsls_random_seed_get();
       }
             break;

       case 2: /********* Chi-Squared(3) ***************************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
      imsls_d_random_chi_squared(n[j],3,IMSLS_RETURN_USER,
                                                 random[j],0);
      for(k = 0; k < n[j]; k++){
           random[j][k] = (random[j][k]-3.0)*std[j] + avg[j];
                 }
                 seed = imsls_random_seed_get();
        }

                 break;

      case 3: /********* t(3) ***************************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
```

```
      imsls_d_random_student_t(n[j],4,IMSLS_RETURN_USER,
                                                          random[j],0);
      for(k = 0; k < n[j]; k++){
            random[j][k] = random[j][k]*std[j] + avg[j];
                 }
                 seed = imsls_random_seed_get();
                }
                break;

      case 4: /********* U(-1,1) ******************/
for(j = 0; j < GROUP; j++){
       imsls_random_seed_set(seed);
                  imsls_d_random_uniform(n[j],IMSLS_RETURN_USER,
                                                          random[j],0);
      for(k = 0; k < n[j]; k++){
           random[j][k] = (2*random[j][k]-1)*std[j]+avg[j];
                 }
                 seed = imsls_random_seed_get();
                }
                break;
      default:
   fprintf(fout,"No Population Defined\n");
                break;
  } /* switch(POP) */



   HD=hdbf(n,random);

   if(HD < 0.0)
          ;
   else{   /* HERE IS THE BIGGGGGGGGGGGGGGGGGGGGGGGG LOOP */

   if( HD > 0.90 )
    HDBF90++;
   if( HD > 0.95 )
    HDBF95++;
   if( HD > 0.99 )
    HDBF99++;

   Tiku = tan(n,random);

   if( Tiku > 0.90)
    TikuBF90++;
   if( Tiku > 0.95)
    TikuBF95++;
   if(Tiku > 0.99)
```

```
    TikuBF99++;

   wSum = 0.0;
   sum1 = 0.0;
   sum2 = 0.0;

   for(j = 0; j < GROUP; j++)
   {
      statistics = imsls_d_simple_statistics(n[j],1,random[j],0);
      xbar[j] = statistics[0];
      svar[j] = statistics[1];
      v[j] = n[j] - 1;
      w[j] = (double) n[j] / svar[j];
      sum1 = sum1 + n[j]*xbar[j];
      wSum = w[j] + wSum;
      sum2 = xbar[j]*w[j] + sum2;
   }
   xBar = sum1 / sampleTot;

   term = 0.0;
   bfnume = 0.0;
   bfdeno = 0.0;
   anovaDeno = 0.0;
   for(i = 0; i < GROUP; i++){
      term = term + (1.0 - w[i]/wSum)*(1.0 - w[i]/wSum)/v[i];
      bfnume = bfnume  + n[i]*(xbar[i] - xBar)*(xbar[i] - xBar);
      aa[i] = (1 - (double)n[i]/sampleTot)*svar[i];
      bfdeno = bfdeno + aa[i];
      anovaDeno  += v[i]*svar[i];

   }



  /********* Brown and Forsythe Test *****************/

  BF = bfnume / bfdeno;

  fdeno = 0.0;
  for(k = 0; k < GROUP; k++)
   fdeno = fdeno + (double)(aa[k]*aa[k])/(n[k]-1.0);

//  bff = ceil( bfdeno*bfdeno/fdeno );
bff = (double) (bfdeno*bfdeno/fdeno);
  bf1 = imsls_d_F_inverse_cdf(0.90,GROUP-1,bff);
  bf2 = imsls_d_F_inverse_cdf(0.95,GROUP-1,bff);
  bf3 = imsls_d_F_inverse_cdf(0.99,GROUP-1,bff);
```

```
    if( BF > bf1 )
     bf90++;
    if( BF> bf2 )
     bf95++;
    if( BF > bf3 )
     bf99++;

    /********** ANOVA F  test *********************************/

    F = (bfnume*(sampleTot-GROUP)) / (anovaDeno*(GROUP-1));

    f1 = imsls_d_F_inverse_cdf(0.90,GROUP-1,sampleTot-GROUP);
    f2 = imsls_d_F_inverse_cdf(0.95,GROUP-1,sampleTot-GROUP);
    f3 = imsls_d_F_inverse_cdf(0.99,GROUP-1,sampleTot-GROUP);

    if( F > f1 )
     f90++;
    if( F > f2 )
     f95++;
    if( F > f3 )
     f99++;

  rep++;
  printf("%d\n", rep);
  } /* THIS IT END OF THE BIGGGGGGGGGGGGGGGGGGGGGG LOOP */
  }

  fprintf(fout,"*****************************************\n");
  fprintf(fout,"%26s\n",dist[POP]);
  fprintf(fout,"Number of Groups = %d\n",GROUP);
  for(j = 0; j < GROUP; j++)
  fprintf(fout,"Mu%d = % 4.2f\t Variance%d = % 3.2f\t n%d = %d\n",
           j+1, avg[j], j+1, std[j]*std[j], j+1, n[j]);
  fprintf(fout,"*****************************************\n");
  fprintf(fout, "\n");
  fprintf(fout,"TEST\t alpha10\t alpha05\t alpha01\n\n");
  fprintf(fout,"----------------------------------------\n");
  fprintf(fout,"ANOVA\t %f\t %f\t %f\n",f90/ITER, f95/ITER, f99/ITER);
  fprintf(fout,"B-F\t %f\t %f\t %f\n",bf90/ITER, bf95/ITER, bf99/ITER);
  fprintf(fout,"HDBF\t %f\t %f\t %f\n",HDBF90/ITER, HDBF95/ITER,
                                                    HDBF99/ITER);
  fprintf(fout,"TanBF\t %f\t %f\t %f\n",TikuBF90/ITER, TikuBF95/ITER,
              TikuBF99/ITER);
} /* MEAN */
} /* SD */
} /*POP */
```

