

# Testing Benford's Law with the First Two Significant Digits

By

STANLEY CHUN YU WONG

B.Sc. Simon Fraser University, 2003

A Thesis Submitted in Partial Fulfillment of the  
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

© STANLEY CHUN YU WONG, 2010

University of Victoria

*All rights reserved. This thesis may not be reproduced in whole or in part, by  
photocopying or other means, without the permission of the author.*

# **Supervisory Committee**

## **Testing Benford's Law with the First Two Significant Digits**

By

STANLEY CHUN YU WONG

B.Sc. Simon Fraser University, 2003

### **Supervisory Committee**

Dr. Mary Lesperance, (Department of Mathematics and Statistics)

**Supervisor**

Dr. William J. Reed, (Department of Mathematics and Statistics)

**Departmental Member**

## Supervisory Committee

Dr. Mary Lesperance, (Department of Mathematics and Statistics)

**Supervisor**

Dr. William J. Reed, (Department of Mathematics and Statistics)

**Departmental Member**

## Abstract

Benford's Law states that the first significant digit for most data is not uniformly distributed. Instead, it follows the distribution:  $P(d = d_1) = \log_{10}(1 + 1/d_1)$  for  $d_1 \in \{1, 2, \dots, 9\}$ . In 2006, my supervisor, Dr. Mary Lesperance et. al tested the goodness-of-fit of data to Benford's Law using the first significant digit. Here we extended the research to the first two significant digits by performing several statistical tests – LR-multinomial, LR-decreasing, LR-generalized Benford, LR-Rodriguez, Cramèr-von Mises  $W_d^2$ ,  $U_d^2$ , and  $A_d^2$  and Pearson's  $\chi^2$ ; and six simultaneous confidence intervals – Quesenberry, Goodman, Bailey Angular, Bailey Square, Fitzpatrick and Sison.

When testing compliance with Benford's Law, we found that the test statistics LR-generalized Benford,  $W_d^2$  and  $A_d^2$  performed well for Generalized Benford distribution, Uniform/Benford mixture distribution and Hill/Benford mixture distribution while Pearson's  $\chi^2$  and LR-multinomial statistics are more appropriate for the contaminated additive/multiplicative distribution. With respect to simultaneous confidence intervals, we recommend Goodman and Sison to detect deviation from Benford's Law.

# Table of Contents

Supervisory Committee .....	ii
Abstract.....	iii
Table of Contents .....	iv
List of Tables.....	v
List of Figures.....	viii
Acknowledgments.....	xi
1. Introduction .....	1
2. Benford's Law.....	4
2.1 Description and History .....	4
2.2 Research and Applications .....	6
2.2.1 Benford's Law and the screening of analytical data: the case of pollutant concentrations in ambient air .....	6
2.2.2 Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance ...	13
2.2.3 Price developments after a nominal shock: Benford's Law and psychological pricing after the euro introduction .....	19
2.2.4 Benford's Law and psychological barriers in certain eBay auctions.....	23
3. Test Statistics.....	26
3.1 Likelihood ratio tests for Benford's Law.....	26
3.2 Tests based on Cramér-von Mises statistics.....	28
3.3 Simultaneous confidence intervals for multinomial probabilities .....	33
4. Numerical Results.....	38
5. Conclusion.....	85
Bibliography .....	87
Appendix A .....	90

## List of Tables

Table 1.1: Nominal GDP (millions of USD/CAD) of top 20 countries.....	2
Table 2.1: Real and faked population data for 20 countries .....	4
Table 2.2: Details of the pollution data sets analyzed by Brown (2005).....	8
Table 2.3: Comparison of the ambient air pollution data sets in Table 2.2 with the expected initial digit frequency predicted by Benford's Law .....	9
Table 2.4: The effect on the initial digit frequency of Brown's digit manipulation of dataset B .	11
Table 2.5: The percentage change in $\Delta_{bl}$ , $\bar{x}$ , and $\sigma$ as a function of the percentage of modified data for dataset B .....	12
Table 2.6: Relative frequencies of initial digits of committee-to-committee in-kind contributions (first digits), 1994-2004 .....	15
Table 2.7: Relative frequencies of first digits for in-kind contributions by contribution size .....	18
Table 2.8: Leading digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after) .....	21
Table 2.9: Second digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after) .....	22
Table 2.10: Third digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after) .....	23
Table 3.1: Eigenvalues for Cramèr-von Mises statistics - $W_d^2$ .....	32
Table 3.2: Eigenvalues for Cramèr-von Mises statistics - $U_d^2$ .....	32
Table 3.3: Eigenvalues for Cramèr-von Mises statistics - $A_d^2$ .....	33
Table 3.4: Asymptotic percentage points for Cramer-von Mises statistics.....	33
Table 4.1: Multinomial distribution used in simulation and numerical study .....	38
Table 4.2: The relative frequencies of the 1st two digits of Benford's distribution .....	39
Table 4.3: The relative frequencies of the 1st two digits of "Hill" distribution .....	40

Table 4.4: Proportion of simulated data sets rejecting the null hypothesis of Benford's Law, N = 1000 replications .....	41
Table 4.5: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Uniform distribution, N = 1000 replications .....	41
Table 4.6: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from the contaminated additive Benford distribution for digit 10 with $\alpha = 0.02$ , N = 1000 replications .....	42
Table 4.7: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from the contaminated additive Benford distribution for digit 10 with $\alpha = 0.06$ , N = 1000 replications .....	42
Table 4.8: Proportion of simulated data set rejecting the null hypothesis when simulated data are from the contaminated multiplicative Benford distribution for digit 10 with $\alpha = 1.2$ , N =1000 replications .....	43
Table 4.9: Proportion of simulated data set rejecting the null hypothesis when simulated data are from the contaminated multiplicative Benford distribution for digit 10 with $\alpha = 1.5$ , N =1000 replications .....	43
Table 4.10: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Generalized Benford distribution with $\alpha = -0.1$ , N = 1000 replications .....	44
Table 4.11: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Generalized Benford distribution with $\alpha = 0.1$ , N = 1000 replications .....	44
Table 4.12: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Uniform/Benford Mixture distribution with $\alpha = 0.1$ , N = 1000 replications.....	45
Table 4.13: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Hill/Benford Mixture distribution with $\alpha = 0.1$ , N = 1000 replications .....	45
Table 4.14: Coverage proportions for 90%, 95% and 99% simultaneous confidence intervals for data generated using the Benford distribution.....	74
Table 4.15: Coverage proportions for 90%, 95% and 99% simultaneous confidence intervals for data generated using the Uniform distribution .....	76
Table 4.16: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.02$ ) with digits 10 to 14, n=1000 ..	76
Table 4.17: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.02$ ) with digits 10 to 14, n=2000 ..	77

Table 4.18: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.06$ ) with digits 10 to 14, $n=1000$ ..	77
Table 4.19: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.06$ ) with digits 10 to 14, $n=2000$ ..	78
Table 4.20: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.2$ ) with digits 10 to 14, $n=1000$ .....	78
Table 4.21: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.2$ ) with digits 10 to 14, $n=2000$ .....	79
Table 4.22: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.5$ ) with digits 10 to 14, $n=1000$ .....	79
Table 4.23: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.5$ ) with digits 10 to 14, $n=2000$ .....	80
Table 4.24: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = -0.5, -0.4, -0.3, -0.2, -0.1$ ), $n=1000$ .....	80
Table 4.25: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = -0.5, -0.4, -0.3, -0.2, -0.1$ ), $n=2000$ .....	81
Table 4.26: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=1000$ .....	81
Table 4.27: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=2000$ .....	82
Table 4.28: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Uniform/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=1000$ .....	82
Table 4.29: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Uniform/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=2000$ .....	83
Table 4.30: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Hill/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=1000$ .....	83
Table 4.31: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Hill/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ), $n=2000$ .....	84

## List of Figures

Figure 4.1: Simulated power for $n = 1000$ samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.02$ , $N = 1000$ replications, significance level 0.05. ....	47
Figure 4.2: Simulated power for $n = 2000$ samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.02$ , $N = 1000$ replications, significance level 0.05. ....	48
Figure 4.3: Simulated power for $n = 1000$ samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.06$ , $N = 1000$ replications, significance level 0.05. ....	49
Figure 4.4: Simulated power for $n = 2000$ samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.06$ , $N = 1000$ replications, significance level 0.05. ....	50
Figure 4.5: Simulated power for $n = 1000$ samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 1.2$ , $N = 1000$ replications, significance level 0.05. Note y-axis scale is not 0 to 1. ....	51
Figure 4.6: Simulated power for $n = 2000$ samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 1.2$ , $N = 1000$ replications, significance level 0.05. Note y-axis scale is not 0 to 1. ....	52
Figure 4.7: Simulated power for $n = 1000$ samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 1.5$ , $N = 1000$ replications, significance level 0.05. Note y-axis scale is not 0 to 1. ....	53
Figure 4.8: Simulated power for $n = 2000$ samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 1.5$ , $N = 1000$ replications, significance level 0.05. Note y-axis scale is not 0 to 1. ....	54



Figure 4.9: Simulated power for $n = 1000$ samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = -1.0, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1$ , $N = 1000$ replications, significance level 0.05. ....	55
Figure 4.10: Simulated power for $n = 2000$ samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = -1.0, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1$ , $N = 1000$ replications, significance level 0.05. ....	56
Figure 4.11: Simulated power for $n = 1000$ samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0$ , $N = 1000$ replications, significance level 0.05. ....	57
Figure 4.12: Simulated power for $n = 2000$ samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0$ , $N = 1000$ replications, significance level 0.05. ....	58
Figure 4.13: Simulated power for $n = 1000$ samples generated under Mixed Uniform/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ , $N = 1000$ replications, significance level 0.05. ....	59
Figure 4.14: Simulated power for $n = 2000$ samples generated under Mixed Uniform/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ , $N = 1000$ replications, significance level 0.05. ....	60
Figure 4.15: Simulated power for $n = 1000$ samples generated under Mixed Hill/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ , $N = 1000$ replications, significance level 0.05. ....	61
Figure 4.16: Simulated power for $n = 2000$ samples generated under Mixed Hill/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod, $W_d^2$ , $U_d^2$ , $A_d^2$ , and $\chi^2$ with $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ , $N = 1000$ replications, significance level 0.05. ....	62
Figure 4.17: Comparison of approximate and simulated power for the contaminated additive Benford distribution ( $\alpha = 0.02, 0.06$ ) with digits 10 to 18, $n = 1000$ (black solid line), 2000 (red dashed line) .....	64
Figure 4.18: Comparison of approximate and simulated power for the contaminated multiplicative Benford distribution ( $\alpha = 1.2, 1.5$ ) with digits 10 to 18, $n = 1000$ (black solid line), 2000 (red dashed line) , significance level 0.05. ....	65

Figure 4.19: Comparison approximate and simulated power for $n = 1000$ samples generated under Uniform/Benford mixture distribution for two CVM statistics, $W_d^2$ and $A_d^2$ , significance level 0.05. ....	66
Figure 4.20: Comparison approximate and simulated power for $n = 2000$ samples generated under Uniform/Benford mixture distribution for two CVM statistics, $W_d^2$ and $A_d^2$ , significance level 0.05. ....	67
Figure 4.21: Comparison approximate and simulated power for $n = 1000$ samples generated under Hill/Benford mixture distribution for two CVM statistics, $W_d^2$ and $A_d^2$ , significance level 0.05. ....	68
Figure 4.22: Comparison approximate and simulated power for $n = 2000$ samples generated under Hill/Benford mixture distribution for two CVM statistics, $W_d^2$ and $A_d^2$ , significance level 0.05. ....	69
Figure 4.23: Approximate power for $W_d^2$ for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05. ....	70
Figure 4.24: Approximate power for $U_d^2$ for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05. ....	71
Figure 4.25: Approximate power for $A_d^2$ for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05. ....	72
Figure 4.26: Approximate power for $\chi^2$ for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05. ....	73

## Acknowledgments

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Mary Lesperance, for her patience, encouragement, and guidance throughout all stages of my thesis. Her expertise and experience in the statistics area enabled me to advance my knowledge in the subject to a more profound and practical level. In addition, she has made my research a rewarding and invaluable part of my learning process. Without her continuous direction and support, this thesis would not have been possible.

Also, I am heartily grateful for the unconditional and endless support from my parents, Sik-Wah and Bonnie (Chun-Lai) Wong; my sister, Elaine (Yee-Ling) Wong; and my fiancé, Florence (Kit-Yee) Liu. They always stood by me at moments of frustration and disappointment when problems arose in the research project. Their kindness and understanding was the key driving force behind my achievement of this thesis.

Lastly, I would like to offer my sincere appreciation and regards for everyone who has contributed to the completion of this thesis and toward the success in my life.

# Chapter 1





















## 1. Introduction

Statistical methodologies have been widely used in accounting practice to enhance the accuracy of accounting work. After the occurrence of the accounting scandals of Enron and Worldcom several years ago [13, 40], there is an increasing interest in applying statistical techniques in accounting, especially auditing, to help identify fraud and errors in large volumes of accounting data. Along with tighter regulations and greater legal liability borne by the auditing profession, it is of significant interest for statisticians and accounting researchers to explore some statistical tools to assist with the analysis of accounting data.

One such tool is Benford's Law, which is also called the first significant digit phenomenon. Benford's Law was first discovered by Simon Newcomb in 1881 [29] and then examined by Frank Benford with actual datasets in 1938 [3]. Newcomb's concept was based on his observation of the logarithmic book from which he noticed that pages of smaller digits were more worn and thereby, he realized that smaller digits appear more often than larger digits as the first significant digit. On the other hand, Benford's research was built upon the empirical results of the application of Benford's Law to real-life data. He used the data to demonstrate the validity of the law without proving it using a mathematical approach. Note that neither of the above [29 or 3] provides a theoretical foundation to support Benford's Law. A quantitative proof of the law was not developed until the late 1990's when Theodore P. Hill [17] explained the law with statistical probabilities.

Hill's analysis involved two assumptions: scale-invariance and base-invariance. Scale-invariance implies that the measuring unit (scale) of a dataset is irrelevant. In other words, the distribution of numbers will not change due to a conversion of the units. For example, the distribution of GDP of the top twenty countries in Table 1.1 that is expressed in millions of USD will stay the same if the dataset is converted to millions of CAD.

Table 1.1: Nominal GDP (millions of USD/CAD) of top 20 countries

Exchange Rate: 1 USD = 1.04496 CAD			
Country		nominal GDP (millions of USD)	nominal GDP (millions of CAD)
United States		14,441,425	13,820,074
Japan		4,910,692	4,699,407
China		4,327,448	4,141,257
Germany		3,673,105	3,515,068
France		2,866,951	2,743,599
United Kingdom		2,680,000	2,564,691
Italy		2,313,893	2,214,336
Russia		1,676,586	1,604,450
Spain		1,601,964	1,533,039
Brazil		1,572,839	1,505,167
Canada		1,499,551	1,435,032
India		1,206,684	1,154,766
Mexico		1,088,128	1,041,311
Australia		1,013,461	969,856
Korea		929,124	889,148
Netherland		876,970	839,238
Turkey		729,983	698,575
Poland		527,866	505,154
Indonesia		511,765	489,746
Belgium		506,183	484,404

Theodore P. Hill stated the definition for base-invariance as follows: “A probability measure  $P$  on  $(\mathbb{R}^+, \mathcal{U})$  is base invariant if  $P(S) = P(S^{1/n})$  for all positive integers  $n$  and all  $S \in \mathcal{U}$ ”. This indicates that if a probability is base invariant, the measure of any given set of real numbers (in the mantissa  $\sigma$ -algebra  $\mathcal{U}$ ) should be the same for all bases and, in particular, for bases which are powers of the original base [17].

It is remarkable to note that Hill's work not only provided a theoretical basis for Benford's Law but also strengthened the "robustness" of the law by showing that while not all numbers conform to Benford's Law, when distributions are chosen randomly and then random samples are taken from each of those distributions, the combined set will have leading digits that exhibit patterns following Benford's distribution despite the fact that the randomly selected distributions may deviate from the law [17, 18, 19, 20 and 21].

In 2006, Lesperance, Reed, Stephens, Wilton, Cartwright tested the goodness-of-fit of data to Benford's Law using the first significant digit [25]. The purpose of this thesis is to extend the data examination to the first two significant digits. The three approaches for testing the goodness-of-fit are similar to those used by Lesperance et al. They are likelihood ratio test, Cramér-von Mises statistics test, six different simultaneous confidence intervals test: Quesenberry and Hurst [31]; Goodman [16]; Bailey angular transformation [2]; Bailey square root transformation [2]; Fitzpatrick and Scott [14]; Sison and Glaz [37], and univariate approximate binomial confidence interval test.

To give readers a general understanding of Benford's Law, we will start with its description, history, research, and application in Chapter 2. Chapter 3 will go on to perform the various procedures mentioned above to test the goodness-of-fit of the first two significant digits of the data. In Chapter 4, we will summarize the results of different methodologies. The last section, Chapter 5, will generate conclusions based on the analysis performed.





















## Chapter 2

### 2. Benford's Law

#### 2.1 Description and History

Let's start our discussion with a simple question. From Table 2.1, there are two columns of figures that correspond to the population of twenty countries. One of the columns contains real data while the other is made up of fake numbers. Which set of data do you think is fake?

Table 2.1: Real and faked population data for 20 countries.