```
  times = imsls_ctime();
  minute = times/60;
  second = times - 60*minute;
 fprintf(fout,"\n\nCPU TIME USED is %d min %4.2f seconds\n", minute, second);

}

double hdbf( int m[], double x[][SIZE])
{
   double mstar[GROUP][SIZE], fstar[GROUP][SIZE], z[GROUP][SIZE];
   double muhat[GROUP], sdHat[GROUP], h[GROUP], correct[GROUP], c[GROUP];
   double muHD[GROUP], sdHD[GROUP], muHDall, numerator, denominator;
//   double raf[GROUP][SIZE], delta[GROUP][SIZE], weighttmp[GROUP][SIZE];
   double  weight[GROUP][SIZE], BFfdeno;
   double sum, expo, muBunja, Bunmo, sigmaBunja, diff;
   double hdBF, HDBF, demon;
   double wTot, allMu, wSum[GROUP]={0.0}, BFf;
   double MED, BAND;
   double mSum;
//   double *desc;
   int i, j, k, converge=0;


/*
   for(i = 0; i < GROUP; i++){
      desc=imsls_d_simple_statistics(m[i],1,x[i],IMSLS_MEDIAN_AND_SCALE,0);
      muhat[i] = desc[14];
      sdHat[i] = desc[16];
      h[i] = BAND * desc[15];
   }

*/


   for( i = 0; i < GROUP; i++){
     muhat[i] = MEDIAN(x[i],m[i]);
     for( j = 0; j < m[i]; j++){
       z[i][j] = fabs(x[i][j] - muhat[i]);
     }
     MED = MEDIAN(z[i],m[i]);
     sdHat[i] = 1.48258*MED;

      if(m[i]<30)
       BAND=0.9;
```

```
      else
       BAND=0.5;

      h[i] = BAND*MED;
//   fprintf(fout,"muHat[%d]=% f\t sdHat[%d]=% f\t h[%d]=% f\n",i,muhat[i],
//   i,sdHat[i],i,h[i]);
   }

   /* Estimated Sample Densties */

   for(i = 0; i < GROUP; i++){
      for(j = 0; j < m[i]; j++){
       sum = 0.0;
         for(k = 0; k < m[i]; k++){
           expo=( (x[i][k]-x[i][j])* (x[i][k]-x[i][j]))/
            (2.0* (h[i])* (h[i]));
            sum += exp(-expo);
         }

         fstar[i][j] = (double) sum / (m[i]*sqrt(2.0*PI)*h[i]);
      }
   }



   /*  Model Smoothing */

  for(  i = 0; i < GROUP; i++){
     do{
         demon = 2.0*(sdHat[i]*sdHat[i]+h[i]*h[i]);
         for(k = 0; k < m[i]; k++)
         {
           expo =  (x[i][k] - muhat[i])*(x[i][k] - muhat[i])/demon ;
     mstar[i][k] = exp(-expo)/(sqrt(demon*PI));
          if((fstar[i][k] <= DBL_EPSILON))
            weight[i][k] = 0.0;
           else
           {
//             delta[i][k] =  fstar[i][k]/mstar[i][k] - 1.0;
//              raf[i][k] =  2.0 - (2.0+delta[i][k])*exp(-delta[i][k]);
//              raf[i][k] = 2.0*(sqrt(delta[i][k]+1.0)-1.0);
//              raf[i][k] = (raftmp < -1.0) ? -1.0:raftmp;
//              weighttmp[i][k] = (raf[i][k]+1)/(delta[i][k]+1);

            weight[i][k]=2.0*sqrt(mstar[i][k]/fstar[i][k])
                      -(mstar[i][k]/fstar[i][k]);
//              weight[i][k] = (raf[i][k]+1)/(delta[i][k]+1);
```

```
/*
              if(weighttmp < 0.0)
               weight[i][k] = 0.0;
              else if(weighttmp > 1.0)
               weight[i][k] = 1.0;
              else
               weight[i][k] =weighttmp;
*/
            }

            if(weight[i][k] <= 0.0)
                weight[i][k] = 0.0;
            else if(weight[i][k] >= 1.0)
                weight[i][k] = 1.0;
            else
                          ;



// fprintf(fout,"% f\t % f\t % f\t % f\t % f\t % f\t % f\n",x[i][k],
//            mstar[i][k],fstar[i][k],delta[i][k],raf[i][k],weight[i][k],
//                                                    weighttmp[i][k]);

      }    /* do */


       muBunja = 0.0;
       sigmaBunja = 0.0;
       Bunmo = 0.0;

      for(j = 0; j < m[i]; j++){
        muBunja  += weight[i][j]*x[i][j];
        Bunmo += weight[i][j];
      }

      muHD[i] = muBunja / Bunmo;


      for(j = 0; j < m[i]; j++){
         sigmaBunja+=weight[i][j]*( (x[i][j]-muHD[i])* (x[i][j]-muHD[i]));
       }

      sdHD[i] = sqrt(sigmaBunja / Bunmo);

      diff =  (muhat[i] - muHD[i])*(muhat[i] - muHD[i])
                + (sdHat[i]-sdHD[i])*(sdHat[i]-sdHD[i]);
```