Country		Real or Faked Population?!	
Afghanistan		19,340,000	28,150,000
Albania		4,370,000	3,170,000
Algeria		44,510,000	34,895,000
Andorra		81,000	86,000
Angola		37,248,000	18,498,000
Antigua and Barbuda		95,000	88,000
Argentina		48,254,389	40,134,425
Armenia		6,015,000	3,230,100
Australia		31,257,000	22,157,000
Austria		8,605,852	8,372,930
Bahamas		556,000	342,000
Bahrain		694,000	791,000
Bangladesh		201,689,000	162,221,000
Barbados		511,000	256,000
Belarus		7,538,000	9,489,000
Belgium		9,951,953	10,827,519
Belize		315,400	322,100
Botswana		1,810,000	1,950,000
Brazil		203,217,000	192,497,000
Brunei		510,030	400,000

Benford's Law illustrates the empirical observation that smaller digits occur more often than greater digits as the initial digits of a multi-digit number in many different types of large datasets. This concept is contrary to the common intuition that each of the digits from 1 to 9 has an equal probability of being the first digit in a number. Although this interesting phenomenon was named after Frank Benford, it was originally discovered by an astronomer and mathematician, Simon Newcomb.

In 1881, Newcomb reported his observation in the American Journal of Mathematics about the uneven occurrence of each of the digits from 1 to 9 as the initial digit in a multi-digit number because he noticed that the beginning pages of the logarithms book were more worn and must have been referred to more frequently. However, he did not investigate this phenomenon further. Benford extended the research on Newcomb's findings and published the results with testing support in 1938. In Benford's study, he found support for the statistical and mathematical merit of Newcomb's hypothesis by analyzing more than 20,000 values from dissimilar datasets including the areas of rivers, population figures, addresses, American League baseball statistics, atomic weights of elements, and numbers appearing in Reader's Digest articles. His results suggested that 1 has a probability of 30.6% as being the first digit in a multi-digit number, 18.5% for the digit 2, and just 4.7% for the digit 9. His testing demonstrated the (approximate) conformity of large datasets to the law that was named after him. His contributions included setting out a formal description and analysis of what is now known as the Benford's Law (which is also called the law of leading digit frequencies, law of anomalous numbers, significant digit law, or the first digit phenomenon). Benford's Law for the first one, two and three digits is expressed as a logarithm distribution:

$$P(d = d_1) = \log_{10}(1 + 1/d_1) \text{ for } d \in \{1, 2, \dots, 9\}$$

$$P(d = d_1 d_2) = \log_{10}(1 + 1/[10 * d_1 + d_2]) \text{ for } d \in \{10, 11, \dots, 99\}$$

$$P(d = d_1 d_2 d_3) = \log_{10}(1 + 1/[100 * d_1 + 10d_2 + d_3]) \text{ for } d \in \{100, 101, \dots, 999\}$$



Since Benford's release of his publication, there were other studies which confirmed the applicability of the law using accounting figures [10], eBay bids [15], Fibonacci series [7, 11], physical data [36], stock market prices [26], and survey data [23]. Benford's Law is now recognized for its significance and rigor in the academic field and its utility in practical applications.

Yet, it is important to note that Benford's Law can at best be held as an approximation because a scale invariant distribution has density proportional to  $1/x$  on  $R^+$  and no such proper distribution exists.

After the brief introduction of Benford's Law above, the answer to the question about Table 2.1 becomes apparent. The first digit of numbers in the first column occurs almost evenly among digits 1 to 9. On the other hand, those in the second column exhibit a pattern that closely conforms to Benford's Law where the digit 1 has the most occurrences with each greater digit having successively lower chance of being the first digit of a number. With the essence of Benford's Law in mind, the following sub-section presents a few notable examples of the use of Benford's Law.

## 2.2 Research and Applications

Benford's Law was applied in many types of research. Some applications demonstrated close resemblance of the data with Benford's Law while others tended to deviate from the law. Selective illustrations of the use of Benford's Law in diverse areas of interest are provided below.

### 2.2.1 Benford's Law and the screening of analytical data: the case of pollutant concentrations in ambient air

The first application to be introduced here is the research by Richard J. C. Brown on the use of Benford's Law to screen data related to pollutant concentrations in ambient air [5]. Air quality is often monitored by government agencies to ensure the amount of pollutants does not exceed an acceptable level as hazardous substances can harm public and environmental safety. The process of gathering data on pollutant concentrations in ambient air requires many steps including data collection on data-loggers, electronic transmission of collected data, translation and formatting of electronic data, and data-entry and manipulation on computer software programs.

Since the collected data have to go through a series of phases before they are ready for analysis, it is not unreasonable to expect that some types of errors are included in the dataset. Furthermore, data on air quality measurement often have a very high volume, which also increases the likelihood of bringing errors into the dataset. In cases where the errors result from the manipulation, omission, or transposition of the initial digit, Benford's Law is a possible way to detect them.

To expand this idea, Brown's studies attempted to evaluate the possibility of applying Benford's Law as a detection tool to identify data mishandling and to examine how small changes made to the dataset can lead to deviations from the law, which in turn, indicate the introduction of errors into the data. Brown selected a number of pollution datasets collected in the UK for his experiment. The datasets are described in Table 2.2.

Table 2.2: Details of the pollution data sets analyzed by Brown (2005)

Assigned Code	Description	Number of Observations
A	The annual UK average concentrations of the 12 measured heavy metals at all 17 monitoring sites between 1980 and 2004	1,174
B	The weekly concentrations of 12 measured heavy metals at all 17 monitoring sites across the UK during October 2004	821
C	The quarterly average concentrations of benzo[a]pyrene (a PAH) at all 25 monitoring sites during 2004	570
D	Hourly measurements of benzene at the Marylebone Road site during 2004	6,590
E	Hourly measurements of particulate matter (PM <sub>10</sub> size fraction) at the Marylebone Road site during 2004	8,593
F	Hourly measurements of particulate matter (PM <sub>10</sub> size fraction) at the Marylebone Road site during May 2004	689
G	Weekly measurements of lead at the Sheffield site during 2004	51
H	Hourly measurements of carbon monoxide at the Cardiff site during 2004	8,430

The outcome of the experiment showed that datasets A and B closely follow the distribution suggested by Benford's Law while the other datasets do not exhibit patterns consistent with the law. To quantify the degree to which each dataset deviates from (or agrees with) the law, the sum of normalized deviations,  $\Delta_{bl}$  was calculated for each dataset based on this formula:

$$\Delta_{bl} = \sum_{d_1=1}^{d_1=9} \left| \frac{P(d_1) - P_{obs}(d_1)}{P(d_1)} \right|$$

where  $P_{obs}(d_1)$  is the normalized observed frequency of initial digit  $d_1$  in the experimental dataset. A value of zero for  $\Delta_{bl}$  means that the dataset matches Benford's Law completely.

To assess if the numerical range of the data ( $R$ ) has an effect on the conformity of the dataset to Benford's Law,  $R$  is computed as:

$$R = \log_{10}(x_{max}/x_{min})$$

where  $x_{max}$  and  $x_{min}$  represent the maximum and minimum numbers, respectively, in the dataset. The result and analysis of Brown's experiment are reproduced in Table 2.3.

Table 2.3: Comparison of the ambient air pollution data sets in Table 2.2 with the expected initial digit frequency predicted by Benford's Law

Dataset	Benford's Law	A	B	C	D	E	F	G	H
Number of obs.	-	1,174	821	570	6,590	8,593	689	51	8,430
$R$	-	6.5	6.2	4.0	2.7	1.9	1.4	1.3	1.5
Relative frequency of initial digit:									
1	<b>0.301</b>	0.304	0.343	0.286	0.286	0.089	0.091	0.157	0.134
2	<b>0.176</b>	0.162	0.166	0.211	0.195	0.184	0.247	0.314	0.419
3	<b>0.125</b>	0.115	0.106	0.156	0.174	0.211	0.186	0.255	0.244
4	<b>0.074</b>	0.106	0.085	0.084	0.082	0.187	0.152	0.078	0.000
5	<b>0.079</b>	0.089	0.091	0.074	0.055	0.130	0.147	0.157	0.109
6	<b>0.067</b>	0.063	0.064	0.063	0.084	0.106	0.109	0.000	0.049
7	<b>0.058</b>	0.056	0.058	0.053	0.031	0.057	0.051	0.000	0.023
8	<b>0.051</b>	0.053	0.042	0.039	0.026	0.023	0.012	0.020	0.014
9	<b>0.046</b>	0.052	0.046	0.035	0.067	0.013	0.006	0.020	0.009
$\Delta_{bl}$	<b>0.00</b>	<b>0.64</b>	<b>0.85</b>	<b>1.32</b>	<b>2.69</b>	<b>4.86</b>	<b>5.40</b>	<b>6.66</b>	<b>6.67</b>

As indicated from the chart above, conformity of the data to Benford's Law as measured by the size of  $\Delta_{bl}$  roughly increases as the numerical range ( $R$ ) of the data increases.

Note:  $R$  is not related to the sample size.

Factors that can reduce the numerical range of a dataset include fewer types of pollutants or number of monitoring sites and shorter time span, if seasonal fluctuations are significant.

Having discussed the types of datasets that tend to fit or not fit into the distributions underlying Benford's Law, Brown re-analyzed dataset B but modified the dataset by removing the initial digit from part of observations so that the second digit becomes the first digit (except where the second digit is zero, then the third digit will become the first

digit). For example, datum 248 becomes 48 and datum 307 becomes 7. The purpose of this adjustment was to evaluate the potential effect of errors during data processing and manipulation on the datasets that follow Benford's Law. The modification of the data was made to 0.2% of the dataset to begin with and then the percentage was gradually increased to a maximum of 50%. The results and analysis of the modified data are reproduced in Table 2.4. The expected relative frequency for second digit column is only an approximation distribution because Brown removed the probability of zero occurring as a second digit and adjusted the second digit distribution by allocating the probability of zero, on a pro-rata basis, to the original distribution for digits 1 to 9. The modified probabilities of one to nine,  $Pr'(i)$ , were computed as follows:

$$Pr'(i) = Pr(i) + \left[ Pr(i) / \sum_{j=1}^9 Pr(j) \right] * Pr(0)$$

where  $i = 1, \dots, 9$  and  $Pr(i) = \sum_{j=1}^9 \log_{10}[1 + 1/(10 * j + i)]$ .

Since Brown only included three decimal points in the expected frequency of second digit column and due to the rounding errors, the column does not add up to 1.

Table 2.4: The effect on the initial digit frequency of Brown's digit manipulation of dataset B

	Percentage of Modified Data (%)									
	0	0.2	0.4	1	2	4	10	25	50	
Initial Digit After Modification	Relative frequency of initial digit after modification									Expected Relative Frequency of Second digit [ $Pr'(i)$ ]
1	0.343	0.343	0.343	0.339	0.339	0.329	0.328	0.289	0.235	0.129
2	0.166	0.166	0.164	0.164	0.164	0.158	0.152	0.150	0.137	0.124
3	0.106	0.106	0.108	0.106	0.106	0.106	0.108	0.116	0.112	0.119
4	0.085	0.083	0.083	0.083	0.085	0.083	0.083	0.092	0.094	0.114
5	0.091	0.092	0.092	0.092	0.092	0.092	0.089	0.094	0.102	0.110
6	0.064	0.064	0.064	0.064	0.067	0.069	0.071	0.073	0.079	0.106
7	0.058	0.058	0.058	0.060	0.062	0.064	0.064	0.062	0.077	0.103
8	0.042	0.042	0.042	0.044	0.042	0.048	0.052	0.064	0.094	0.099
9	0.046	0.046	0.046	0.048	0.042	0.050	0.054	0.060	0.069	0.097
$\Delta_{bl}$	0.85	0.90	0.89	0.93	0.96	0.95	0.98	1.20	2.73	4.89

Table 2.5 below demonstrates the sensitivity of Benford's Law to even small percentages of data manipulation. This is an important advantage that distinguishes it from common data screening techniques such as arithmetic mean and standard deviation. It can be seen from the computation below that the arithmetic mean and standard deviation of dataset B (after the same modification is made) are quite insensitive to data mishandling until the error percentage approaches 25%.

Table 2.5: The percentage change in  $\Delta_{bl}$ ,  $\bar{x}$ , and  $\sigma$  as a function of the percentage of modified data for dataset B

Percentage of Modified Data (%)	% Change, Resulting from Data Modification		
	% change in $\Delta_{bl}$	% change in $\bar{x}$	% change in $\sigma$
0.2	5.5	0.10	0.005
0.4	5.0	0.56	0.037
1	8.6	1.9	0.30
2	12	3.2	0.74
4	11	4.9	1.0
10	15	8.5	2.7
25	41	15	4.8
50	220	34	10

Brown's research on Benford's Law and the screening of pollutant data revealed that some datasets conformed closely to Benford's Law while others varied. The results can be rephrased in the major conclusions below:

1) The fit of the data depended on the number of orders of magnitude of the data range computed as  $R = \log_{10}(x_{max}/x_{min})$ . Datasets with greater numerical ranges, in particular, four orders of magnitude or above, are more likely to follow Benford's Law.

2) In addition, datasets having a larger size and covering a longer time period will show higher consistency with Benford's Law. Large sets tended to be more representative of the population and a long time span might include temporal and seasonal fluctuations in the data.

3) Furthermore, the data range increased with the number of monitoring sites and species included in a data set.

4) Because of the strong sensitivity of Benford's Law to even small percentages of errors, it is potentially a more effective tool than arithmetic mean and standard deviation in detecting data mishandling because the latter techniques may not signal a red flag until

errors approach 25%. In conclusion, Brown recommended the use of Benford's Law to screen pollutant data where the data range had four orders of magnitude or above.

### **2.2.2 Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance**

Other than the scientific application of Benford's Law, the law was also utilized in political science. One of its uses was in campaign finance. Cho and Gaines attempted to test for any irregularities in data related to in-kind political contributions [8]. They began their introduction with a list of the types of datasets to which Benford's Law may apply [12]:

1. Values that are the result of a mathematical computation (e.g. total units × unit cost)
2. Transaction-level data (e.g. sales)
3. Large datasets
4. Datasets where the mean is greater than the median and the skew is positive.

On the other hand, the characteristics to contraindicate its use were also identified:

1. Assigned numbers (e.g. cheque numbers, invoice numbers)
2. Values that are subject to human bias or inclination (e.g. prices)
3. Numbers that are completely or partially specified in a pre-determined way (e.g. account numbers, product codes)
4. Datasets bound by a minimum or maximum
5. Where no transaction was recorded (e.g. kickbacks).

Campaign finance regulations have a long history in the U.S. political system and have undergone many changes to improve the government's oversight on political



contributions and to prevent candidates from taking advantage of loopholes in the system. The regulations place various rules and limits on the type and amount of the political contributions, whether in the form of cash or in-kind contributions i.e. goods and services and whether received directly by the candidates (commonly described as “hard money”) or indirectly via a mechanism known as the joint fundraising committees (JFC) (often referred to as “soft money”).

Although data on cash contributions are readily available for analysis from Federal Election Commission (FEC) filings, some numbers are likely to occur more often than others due to an “artificial” rule – a maximum amount of \$2,000 set by the government. Historically, cash contributions were shown to skew toward the maximum amount. Therefore, cash contributions data are not suitable for further studies using Benford’s Law.

On the contrary, although in-kind contributions are also subject to the same maximum limit, they are less likely to fall within certain ranges because of the retail prices and wages or working hours that are pre-determined in most cases. This makes it harder to manipulate the dollar value of the goods or services paid by the supporters for the candidate. Hence, Cho and Gaines tested the data on in-kind contributions with Benford’s Law.

The data were from in-kind contributions made for the last six federal election cycles from 1994 to 2004 in the United States. Table 2.6 summarizes the first digit frequencies of the in-kind contributions data with comparison to Benford’s Law and Benford’s data:

Table 2.6: Relative frequencies of initial digits of committee-to-committee in-kind contributions (first digits), 1994-2004

Dataset	Benford's Law	Benford's data	1994	1996	1998	2000	2002	2004
1	<b>0.301</b>	<b>0.289</b>	0.329	0.244	0.274	0.264	0.249	0.233
2	<b>0.176</b>	<b>0.195</b>	0.187	0.217	0.185	0.211	0.226	0.211
3	<b>0.125</b>	<b>0.127</b>	0.136	0.158	0.153	0.111	0.107	0.085
4	<b>0.097</b>	<b>0.091</b>	0.079	0.096	0.103	0.107	0.116	0.117
5	<b>0.079</b>	<b>0.075</b>	0.089	0.102	0.118	0.101	0.105	0.095
6	<b>0.067</b>	<b>0.064</b>	0.083	0.063	0.059	0.043	0.043	0.042
7	<b>0.058</b>	<b>0.054</b>	0.041	0.048	0.037	0.064	0.034	0.037
8	<b>0.051</b>	<b>0.055</b>	0.024	0.032	0.039	0.024	0.030	0.040
9	<b>0.046</b>	<b>0.051</b>	0.032	0.040	0.033	0.075	0.090	0.141
<i>N</i>		<b>20,229</b>	9,632	11,108	9,694	10,771	10,348	8,396
$\chi^2$		<b>85.1</b>	349	507	431	4,823	1,111	2,181
$V_N^*$		<b>2.9</b>	5.7	10.1	8.1	5.5	7.8	8.7
$d^*$		<b>0.024</b>	0.052	0.081	0.061	0.071	0.097	0.131

Note: Benford's data refers to the 20,229 observations Benford collected

A quick look at Table 2.6 suggests that the adherence to Benford's Law worsened over time. In particular, the three latest elections exhibited conflicting initial digit distributions with increasingly more 9's as the first digit while the frequencies for 1's fell from election to election. To quantify the discrepancies between the actual and expected (Benford's) frequencies, three statistics were calculated for comparison: 1) Pearson goodness-of-fit test statistic  $\chi^2$ , 2) modified Kolmogorov-Smirnov test statistic  $V_N^*$  [24], and 3) Euclidean distance from Benford's Law  $d^*$ .