```
//   fprintf(fout,"muHD[%d]=% f\t sdHD[%d]=% f\n", i,muHD[i],i,sdHD[i]);

     muhat[i] = muHD[i];
     sdHat[i] = sdHD[i];

     converge++;
     if(converge == NEH)
   return(-1);

 }while(diff > ERROR);
}

/*************** Reweighting ************************/
/*
for( k = 0; k < GROUP; k++){
  for( j = 0; j < m[k]; j++){
  weight[k][j] = sqrt(weight[k][j]);
  }
}

for( i = 0; i < GROUP; i++){

  muBunja = 0.0;
  Bunmo = 0.0;

  for(j = 0; j < m[i]; j++){
     muBunja  += weight[i][j]*x[i][j];
     Bunmo += weight[i][j];
  }

  if(Bunmo <= 0)
   muHD[i] = 0;
  else
   muHD[i] = muBunja / Bunmo;


  sigmaBunja = 0.0;
  for(j = 0; j < m[i]; j++){
    sigmaBunja+=weight[i][j]*(x[i][j]-muHD[i])*(x[i][j]-muHD[i]);
  }
    if( Bunmo <= 0 || sigmaBunja <= 0)
     sdHD[i] = DBL_EPSILON;
    else
       sdHD[i] = sqrt(sigmaBunja / Bunmo);
}
*/
/*************** End of Reweighting ***********************/
```

```
  allMu = 0.0;
  wTot = 0.0;
  mSum = 0.0;

  for( i = 0; i < GROUP; i++){
    for( j = 0; j < m[i]; j++){
    allMu += weight[i][j]*x[i][j];
        wSum[i] += weight[i][j];   /* flag */
     }
    wTot += wSum[i];
    sdHD[i] = sqrt( wSum[i]*sdHD[i]*sdHD[i] / (wSum[i]-1.0) ); /*flag*/
    mSum += m[i];
  }

  for(k = 0; k < GROUP; k++){
    correct[k] = (1.0 - (m[k] / mSum))*(sdHD[k]*sdHD[k]);
//   correct[k] = (1.0 - (wSum[k] / wTot))*(sdHD[k]*sdHD[k]);
   }

  if( wTot <= 0)
    muHDall = 0.0;
  else
    muHDall = allMu / wTot;

  numerator = 0.0;
  denominator  = 0.0;

  for(i = 0; i < GROUP; i++){
   numerator += m[i]*(muHD[i] - muHDall)* (muHD[i] - muHDall) ;
//   numerator += wSum[i]*(muHD[i] - muHDall)* (muHD[i] - muHDall) ;
   denominator += correct[i];
  }

 BFfdeno = 0.0;
 for(i = 0; i < GROUP; i++){
   c[i] = correct[i] / denominator;
   BFfdeno += (c[i]*c[i])/(m[i]- 1.0);
 }


  BFf = (double) (1.0/BFfdeno);
//  BFf = (ceil) (1.0/BFfdeno);

  HDBF = (numerator)/(denominator);
  hdBF = imsls_d_F_cdf(HDBF,GROUP-1, BFf);
```

```
   return(hdBF);
}

/*************************************************************************/
/* Calculation of Tiku's Modified MLE Estimate and  Tan-Tabatabai Test */
/*************************************************************************/



double tan(int n[], double x[][SIZE])
{

   void bubble(double *, const int);

   double A[GROUP]={0}, B, C, D, K, m[GROUP],  M1, M2, d, Tan, tan;
   double alpha1, beta1, alpha2, beta2, alpha, beta;
   double muTiku[GROUP], sigmaTiku[GROUP], grandmu;
   double temp, tmp, ran[SIZE+1], dof;
   int choice, mTan;
   int n1, r1, r2, i, j,  k,l;

   mTan=0;
   grandmu = 0.0;
   for(i = 0; i < GROUP; i++)
   {
     choice = n[i];
     switch(choice)
     {
    case 5:
            alpha = 0.7540;
            beta = 0.8108;
            break;
       case 8:
            alpha = 0.6968;
            beta = 0.8481;
            break;
         case 10:
            alpha = 0.6737;
            beta = 0.8610;
            break;
       case 12:
            alpha = 0.6564;
            beta = 0.8700;
           break;
       case 16:
            alpha = 0.7048;
            beta = 0.8274;
```

```
        break;
    case 20:
        alpha = 0.6864;
         beta = 0.8401;
         break;
    case 24:
        alpha = 0.6724;
         beta = 0.8489;
         break;
    case 30:
         alpha = 0.6880;
         beta = 0.8365;
         break;
    default:
        alpha = 0.6902;
         beta = 0.8309;
        break;
}

    /* re-indecies to keep the subscripts in the formulat */

alpha1 = alpha;  beta1 = beta ;
alpha2 = alpha;  beta2 = beta ;

bubble(&x[i][0],n[i]);

for( k = 0; k < n[i]; k++)
   ran[k+1] = x[i][k];

n1 = n[i]; ;
r1 = (int) floor(0.5+0.1*n1);
r2 = (int) floor(0.5+0.1*n1);
A[i] = n1 - r1 -r2;
m[i] = n1 - r1 - r2 + r1*beta1 + r2*beta2;
D = ( r2*alpha2 - r1*alpha1 )/ m[i];

tmp = 0.0;
for( j = r1+1; j <= n1-r2; j++)
 tmp += ran[j];