### Goodness-of-Fit Test Statistic $\chi^2$

The null hypothesis made in the goodness-of-fit test is that the data will follow the Benford's Law. The test statistic, having the  $\chi^2$  distribution with 8 degrees of freedom under the null hypothesis, is defined as

$$\chi^2 = \sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i}$$

where  $O_i$  and  $E_i$  are the observed and expected frequencies for digit  $i$ , respectively. If  $\chi^2 > \chi^2_{\alpha,8}$ , where  $\alpha$  is the level of significance, the null hypothesis will be rejected. That is, the in-kind contribution data is assumed not to conform to Benford's Law. Referring to the table above, the  $\chi^2$  statistics for all elections are large enough to reject the null hypothesis. However, Cho and Gaines noted a drawback of the goodness-of-fit test, which is the sensitivity of the test statistic to the sample size. Since Benford's data which were used to demonstrate Benford's Law rejects the null hypothesis, this chi-square test may be too strict to be a goodness-of-fit test tool. Therefore, another test statistic is computed to provide a different assessment of the deviation from Benford's Law.

### Modified Kolmogorov-Smirnov Test Statistic $V^*$

The modified Kolmogorov-Smirnov test statistic is defined as

$$V_N = D_N^+ + D_N^-,$$

where  $D_N^+ = \sup_{-\infty < x < \infty} [F_N(x) - F_0(x)]$  and  $D_N^- = \sup_{-\infty < x < \infty} [F_0(x) - F_N(x)]$

Giles [15] and Stephens [39] preferred the use of the modified  $V_N$ , that is,

$$V_N^* = V_N [N^{1/2} + 0.155 + 0.24N^{-1/2}]$$

because the revised form is independent of sample size with a critical value of 2.001 for  $\alpha = 0.01$ . Similar to the  $\chi^2$  statistics, the  $V_N^*$  statistics for all the elections rejected the null hypothesis. The  $V_N^*$  statistic for Benford's data also rejected the hypothesis.

## Euclidean Distance

An alternative framework introduced by Cho and Gaines is the Euclidean distance formulae, which is different from the hypothesis-testing model. The Euclidean distance from Benford's distribution is independent of sample size and defined below as the nine-dimensional space occupied by any first-digit vector:

$$d = \sqrt{\sum_{i=1}^9 (p_i - \pi_i)^2}$$

where  $p_i$  and  $\pi_i$  are the proportions of observations with  $i$  as the initial digit and expected by Benford's Law, respectively. Then  $d$  is divided by the maximum possible distance ( $\cong 1.0363$ ) which is computed by letting  $p_1, p_2, \dots, p_8 = 0$  and 9 is the only first digit observed ( $p_9 = 1$ ) to obtain a score between 0 and 1, which is labelled as  $d^*$  in the table. Although it is difficult to determine a reference point for gauging the closeness of the data to Benford's distribution, it is worthwhile to note that the more recent elections had relatively higher  $d^*$  scores than did the earlier elections. This observation shows that in-kind contribution data for the later elections tended to deviate from Benford's Law. It is also consistent with the relative frequencies summarized in the above table where 9's occurred more and 1's appeared less than expected as the leading digit.

To investigate further, Cho and Gaines tested the data again in four subsets that were defined by dollar values i.e. \$1 - \$9, \$10 - \$99, \$100 - \$999 and \$1000+. This time subsets with smaller amounts corresponded with the law more poorly as expected. However, year 2000 data exhibited close conformity among other subsets of small amounts unexpectedly because of the high volume of \$1 transactions. On the other hand, the three most recent elections demonstrated poor fit due to a large number of

\$90 - \$99 transactions. In addition, it is interesting to see that two- and three-digit numbers conformed to Benford's Law better than did other subsets. The results of the subset analysis are reproduced in Table 2.7.

Table 2.7: Relative frequencies of first digits for in-kind contributions by contribution size

	1	2	3	4	5	6	7	8	9	N	d*
<b>1994 – 2004</b>											
<i>Benford's Theoretical Frequencies in Various Digital Orders (p.569 Table V in "The Law of Anomalous Numbers")</i>											
1 <sup>st</sup> Order (\$1 - \$9)	0.393	0.258	0.133	0.082	0.053	0.036	0.024	0.015	0.007		0.140
2 <sup>nd</sup> Order (\$10 - \$99)	0.318	0.179	0.124	0.095	0.076	0.064	0.054	0.047	0.042		0.018
3 <sup>rd</sup> Order (\$100 - \$999)	0.303	0.176	0.125	0.097	0.079	0.067	0.058	0.051	0.045		0.002
Limiting Order (\$1000+)	0.301	0.176	0.125	0.097	0.079	0.067	0.058	0.051	0.046		0.000
<b>1994</b>											
\$1 - \$9	0.090	0.067	0.073	0.060	0.062	0.502	0.054	0.034	0.058	536	0.535
\$10 - \$99	0.349	0.206	0.126	0.083	0.083	0.047	0.051	0.027	0.027	3,493	0.051
\$100 - \$999	0.305	0.187	0.153	0.077	0.104	0.075	0.038	0.023	0.038	4,902	0.055
\$1000+	0.579	0.190	0.108	0.081	0.027	0.000	0.001	0.011	0.001	701	0.294
<b>1996</b>											
\$1 - \$9	0.057	0.116	0.210	0.099	0.080	0.080	0.080	0.077	0.202	352	0.389
\$10 - \$99	0.159	0.218	0.154	0.096	0.109	0.088	0.085	0.048	0.043	3,875	0.166
\$100 - \$999	0.259	0.226	0.172	0.090	0.108	0.056	0.028	0.024	0.036	5,925	0.093
\$1000+	0.558	0.191	0.073	0.127	0.044	0.002	0.005	0.000	0.000	956	0.278
<b>1998</b>											
\$1 - \$9	0.101	0.084	0.054	0.027	0.104	0.191	0.054	0.289	0.097	298	0.437
\$10 - \$99	0.188	0.144	0.192	0.105	0.110	0.100	0.060	0.046	0.054	3,305	0.153
\$100 - \$999	0.282	0.192	0.158	0.113	0.141	0.037	0.029	0.027	0.022	5,017	0.090
\$1000+	0.548	0.306	0.039	0.065	0.039	0.001	0.001	0.000	0.000	1,074	0.305
<b>2000</b>											
\$1 - \$9	0.427	0.036	0.056	0.021	0.053	0.167	0.062	0.058	0.120	468	0.274
\$10 - \$99	0.184	0.213	0.101	0.077	0.105	0.045	0.101	0.031	0.144	4,297	0.176
\$100 - \$999	0.249	0.203	0.142	0.154	0.117	0.040	0.047	0.021	0.027	4,855	0.100
\$1000+	0.560	0.308	0.045	0.050	0.036	0.000	0.001	0.000	0.000	1,151	0.316
<b>2002</b>											
\$1 - \$9	0.034	0.073	0.069	0.019	0.203	0.165	0.119	0.111	0.207	261	0.466
\$10 - \$99	0.195	0.206	0.124	0.078	0.097	0.051	0.038	0.030	0.181	4,356	0.183
\$100 - \$999	0.250	0.234	0.107	0.172	0.118	0.038	0.032	0.031	0.018	4,760	0.123
\$1000+	0.543	0.316	0.040	0.041	0.057	0.000	0.000	0.001	0.002	971	0.307
<b>2004</b>											
\$1 - \$9	0.035	0.031	0.040	0.035	0.256	0.172	0.154	0.181	0.097	227	0.495
\$10 - \$99	0.165	0.155	0.089	0.071	0.055	0.052	0.041	0.055	0.316	3,345	0.305
\$100 - \$999	0.238	0.231	0.095	0.180	0.129	0.035	0.037	0.027	0.028	3,836	0.136
\$1000+	0.490	0.359	0.040	0.043	0.064	0.002	0.000	0.002	0.000	988	0.292

Note: contribution amounts in whole dollars

The analysis above on the data on in-kind contributions and the subsets of these data merely showed the divergence from Benford's Law but did not explain the reason(s) for the deviations. Cho and Gaines pointed out that the merit of Benford's Law is its use as a screening tool for large volumes of data. Where actual results differ from expected distributions, it does not indicate fraud, but rather, it signals potential areas for further investigation. In conclusion, Cho and Gaines suggested the application of Benford's Law to help identify potential problems so that extra effort can be directed to uncover possible errors, loopholes, or illegality in campaign finance and other fields.

### **2.2.3 Price developments after a nominal shock: Benford's Law and psychological pricing after the euro introduction**

To demonstrate the wide applicability of Benford's Law to diverse areas, the following is an example related to business research. Sehity, Hoelzl, and Kirchler examined price developments in the European Union region after the introduction of euro dollars on January 1, 2002 to replace the national currencies of the participating countries [35]. Their paper also attempted to assess the existence of psychological pricing before and after the euro introduction.

Psychological pricing is a concept in the marketing field that describes the tendency to include certain nominal values in setting prices, such as the common use of "9" as the ending digit. It is also referred to as "just-below-pricing" or "odd-pricing" because of the practice to set a price marginally below a round number with the intention of making it appear considerably lower than the round number price. There are two forms of psychological pricing. The first form is to use "9" as ending digit while the other approach involves setting all digits but the first to be "9." A study performed by Schindler and Kirby on 1,415 newspaper price advertisements revealed that 27% of the prices ended in 0; 19% in 5; and 31% in 9 [34]. Another research by Stiving and Winer on

27,000 two-digit dollar prices of tuna and yogurt showed that from 36% to 50% of the prices had “9” as the last digit [38]. Furthermore, Brambach found similar patterns in a German price report where approximately 13% of the German mark prices ended with “0” and “5” each while 45% ended with “9” [4]. The results of these analyses suggested a W-like distribution of digits with digits 0, 5, and 9 occurring more often than others as the rightmost digit.

If prices are driven by market factors, it is reasonable to expect the distribution of the price digits to follow Benford’s Law. Nonconformity to the law can suggest that forces other than market vectors are in place to influence the determination of prices. The focus of Sehity, Hoelzl, and Kirchler’s paper is on evaluating the existence of and tendency toward psychological pricing after the euro introduction. This conversion of monetary measure from each EU member’s currency to a single currency is considered a nominal shock to the economy because a change in units should not affect the real value of the goods. In their studies, about 15 – 20 typical consumer goods were chosen from each of (a) bakery products, (b) drinks, and (c) cosmetics and then prices of the selected goods were gathered from supermarkets in 10 European countries: Austria, Belgium, Germany, Finland, France, Greece, Ireland, Italy, Portugal, and Spain. Data were collected at three different points in time: (a) before the euro introduction (from November to December 2001), (b) half a year after the introduction (from July to September 2002), and (c) one year after the conversion (from November to December 2002). For easier reference to the three points in time, the authors described them as wave 1, wave 2, and wave 3, respectively. Tables 2.8 – 2.10 also include the relative frequencies according to Benford’s Law for comparing the results of wave 1, wave 2 and wave 3. The relative frequencies under Benford column in Tables 2.9 and 2.10 are the marginal probabilities computed as follows:

$$Pr(2^{nd} \text{ digit} = d) = \sum_{i=1}^9 \log_{10}(1 + 1/(10i + d))$$

$$Pr(3^{rd} \text{ digit} = d) = \sum_{i=1}^9 \sum_{j=0}^9 \log_{10}(1 + 1/(100i + 10j + d))$$

Table 2.8: Leading digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after)

Digit	Wave 1 National currency		Wave 1 Euro		Wave 2 Euro		Wave 3 Euro		Benford
	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	
1	231	0.308	252	0.336+	214	0.326	209	0.312	0.30103
2	134	0.179	130	0.174	123	0.188	127	0.190	0.17609
3	91	0.121	79	0.105	65	0.099	67	0.100	0.12494
4	71	0.095	53	0.071-	55	0.084	52	0.078	0.09691
5	56	0.075	58	0.077	50	0.076	51	0.076	0.07918
6	40	0.053	51	0.068	39	0.059	40	0.060	0.06695
7	35	0.047	57	0.076+	47	0.072	54	0.081+	0.05799
8	36	0.048	27	0.036	24	0.037	25	0.039	0.05115
9	55	0.073+	42	0.056	39	0.059	43	0.064+	0.04576
	$\chi^2(8, n = 749) = 16.82, p = 0.032$		$\chi^2(8, n = 749) = 20.07, p = 0.010$		$\chi^2(8, n = 656) = 14.67, p = 0.066$		$\chi^2(8, n = 669) = 20.37, p = 0.009$		

Frequencies ( $f$ ), observed proportions ( $f_{rel}$ ), and expected proportions according to Benford's Law. +: significant overrepresentation, -: significant underrepresentation,  $p < 0.05$ .

In general, all of the first digit distributions conformed to Benford's Law reasonably well with a few exceptions. In wave 1, prices denominated in national currencies showed a high frequency of "9" while prices expressed in euro had an overly high occurrence of "1" and "7" as the leading digit. Similar to wave 1, the patterns at wave 3 contained more "7" and "9" than expected by Benford's Law. On the other hand, wave 2 seemed to follow the law more closely when compared to wave 1 and wave 3.

Contrary to the first digit distributions, the second digit distributions exhibited strong divergence from Benford's Law. The analysis results are replicated in Table 2.9.



Table 2.9: Second digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after)

Digit	Wave 1 National currency		Wave 1 Euro		Wave 2 Euro		Wave 3 Euro		Benford
	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	
0	67	0.089-	66	0.088-	73	0.111	64	0.096	0.11968
1	60	0.080-	85	0.113	76	0.116	60	0.090	0.11389
2	63	0.084-	94	0.126	81	0.123	84	0.126	0.10882
3	57	0.076-	63	0.084	53	0.081	56	0.084	0.10433
4	87	0.116	89	0.119	63	0.096	66	0.099	0.10031
5	74	0.099	73	0.097	89	0.136+	82	0.123+	0.09668
6	59	0.079	71	0.095	45	0.069-	41	0.061-	0.09337
7	48	0.064-	57	0.076	37	0.056-	46	0.069	0.09035
8	55	0.073	65	0.087	50	0.076	45	0.067	0.08757
9	179	0.239+	86	0.115+	89	0.136+	125	0.187+	0.08500
		$\chi^2 (9, n = 749) = 243.13, p < 0.001$			$\chi^2 (9, n = 749) = 23.19, p = 0.006$			$\chi^2 (9, n = 656) = 49.08, p < 0.001$	$\chi^2 (9, n = 669) = 111.39, p < 0.001$

Frequencies ( $f$ ), observed proportions ( $f_{rel}$ ), and expected proportions according to Benford's Law. +: significant overrepresentation, -: significant underrepresentation,  $p < 0.05$ .

In wave 1, prices denominated in national currencies deviated significantly from Benford's distribution with an overrepresentation of "9" (24%) and underrepresentation of "7" and smaller digits 0, 1, 2, and 3 (ranged from 6% to 9%). At the same point in time, prices stated in euro still departed from the law although the discrepancies were not as prominent. Again, "9" occurred more but "1" appeared less than expected. In both wave 2 and wave 3, the results were much alike with "5" and "9" excessively represented.

For those prices consisting of three or more digits, the results were even more divergent from Benford's Law. The summary chart is duplicated in Table 2.10.

Table 2.10: Third digits of prices of bakery products, drinks, and cosmetics in three different waves in the euro introduction (wave1=before, wave2=half a year after, wave3=a full year after)

Digit	Wave 1 National currency		Wave 1 Euro		Wave 2 Euro		Wave 3 Euro		Benford
	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	$f$	$f_{rel}$	
0	73	0.135+	41	0.093	55	0.140+	56	0.144+	0.10178
1	5	0.009-	43	0.097	22	0.056-	18	0.046-	0.10138
2	24	0.045-	34	0.077	27	0.069-	19	0.049-	0.10097
3	25	0.046-	39	0.088	13	0.033-	9	0.023-	0.10057
4	30	0.056-	52	0.118	30	0.076	27	0.069	0.10018
5	89	0.165+	52	0.118	70	0.178+	67	0.172+	0.09979
6	16	0.030-	29	0.066-	26	0.066-	22	0.057-	0.09940
7	16	0.030-	65	0.147+	25	0.064-	21	0.054-	0.09902
8	43	0.080	40	0.090	26	0.066-	25	0.064-	0.09864
9	218	0.404+	47	0.106	99	0.252+	125	0.321+	0.09827
		$\chi^2 (9, n = 539) = 686.23, p < 0.001$			$\chi^2 (9, n = 442) = 22.35, p = 0.008$			$\chi^2 (9, n = 393) = 169.81, p < 0.001$	$\chi^2 (9, n = 389) = 293.03, p < 0.001$

Frequencies ( $f$ ), observed proportions ( $f_{rel}$ ), and expected proportions according to Benford's Law. +: significant overrepresentation, -: significant underrepresentation,  $p < 0.05$ .

Prices denominated in national currencies departed from Benford's Law substantially with the occurrence of digits 0, 5, and 9 greatly exceeding the expected frequencies. On the other hand, prices in euro in wave 1 deviated in a much smaller degree with digits 6 and 7 as the only exceptions. Wave 2 and wave 3 also had digits 0, 5, and 9 overrepresented. In addition, the authors noted a stronger deviation in the second wave compared to the first and an even more pronounced disagreement in the third wave compared to the second.

#### 2.2.4 Benford's Law and psychological barriers in certain eBay auctions

Another application of Benford's Law involved economics research by Lu, F.O. and Giles, D.E. They used the prices of 1159 professional football tickets to test whether there were psychological price level barriers in eBay auctions [28].