K = ( tmp + r1*beta1*ran[r1+1] + r2*beta2*ran[n1-r2] )/m[i];
B = r2*alpha2*(ran[n1-r2] - K) - r1*alpha1*(ran[r1+1] - K);

temp = 0.0;
for( l = r1+1; l <= n1-r2; l++)
 temp += ran[l]*ran[l];
```

```
    C = temp + r1*beta1*ran[r1+1]*ran[r1+1]
             + r2*beta2*ran[n1-r2]*ran[n1-r2] - m[i]*K*K;

    sigmaTiku[i] = (B+sqrt(B*B+4.0*A[i]*C))/(2.0*sqrt(A[i]*(A[i]-1)));
    muTiku[i] = K + D*sigmaTiku[i];
    grandmu +=  m[i]*muTiku[i];
    mTan += m[i];

  }

   grandmu = grandmu/(double) mTan;

   M1 = 0.0;
   M2 = 0.0;
   d = 0.0;
   for(j = 0; j < GROUP; j++){
   M1 += m[j]*(muTiku[j] - grandmu)*(muTiku[j] - grandmu);
   M2 += (1.0 - (double)m[j]/mTan)*sigmaTiku[j]*sigmaTiku[j];
   d +=  (1.0 - (double)m[j]/mTan)*(1.0-(double)m[j]/mTan)
                     *pow(sigmaTiku[j], 4)/(A[j]-1);
   }

   dof = (double) ((M2*M2)/d);
//  dof = ceil((M2*M2)/d);

   Tan = M1/M2;

   tan = imsls_d_F_cdf(Tan, GROUP-1, dof);
   return(tan);

}


void bubble(double *array, const int size)
{
    int pass, j;

    void swap(double *, double *);

    for(pass = 1 ; pass < size; pass++)
     for(j = 0; j < size -1; j++)
          if(array[j] > array[j+1])
            swap(&array[j], &array[j+1]);
    }


void swap(double *Ptr1, double *Ptr2)
```

```
{
   double temp;

   temp = *Ptr1;
   *Ptr1 = *Ptr2;
   *Ptr2 = temp;
}
```

APPENDIX O

C PROGRAM 7: BANDWIDTH SELECTION

```
/* ================================================================== */
/*    Simulation on the Bandwidth selection for NED family            */
/* ================================================================== */


#include <stdio.h>
#include <iostream.h>
#include <math.h>
#include <float.h>
#include <imsls.h>
#include <time.h>

#define N  5000
#define PI  M_PI
#define ERROR  0.00001

#define SIZE1   40
#define SIZE2  40

#define SIZE   ((SIZE1) + (SIZE2))
// #define BAND   0.5
#define  NEH 100
#define  SAM  3
#define BAND1 0.9
#define BAND2 0.9


FILE *fout;



/*********************************************************************/
/*                 Normal Random Number generation                   */
/*********************************************************************/


void randomNormal(double r1[],double r2[], double avg, int m1, int m2)
{

   int i;
   static int seed = 123475;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   for(i = 0; i < m1; i++)
    r1[i] = r1[i] + avg;
```

```
    seed = imsls_random_seed_get();

    imsls_random_seed_set(seed);
    imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

    seed=imsls_random_seed_get();

}



main()
{
    double alpha005 = 0.05, alpha001 = 0.01;
    double random_1[SIZE1]={0}, random_2[SIZE2]={0};
    double HD_level,   HD_rej005,   HD_rej001;

    void estimate(double [], double [], int, int);
    double hd(double [], double [], int, int);

    double diff[2]={0,1};
    int n1[SAM] = {10,10,30};
    int n2[SAM] = {10,40,30};

    double times, second;
    int minute, sampleNumber;
    int d, sam,  count=0;


    fout = fopen("band.txt","a");
    fprintf(fout,"==  Normal(0,1)\t band1 = %2.1f band2 = %2.1f ===\n\n",
                                BAND1, BAND2);

    for( sam = 0; sam < SAM; sam++){

      for( d = 0; d < 2; d++)
      {

        fprintf(fout,"-- n1 = %d, n2 = %d  d = %2.1f\n",n1[sam],n2[sam],
                                                          diff[d]);

     sampleNumber = 0;
     HD_rej005 = 0;   HD_rej001=0;

     while( sampleNumber >=0 && sampleNumber < N)
{
```

```
          randomNormal( random_1, random_2, diff[d], n1[sam], n2[sam]);

        estimate(random_1, random_2, n1[sam], n2[sam]);
        HD_level = hd(random_1, random_2, n1[sam], n2[sam]);

        if( (HD_level < 0))
         count++;
        else
        {
        if(HD_level < alpha001)
        HD_rej001 += 1;
        if(HD_level < alpha005)
        HD_rej005  += 1;


         sampleNumber++;
         printf("%d %d\n", sampleNumber,sam);

        }
          /*  else */
     }     /* while */

    fprintf(fout,"Test Statistc   \talpha = 0.05 \talpha = 0.01\n");
    fprintf(fout,"_____________________________________________________"
                                                          "___\n\n");

    fprintf(fout,"  HD            \t% f  \t% f\n", HD_rej005/N,
                                                    HD_rej001/N);
    fprintf(fout,"\n Number of failure = %d\n\n", count);

    }      /* for   */

} /* SAM */
  times = imsls_ctime();
  minute = times/60;
  second = times - 60*minute;
  fprintf(fout,"\n\nCPU TIME USED is %d min %4.2f seconds\n", minute,
                                                          second);
  return 0;
}



/*************************************************************************/
/* Caculation of Initial values and sample density estimate              */
/*************************************************************************/
```

```
double fstar1[SIZE1], fstar2[SIZE2];
double mu1hat, mu2hat, sigmahat, h1, h2;

void estimate(double r1[], double r2[], int m1, int m2)
{
   double *Median1, *Median2, *Sigma;
   double  sum, expo1[SIZE1]={0}, r12[SIZE], expo2[SIZE2]={0};

   int i,j,k,l, m;

/* Calling of initail Values for MU_1, Mu_2, and Sigma */

   Median1 = imsls_d_simple_statistics(m1,1,r1,IMSLS_MEDIAN_AND_SCALE,0);
   Median2 = imsls_d_simple_statistics(m2,1,r2,IMSLS_MEDIAN_AND_SCALE,0);
   mu1hat =  Median1[14];
   mu2hat =  Median2[14];
   h1 = BAND1 * Median1[15];
   h2 = BAND2 * Median2[15];

   for(i=0; i < m1; i++)
    r12[i] = fabs(r1[i]-mu1hat);

   for(j=0; j < m2;j++)
      r12[m1+j] = fabs(r2[j]-mu2hat);

   m = m1+m2;
   Sigma =  imsls_d_simple_statistics(m,1,r12,IMSLS_MEDIAN_AND_SCALE,0);
   sigmahat =  1.48*Sigma[14];



/* Calling of Sample Density Estiamte */


   for(l = 0; l < m1; l++)
   {
    sum = 0.0;
    for(j = 0; j < m1; j++)
      {
        expo1[j] = ( (r1[j]-r1[l])*(r1[j]-r1[l]) )/(2.0*h1*h1);
         sum = sum + exp(-expo1[j]);
      }
     fstar1[l] = (double) sum/(m1*(sqrt(2.0*PI)*h1));

   }
```

```
   for( k = 0; k < m2; k++)
   {
    sum = 0.0;
    for(l = 0; l < m2; l++)
      {
        expo2[l] = ( (r2[l]-r2[k])*(r2[l]-r2[k]) )/(2.0*h2*h2);
         sum = sum + exp(-expo2[l]);
      }
     fstar2[k] = (double) sum/(m2*(sqrt(2.0*PI)*h2));
   }
}
```

```
/*************************************************************************/
/*                  Calculation of Hellinger Disparity                   */
/*************************************************************************/
```

```
double hd(double r1[], double r2[], int m1, int m2)
{
   double mstar1[SIZE1], mstar2[SIZE2];
   double mu1HD, mu2HD, sigmaHD;
//   double raf1[SIZE1], raf2[SIZE2],delta1[SIZE1], delta2[SIZE2];
   double stopPoint;
   double bunmo,expon,weight1[SIZE1], weight2[SIZE2];
   double numerator, numer1, numer2, denominator1, denominator2;
   double HD; /* TEST STATISTICS */
   double hd_dist, mu1Hat, mu2Hat, sigmaHat;
   int i, j, k, l;
   int converge = 0;

   mu1Hat = mu1hat;
   mu2Hat = mu2hat;
   sigmaHat = sigmahat;


 do
 {

/*  First Model Smoothing here  */
```

```
bunmo = 2.0*(sigmaHat*sigmaHat + h1*h1);

   for(i = 0; i < m1; i++)
   {
      expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/bunmo;
      mstar1[i] = exp( -expon )/sqrt(PI*bunmo);
      if(fstar1[i] <= DBL_EPSILON)
   weight1[i] = 0.0;

      else
      {
    weight1[i] = 2.0 * sqrt(mstar1[i]/fstar1[i])
       - mstar1[i]/fstar1[i];
        weight1[i] = weight1[i] < 0.0 ? 0.0:weight1[i];
      }
   }



/*Second Model smoothing here  */


   bunmo = 2.0*(sigmaHat*sigmaHat + h2*h2);

   for(k = 0; k < m2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/bunmo;
      mstar2[k] = exp( -expon )/sqrt(PI*bunmo);
      if(fstar2[k] <= DBL_EPSILON)
    weight2[k] = 0.0;

      else
      {
         weight2[k] = 2.0 * sqrt(mstar2[k]/fstar2[k])
       - mstar2[k]/fstar2[k];
      weight2[k] = weight2[k] < 0.0 ? 0.0:weight2[k];
    }

   }


   /*********************************************************/
   /*              Estiamtors of Mu1 Mu2 Sigma              */
   /*********************************************************/
```

```
/*******************HERE IS MU 1 *******************/

    numerator = 0.0;
   denominator1 = 0.0;

    for(l = 0; l < m1; l++)
     {
        numerator = numerator + weight1[l]*r1[l];
         denominator1 = denominator1 + weight1[l];
      }

    mu1HD = numerator/denominator1;


/******************** HERE IS MU 2 **********************/


   numerator = 0.0;
   denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
        numerator = numerator + weight2[j]*r2[j];
      denominator2 = denominator2 + weight2[j];
     }

    mu2HD = numerator/denominator2;