Psychological barriers had been tested in different areas such as daily stock return and gold prices. De Ceuster et al. proved that there was no evidence of psychological

barriers in various stock market indices [10]; however, Aggarwal and Lucey found that psychological barriers existed at gold price levels such as \$100, \$200, etc. [1]. Lu and Giles' paper applied similar concepts to examine eBay auctions.

In this paper, three different psychological barriers were considered as follows:

1. e.g. 100, 200, 300, ...  
 $k \times 100, \quad k = 1, 2, \dots, etc$
2. e.g. ..., 0.1, 0.2, ..., 1, 2, ..., 10, 20, ..., 100, 200, ...  
 $k \times 10^a, \quad k = 1, 2, \dots, 9, \quad a = \dots, -1, 0, 1, \dots$
3. e.g. ..., 1, 1.1, ..., 10, 11, ..., 100, 110, ..., 1000, 1100, ...  
 $k \times 10^a, \quad k = 10, 11, \dots, 99, \quad a = \dots, -1, 0, 1, \dots$

In order to test whether there were psychological barriers, the authors defined three corresponding  $M$ -values:

1.  $M_t^a = [P_t] \bmod 100$
2.  $M_t^b = [100 \times 10^{(\log p_t) \bmod 1}] \bmod 100$
3.  $M_t^c = [1000 \times 10^{(\log p_t) \bmod 1}] \bmod 100$

where  $[P_t]$  is the integer part of the prices;  $M^a$  picks the pair of digits just before the decimal point;  $M^b$  gets the second and third significant digits; and  $M^c$  selects the third and fourth significant digits.

If no psychological barriers are present, the relative frequencies of the  $M$ -values for sample  $t = 1, 2, \dots, n$  as  $n$  approaches infinity are expressed as the limit probability equations [10] below:

$$\lim_{t \rightarrow \infty} \Pr(M_t^a = k) = \frac{1}{100}$$

$$\lim_{t \rightarrow \infty} \Pr(M_t^b = k) = \sum_{i=1}^9 \log \left( \frac{i \times 10^2 + k + 1}{i \times 10^2 + k} \right)$$

$$\lim_{t \rightarrow \infty} \Pr(M_t^c = k) = \sum_{i=1}^9 \sum_{j=0}^9 \log \left( \frac{i \times 10^3 + j \times 10^2 + k + 1}{i \times 10^3 + j \times 10^2 + k} \right)$$

To determine if the ticket price data conform to the limit probabilities above, Kuiper's test statistic was used in the hypothesis testing in which the null hypothesis is that psychological barriers are nonexistent in the prices of professional football tickets traded in eBay auctions. Kuiper's test statistic and its transformed version [35] are defined as

$$V_N = D_N^+ + D_N^-,$$

$$V_N^* = V_N [N^{1/2} + 0.155 + 0.24N^{-1/2}],$$

where  $D_N^+ = \sup_{-\infty < x < \infty} [F_N(x) - F_0(x)]$  and  $D_N^- = \sup_{-\infty < x < \infty} [F_0(x) - F_N(x)]$

The result of the hypothesis testing suggested that the null hypothesis cannot be rejected. Therefore, the authors concluded that there were no psychological barriers in the prices of professional football tickets auctioned on eBay.

## Chapter 3

### 3. Test Statistics

#### 3.1 Likelihood ratio tests for Benford's Law

In this section, we test whether the first two significant digits of a given set of data with  $N$  entries are compatible with Benford's Law for the first two digits where there are 90 possibilities. The null hypothesis and four alternative hypotheses for the cell probabilities  $\pi_i$  used are as follow:

Null Hypothesis

$$\mathbf{H}_0: \pi_i = \log_{10}(1 + 1/i), i = 10, 11, \dots, 99$$

Alternative Hypotheses

$$1. \mathbf{H}_1: \pi_{10} \geq 0, \pi_{11} \geq 0, \dots, \pi_{99} \geq 0; \quad \sum_{i=10}^{99} \pi_i = 1$$

The probabilities  $\pi_i$  are greater than or equal to zero and the likelihood ratio statistic  $\Lambda_1$  for testing  $\mathbf{H}_0$  versus  $\mathbf{H}_1$  is

$$\Lambda_1 = -2 \sum_{i=10}^{99} n_i \left( \log \pi_i - \log \frac{n_i}{N} \right) \approx \chi^2_{(89)}$$

where  $n_i$  = observed cell frequencies and  $\sum_{i=10}^{99} n_i = N$ .

2.  $\mathbf{H}_2: \pi_{10} \geq \pi_{11} \geq \dots \geq \pi_{99} \geq 0; \quad \sum_{i=10}^{99} \pi_i = 1$

The probabilities  $\pi_i$  are non-increasing with  $i$ . To find the maximum likelihood estimates of the  $\pi_i$ , first we let  $z_i = \pi_i - \pi_{i+1}$  and  $z_{99} = \pi_{99}$  and then maximize

$$\Lambda_2 = -2 \left\{ \sum_{i=10}^{99} n_i \left[ \log \pi_i - \log \left( \sum_{j=i}^{99} z_j \right) \right] \right\} \approx \chi_{(89)}^2$$

subject to  $z_i \geq 0$  and  $\sum_{i=10}^{99} i z_i = 1$ .

3.  $\mathbf{H}_3: \alpha \neq 0$

The distribution of the generalized Benford's Law [30] for the first two significant digits,  $D_2$ , is

$$P(D_2) = \frac{i^{-\alpha} - (i+1)^{-\alpha}}{10^{-\alpha} - 100^{-\alpha}}, i = 10, 11, \dots, 99; \alpha \in \Re$$

As  $\alpha \rightarrow 0$ , the generalized Benford's Law approaches Benford's Law. Therefore, the equivalent null hypothesis is  $\mathbf{H}_0: \alpha = 0$ . Again, to compute the maximum likelihood estimates of  $\alpha$ , we need to numerically maximize

$$\Lambda_3 = -2 \left\{ \sum_{i=10}^{99} n_i [\log \pi_i - \log \{i^{-\alpha} - (i+1)^{-\alpha}\}] + N \log(10^{-\alpha} - 100^{-\alpha}) \right\}$$

where  $n_i$  = observed cell frequencies and  $\sum_{i=10}^{99} n_i = N$ . The likelihood ratio statistic,  $\Lambda_3$ , asymptotically has a  $\chi_{(1)}^2$  distribution.

#### 4. $\mathbf{H}_4: \beta \neq -1$

The distribution for the first two significant digits,  $D_2$ , under the Rodriguez family [32] is

$$P(D_2) = \frac{\beta + 1}{90\beta} - \frac{(i + 1)^{\beta+1} - (i)^{\beta+1}}{\beta(100^{\beta+1} - 10^{\beta+1})}, i = 10, 11, \dots, 99; \beta \in \mathfrak{R}$$

The equivalent null hypothesis is  $\mathbf{H}_0: \beta = -1$  because Benford's Law arises as  $\beta$  approaches -1. The maximum likelihood estimates of  $\beta$  can be calculated by maximizing

$$\Lambda_4 = -2[l(\mathbf{H}_0) - l(\mathbf{H}_4)] \approx \chi^2_{(1)}$$

Where

$$l(\mathbf{H}_0) = \sum_{i=10}^{99} n_i \log \pi_i,$$

$$l(\mathbf{H}_4) = \sum_{i=10}^{99} n_i \log[(\beta + 1)(100^{\beta+1} - 10^{\beta+1}) - 90\{(i + 1)^{\beta+1} - i^{\beta+1}\}] \\ + N \log[90\beta(100^{\beta+1} - 10^{\beta+1})], \text{ and}$$

$n_i$  = observed cell frequencies

### 3.2 Tests based on Cramér-von Mises statistics

Here the goodness-of-fit of Benford's Law is examined using the well-known Cramér-von Mises family of goodness-of-fit statistics for discrete distributions [9, 27]. The statistics we consider here are  $W_d^2$ ,  $U_d^2$  and  $A_d^2$  which are the analogues of Cramér-von Mises,

Watson, and Anderson-Darling respectively. Since they are closely related to the popular Pearson's chi-square test, we also include the Pearson goodness-of-fit statistic in this section.