/************ HERE IS POOLED SIGMA ***********************/


   numer1 = 0.0;
      numer2 = 0.0;

for(l = 0; l < m1; l++)
        numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
        for(k = 0; k < m2; k++)
       numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

    sigmaHD = sqrt((numer1 + numer2)/(denominator1 + denominator2));



   stopPoint = sqrt((mu1HD-mu1Hat)*(mu1HD-mu1Hat)+
                       (mu2HD-mu2Hat)*(mu2HD-mu2Hat)+
                        (sigmaHD-sigmaHat)*(sigmaHD-sigmaHat));
```

```
      mu1Hat = mu1HD;
     mu2Hat = mu2HD;
     sigmaHat = sigmaHD;

       converge++;
    if( converge == NEH )
    {
       printf("HD FAIL TO CONVERGE\n");
       return(-1);
    }
 }while(stopPoint > ERROR);

    HD = (mu1HD - mu2HD) / ( sigmaHD*sqrt( 1.0/m1 + 1.0/m2) );
//   hd_dist = 2.0*(1.0 - imsls_d_t_cdf(fabs(HD),m1+m2-2) );
    hd_dist = 1.0 - imsls_d_t_cdf(HD,m1+m2-2);

     return(hd_dist);

}
```

APPENDIX P

C PROGRAM 8: MSE CALCULATION

```
/* ================================================================  */
/*    Simulation on finding alpha of NED family
/* ================================================================ */


#include <stdio.h>
#include <iostream.h>
#include <math.h>
#include <float.h>
#include <imsls.h>
#include <time.h>

#define N     5000
#define PI  M_PI
#define ERROR  0.00001

#define SIZE1   50
#define SIZE2  50

#define SIZE   ((SIZE1) + (SIZE2))
#define BAND   0.5
#define NEH 100
#define SAM  3



FILE *fout;
static int seed = 123475;


/*********************************************************************/
/*                 Normal Random Number generation                   */
/*********************************************************************/


void randomNormal(double r1[],double r2[], double avg, int m1, int m2)
{

   int i;


   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   for(i = 0; i < m1; i++)
    r1[i] = r1[i] + avg;
```

```
   seed = imsls_random_seed_get();

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

   seed=imsls_random_seed_get();

}

/*************************************************************************/
/*             Normal Mixture with t                                     */
/*************************************************************************/

void random43(double r1[],double r2[], double avg, int m1, int m2)
{

   int i, j;
   double *trandom, *prob;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   trandom = imsls_d_random_student_t(m1,1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   prob = imsls_d_random_uniform(m1,0);

   for(i=0; i < m1;i++)
   {
    if( prob[i] > 0.9 )
       r1[i] = r1[i] + avg;
      else
       r1[i] = trandom[i] + avg;
   }


      /* Second Set */

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
```

```
   trandom=imsls_d_random_student_t(m2,1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   prob=imsls_d_random_uniform(m2,0);

   for(j=0; j < m2;j++)
   {
    if( prob[j] > 0.9 )
       r2[j] = r2[j];
      else
       r2[j] = trandom[j];
   }


seed = imsls_random_seed_get();
}

/*************************************************************************/
/*    Normal Mixture with Uniform                                        */
/*************************************************************************/

void random44(double r1[],double r2[], double avg, int m1, int m2)
{
   int i, j, k;
  static int seed = 123475;
   double *urandom, *prob;

   imsls_random_seed_set(seed);
   imsls_d_random_normal(m1,IMSLS_RETURN_USER,r1,0);

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   urandom=imsls_d_random_uniform(m1,0);

   for( k = 0; k < m1; k++){
    if( urandom[k] < DBL_EPSILON )
       urandom[k] = DBL_EPSILON;
      else
          ;
   }

   seed = imsls_random_seed_get();
   imsls_random_seed_set(seed);
   prob=imsls_d_random_uniform(m1,0);

   for(j=0; j < m1;j++)
```

```
    {
     if( prob[j] > 0.9 )
        r1[j] = r1[j]/urandom[j] + avg;
       else
        r1[j] = r1[j] + avg;
    }

    /* Second set */

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    imsls_d_random_normal(m2,IMSLS_RETURN_USER,r2,0);

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    urandom=imsls_d_random_uniform(m2,0);

    for( k = 0; k < m2; k++){
     if( urandom[k] < DBL_EPSILON )
        urandom[k] = DBL_EPSILON;
       else
           ;
    }

    seed = imsls_random_seed_get();
    imsls_random_seed_set(seed);
    prob=imsls_d_random_uniform(m2,0);

    for(i=0; i < m2;i++)
    {
     if( prob[i] > 0.9 )
        r2[i] = r2[i]/urandom[i];
       else
        r2[i] = r2[i];
    }

seed = imsls_random_seed_get();
}

main()
{
    double random_1[SIZE1]={0}, random_2[SIZE2]={0};
    int ned(double [], double [], double, double, int, int );
    void estimate(double [], double [], int, int);

    double a[9] = {0.15,0.3,0.5,0.7,0.9,1.0,1.5,2.0,2.5};
    double diff[2]={0,1};
```

```
   int n1[SAM] = {10,30,50};
   int n2[SAM] = {10,30,50};

   double times, second;
   int minute, sampleNumber;
   int d, sam, k,flag;


   fout = fopen("mseSigma.txt","w");


for( sam = 0; sam < SAM; sam++){

 for( d = 0; d < 2; d++)
  {

   fprintf(fout,"-- n1 = %d, n2 = %d  d = %2.1f\n",n1[sam],n2[sam],diff[d]);
   fprintf(fout,"         Mean1     Var1    MSE1  Mean2   Var2    MSE2"
   "   E(sig) Var(sig)  MSE(sig)\n");

  for(k=0; k < 9; k++){
      seed = 123475;
    sampleNumber = 0;


    while( sampleNumber >=0 && sampleNumber < N)
{

               /* Select your distribution */

        randomNormal( random_1, random_2, diff[d], n1[sam], n2[sam]);
//        random43( random_1, random_2, diff[d], n1[sam], n2[sam]);
//        random44( random_1, random_2, diff[d], n1[sam], n2[sam]);



       estimate(random_1, random_2, n1[sam], n2[sam]);
         flag = ned(random_1, random_2,a[k], diff[d], n1[sam],n2[sam]);

       if( flag == 0)
           ;
       else
           sampleNumber++;
//       printf("%d\n",sampleNumber);
}     /* while */
  }   /* a[k] */
 }     /*  diff   */
```

```
} /* SAM */
  times = imsls_ctime();
  minute = times/60;
  second = times - 60*minute;
  fprintf(fout,"\n\nCPU TIME USED is %d min %4.2f seconds\n", minute,
                                                          second);
  return 0;
}



/*************************************************************************/
/* Calculation of Initial values and sample density estimate             */
/*************************************************************************/


double fstar1[SIZE1], fstar2[SIZE2];
double mu1hat, mu2hat, sigmahat, h1, h2;

void estimate(double r1[], double r2[], int m1, int m2)
{
double *Median1, *Median2, *Sigma;
   double  sum, expo1[SIZE1]={0}, r12[SIZE], expo2[SIZE2]={0};
   int i,j,k,l, m;

/* initail Values for MU_1, Mu_2, and Sigma */

   Median1 = imsls_d_simple_statistics(m1,1,r1,IMSLS_MEDIAN_AND_SCALE,0);
   Median2 = imsls_d_simple_statistics(m2,1,r2,IMSLS_MEDIAN_AND_SCALE,0);
   mu1hat =  Median1[14];
   mu2hat =  Median2[14];
   h1 = BAND * Median1[15];
   h2 = BAND * Median2[15];

   for(i=0; i < m1; i++)
    r12[i] = fabs(r1[i]-mu1hat);

   for(j=0; j < m2;j++)
      r12[m1+j] = fabs(r2[j]-mu2hat);

   m = m1+m2;
   Sigma =  imsls_d_simple_statistics(m,1,r12,IMSLS_MEDIAN_AND_SCALE,0);
   sigmahat =  1.48*Sigma[14];



/* Calling of Sample Density Estiamte */
```

```
   for(l = 0; l < m1; l++)
   {
    sum = 0.0;
    for(j = 0; j < m1; j++)
      {
       expo1[j] = ( (r1[j]-r1[l])*(r1[j]-r1[l]) )/(2.0*h1*h1);
         sum = sum + exp(-expo1[j]);
      }
     fstar1[l] = (double) sum/(m1*(sqrt(2.0*PI)*h1));

   }


   for( k = 0; k < m2; k++)
   {
    sum = 0.0;
    for(l = 0; l < m2; l++)
      {
       expo2[l] = ( (r2[l]-r2[k])*(r2[l]-r2[k]) )/(2.0*h2*h2);
         sum = sum + exp(-expo2[l]);
      }
     fstar2[k] = (double) sum/(m2*(sqrt(2.0*PI)*h2));
   }
}



int ned(double r1[], double r2[], double b, double mu ,int m1, int m2)
{
   double mstar1[SIZE1], mstar2[SIZE2];
   double mu1NED, mu2NED, sigmaNED;
   double raf1[SIZE1], raf2[SIZE2], weight1[SIZE1], weight2[SIZE2];
   double stopPoint;
   double bunmo,expon, delta1[SIZE1], delta2[SIZE2];
   double numerator, numer1, numer2, denominator1, denominator2;
   double mu1Hat, mu2Hat, sigmaHat;
   static double sumNED1=0,ssumNED1=0,sumNED2=0,ssumNED2=0;
   static double sumSigma=0,ssumSigma=0;
   static double meanSigma, varSigma, MSESigma;
   static double mean1, mean2, var1, var2, MSE1, MSE2;
   static int iteration=0;
   int i,j,k,l;
   int converge = 0;

  mu1Hat = mu1hat;
```

```
   mu2Hat = mu2hat;
   sigmaHat = sigmahat;


/*************************************************************/
/*                  Caculation of Estimators                 */
/*************************************************************/



do
{

/*  First Model Smoothing here  */


bunmo = 2.0*(sigmaHat*sigmaHat + h1*h1);

   for(i = 0; i < m1; i++)
   {
      expon = ( (r1[i]-mu1Hat)*(r1[i]-mu1Hat) )/bunmo;
      mstar1[i] = exp( -expon )/sqrt(PI*bunmo);
      if(mstar1[i] <= DBL_EPSILON)
   weight1[i] = 0.0;
      else
      {
       delta1[i] = fstar1[i]/mstar1[i] - 1.0;
        raf1[i] = ( b+1.0 - (b+1.0+b*delta1[i])*exp( -(b*delta1[i]) ) )
                  /(b*b);
          weight1[i] = (raf1[i]+1)/(delta1[i]+1);
      }
        weight1[i] = weight1[i] < 0.0 ? 0.0:weight1[i];
   }



/*Second Model smoothing here  */


   bunmo = 2.0*(sigmaHat*sigmaHat + h2*h2);

   for(k = 0; k < m2; k++)
   {
      expon = ( (r2[k]-mu2Hat)*(r2[k]-mu2Hat) )/bunmo;
      mstar2[k] = exp( -expon )/sqrt(PI*bunmo);
      if(mstar2[k] <= DBL_EPSILON)
    weight2[k] = 0.0;
```

```
      else
      {
       delta2[k] = fstar2[k]/mstar2[k] - 1.0;
      raf2[k] =( b+1.0 - (b+1.0+b*delta2[k])*exp(-(b*delta2[k])))
                /(b*b);
        weight2[k] = (raf2[k]+1)/(delta2[k]+1);
       }
       weight2[k] = weight2[k] < 0.0 ? 0.0:weight2[k];
   }



/*  Estiamtors of Mu1 Mu2 Sigma  */

/*******************HERE IS MU 1 *******************/


    numerator = 0.0;
    denominator1 = 0.0;


    for(l = 0; l < m1; l++)
    {
     numerator = numerator + weight1[l]*r1[l];
       denominator1 = denominator1 + weight1[l];
    }

    mu1NED = numerator/denominator1;



/******************* HERE IS MU 2 ********************/


    numerator = 0.0;
    denominator2 = 0.0;

    for(j = 0; j < m2; j++)
    {
     numerator = numerator + weight2[j]*r2[j];
       denominator2 = denominator2 + weight2[j];
    }

    mu2NED = numerator/denominator2;


/************** HERE IS POOLED SIGMA ********************/
```

```
numer1 = 0.0;
    numer2 = 0.0;

       for(l = 0; l < m1; l++)
    numer1 = numer1 + weight1[l]*(r1[l]-mu1Hat)*(r1[l]-mu1Hat);
       for( k = 0; k< m2; k++)
       numer2 = numer2 + weight2[k]*(r2[k]-mu2Hat)*(r2[k]-mu2Hat);

    sigmaNED = sqrt((numer1 + numer2)/(denominator1 + denominator2));


   stopPoint = sqrt((mu1NED-mu1Hat)*(mu1NED-mu1Hat) +
                       (mu2NED-mu2Hat)*(mu2NED-mu2Hat)+
                                (sigmaNED-sigmaHat)*(sigmaNED-sigmaHat));

    mu1Hat = mu1NED;
     mu2Hat = mu2NED;
    sigmaHat = sigmaNED;

   converge++;
   if( converge == NEH )
   {
      printf("NED(%3.2f) FAIL TO CONVERGE\n", b);
      return(0);
   }

}while(stopPoint > ERROR);


   sumNED1 += mu1NED;
   ssumNED1 = ssumNED1 + mu1NED*mu1NED;
   sumNED2 += mu2NED;
   ssumNED2 = ssumNED2 + mu2NED*mu2NED;
   sumSigma += sigmaNED*sigmaNED;
   ssumSigma = ssumSigma + sigmaNED*sigmaNED*sigmaNED*sigmaNED;
   iteration++;

   if( iteration == N-1){
    mean1 = sumNED1 / ((double) N);
    mean2 = sumNED2 / ((double) N);
        meanSigma = sumSigma / ((double) N);
    var1 = (ssumNED1 - N*mean1*mean1)/ ((double) (N-1));
    var2 = (ssumNED2 - N*mean2*mean2)/ ((double) (N-1));
    MSE1 = var1 + (mean1-mu)*(mean1-mu);
    MSE2 = var2 + mean2*mean2;
```

```
      varSigma = (ssumSigma - N*meanSigma*meanSigma)/((double) (N-1));
      MSESigma = varSigma + (meanSigma - 1.0)*(meanSigma - 1.0);

    fprintf(fout,"NED%3.2f % 5.4f % 5.4f % 5.4f % 5.4f % 5.4f % 5.4f "
       "% 5.4f % 5.4f % 5.4f\n",b, mean1, var1, MSE1, mean2,var2, MSE2,
                                          meanSigma,varSigma,MSESigma );

    sumNED1=0;
    ssumNED1=0;
    sumNED2 =0;
    ssumNED2=0;
        sumSigma=0;
        ssumSigma=0;
        iteration = 0;
   }

    return(999);
}
```

VITA

Chansoo Kim

Candidate for the Degree of

Doctor of Philosophy

Thesis: ROBUST TESTS USING WEIGHTED LIKELIHOOD ESTIMATION

Major Field: Statistics

Biographical:

Personal Data: Born in Iri, Korea, On October 8, 1961, the son of ChongShik Kim and BoHee Lee.

Education: Graduate from NamSung High School, Iri, Korea in February, 1980; received Bachelor of Science Degree in Statistics from Chung-Ang University, Seoul, Korea in February, 1985; received Master of Science Degree in Statistics from Chung-Ang University, August, 1987; received Master of Science in Statistics from Southern Methodist University, Dallas, Texas in August, 1993; Completed the requirements for the Doctor of Philosophy degree with a Major in Statistics at Oklahoma State University in July, 1997.

Experience: Graduate research assistant,Chung-Ang University, Department of Statistics, 1985-1986; Graduate Teaching assistant, Oklahoma State University, Department of Statistics, 1995-1997.

Professional Membership: American Statistical Association, International Association for Statistical Computing, Korean Statistical Society, Korean-American Scientists and Engineers Association.

Honor Membership: *Mu Sigma Rho* ($\mu\,\sigma\,\rho$), the National Statistics Honor Fraternity; The Honor Society of *Phi Kappa Phi* ($\Phi\,\kappa\,\Phi$).