Again we test Benford's Law

$$\mathbf{H}_0: \pi_i = \log_{10}(1 + 1/i), i = 10, 11, \dots, 99$$

Against the broadest alternative hypothesis

$$\mathbf{H}_1: \pi_{10} \geq 0, \pi_{11} \geq 0, \dots, \pi_{99} \geq 0; \sum_{i=10}^{99} \pi_i = 1$$

First, let  $S_i = \sum_{j=10}^i \hat{\pi}_j$  and  $T_i = \sum_{j=10}^i \pi_j$  be the cumulative observed and expected proportions respectively. Also, let  $Z_i = S_i - T_i$  on which the Cramèr-von Mises statistics are based. Second, let  $t_i = (\pi_i + \pi_{i+1})/2$  for  $i = 10, \dots, 98$  and  $t_{99} = (\pi_{99} + \pi_{10})/2$  be the weights. Then, define the weighted average of the deviations  $Z_i$  as  $\bar{Z} = \sum_{i=10}^{99} t_i Z_i$ . The Cramèr-von Mises statistics are defined as follows [27]

$$W_d^2 = n^{-1} \sum_{i=10}^{99} Z_i^2 t_i;$$

$$U_d^2 = n^{-1} \sum_{i=10}^{99} (Z_i - \bar{Z})^2 t_i;$$

$$A_d^2 = n^{-1} \sum_{i=10}^{99} Z_i^2 t_i / \{T_i(1 - T_i)\}.$$

Since both  $S_{99}$  and  $T_{99} = 1$  and thus  $Z_{99} = 0$  in the above summations, the last term in  $W_d^2$  will be zero. In addition, the last term in  $A_d^2$  will be set to zero because it is an indeterminate form of 0/0.



The above statistics -  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$  and Pearson goodness-of-fit statistic can be expressed in matrix form [27] as follows:

$$W_d^2 = \mathbf{Z}_n^T \mathbf{E} \mathbf{Z}_n / n$$

$$U_d^2 = \mathbf{Z}_n^T (\mathbf{I} - \mathbf{E} \mathbf{1} \mathbf{1}^T) \mathbf{E} (\mathbf{I} - \mathbf{1} \mathbf{1}^T \mathbf{E}) \mathbf{Z}_n / n$$

$$A_d^2 = \mathbf{Z}_n^T \mathbf{E} \mathbf{K} \mathbf{Z}_n / n$$

$$X^2 = n(\hat{\boldsymbol{\pi}} - \boldsymbol{\pi})^T \mathbf{D}^{-1}(\hat{\boldsymbol{\pi}} - \boldsymbol{\pi})$$

where  $\mathbf{A}$  is a lower triangular matrix such that

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

$$\mathbf{Z}_n = (Z_{10}, Z_{11}, \dots, Z_{99})^T;$$

$\mathbf{I}$  is the 90 x 90 identity matrix;

$\mathbf{E}$  is the diagonal matrix with diagonal entries  $t_i$ ;

$\mathbf{D}$  is the diagonal matrix with diagonal entries  $\pi_i$ ;

$\mathbf{K}$  is the diagonal matrix whose  $(i, i)^{th}$  element is  $1/[T_i(1 - T_i)]$ ,

where  $i = 10, 11, \dots, 98$  and  $\mathbf{K}_{99,99} = 0$ ;

$\mathbf{1} = (1, 1, \dots, 1)^T$  is a 90-vector of ones; and

$\boldsymbol{\pi} = (\pi_{10}, \pi_{11}, \dots, \pi_{99})^T$ , the Benford probabilities

$\hat{\boldsymbol{\pi}} = (\hat{\pi}_{10}, \hat{\pi}_{11}, \dots, \hat{\pi}_{99})^T$ , the observed relative frequencies

The asymptotic distributions of the Cramèr-von Mises statistics under the null and contiguous alternative hypotheses are given in Lesperance et al [25].

$$\mathbf{Z} \sim N[E(\mathbf{Z}_n), Var(\mathbf{Z}_n)],$$

$$E(\mathbf{Z}_n) = \boldsymbol{\mu}_z = \mathbf{A}\boldsymbol{\mu} \text{ and } Var(\mathbf{Z}_n) = \boldsymbol{\Sigma}_z = \mathbf{A}(\mathbf{D} - \boldsymbol{\pi}\boldsymbol{\pi}^T)\mathbf{A}^T$$

Consider the asymptotic distribution of statistics of the following form:

$$Q_n = n\mathbf{Z}_n^T \mathbf{M} \mathbf{Z}_n$$

where  $\mathbf{M}$  is a symmetric matrix. Since  $Q_n$  is a continuous function of  $\mathbf{Z}_n$ ,

$$\lim_{n \rightarrow \infty} \mathcal{L}(n\mathbf{Z}_n^T \mathbf{M} \mathbf{Z}_n) = \mathcal{L}(\mathbf{Z}^T \mathbf{M} \mathbf{Z}); \text{ where } E(\mathbf{Z}_n) = \boldsymbol{\mu}_z = \mathbf{A}\boldsymbol{\mu} \text{ and } Var(\mathbf{Z}_n) = \boldsymbol{\Sigma}_z = \mathbf{A}(\mathbf{D} - \boldsymbol{\pi}\boldsymbol{\pi}^T)\mathbf{A}^T$$

Let  $Q = \mathbf{Z}^T \mathbf{M} \mathbf{Z}$  and following Scheffé, H. [33], we write the following:

1.  $\boldsymbol{\Sigma}_z = \mathbf{B}\mathbf{B}^T$  where  $\mathbf{B}$  is a  $m$  by  $m - 1$  matrix. Since the  $m^{th}$  row and column of  $\boldsymbol{\Sigma}_z$  are zeros,  $\boldsymbol{\Sigma}_z$  is non-negative definite with rank  $m - 1$ . Also,  $\mathbf{B}$  is the matrix root of  $\boldsymbol{\Sigma}_z$ .
2. Let  $\mathbf{C}$  be a matrix such that  $\mathbf{C}[\mathbf{B}^T \mathbf{M} \mathbf{B}]\mathbf{C}^T = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{k-1}) = \boldsymbol{\Lambda}$ , where  $\lambda_i$  are the eigenvalues of  $[\mathbf{B}^T \mathbf{M} \mathbf{B}]$  and  $\mathbf{C}$  has the corresponding eigenvectors in its columns.

We transform the random variable from  $\mathbf{Z}$  to  $\mathbf{Y}$  such that  $Var(\mathbf{Y}) = \mathbf{I}$ . If we let

$\mathbf{Y} = \mathbf{C}\mathbf{B}^-\mathbf{Z}$  where  $\mathbf{B}^-$  is a generalized inverse of  $\mathbf{B}$ , then the distribution of  $\mathbf{Y}$  has normal distribution with  $E(\mathbf{Y}) = \mathbf{C}\mathbf{B}^-\boldsymbol{\mu}_z = \mathbf{C}\mathbf{B}^-\mathbf{A}\boldsymbol{\mu}$  and  $Var(\mathbf{Y}) = \mathbf{C}\mathbf{B}^-\boldsymbol{\Sigma}_z(\mathbf{B}^-)^T\mathbf{C}^T = \mathbf{I}$ . Thus,  $\mathbf{Z} = \mathbf{B}\mathbf{C}^T\mathbf{Y}$  and  $Q = \mathbf{Z}^T \mathbf{M} \mathbf{Z} = \mathbf{Y}^T [\mathbf{C}\mathbf{B}^T \mathbf{M} \mathbf{B} \mathbf{C}^T] \mathbf{Y} = \mathbf{Y}^T \boldsymbol{\Lambda} \mathbf{Y} = \sum \lambda_i y_i^2$ .

The statistic  $Q$  is a linear combination of independent  $\chi^2(1, \mu_{y_i}^2)$  random variables where  $\mu_{y_i}^2$  is the non-centrality parameter of the  $i^{th}$  term. The asymptotic distribution of  $Q_n$  is a linear combination of independent  $\chi^2_{(1)}$  random variables when  $\boldsymbol{\mu} = \mathbf{0}$ .

The methods to compute the eigenvalues and the asymptotic percentage points for the Cramèr-von Mises statistics are given in Lesperance et al [25]. Tables 3.1 – 3.3 shows eigenvalues for the null hypothesis case when  $\boldsymbol{\mu} = \mathbf{0}$  and Table 3.4 provides asymptotic percentage points for the Cramèr-von Mises statistics.

Table 3.1: Eigenvalues for Cramèr-von Mises statistics -  $W_d^2$

$W_d^2$									
1.01e-1	2.54e-2	1.13e-2	6.37e-3	4.09e-3	2.85e-3	2.10e-3	1.62e-3	1.29e-3	1.05e-3
8.75e-4	7.42e-4	6.39e-4	5.57e-4	4.92e-4	4.39e-4	3.94e-4	3.57e-4	3.24e-4	2.96e-4
2.70e-4	2.48e-4	2.28e-4	2.10e-4	1.94e-4	1.79e-4	1.66e-4	1.54e-4	1.43e-4	1.33e-4
1.24e-4	1.16e-4	1.08e-4	1.01e-4	9.50e-5	8.91e-5	8.37e-5	7.87e-5	7.40e-5	6.97e-5
6.57e-5	6.19e-5	5.85e-5	5.52e-5	5.22e-5	4.94e-5	4.68e-5	4.43e-5	4.20e-5	3.98e-5
3.78e-5	3.59e-5	3.41e-5	3.24e-5	3.08e-5	2.92e-5	2.78e-5	2.65e-5	2.52e-5	2.40e-5
2.29e-5	2.18e-5	2.08e-5	1.98e-5	1.89e-5	1.80e-5	1.72e-5	1.64e-5	1.56e-5	1.49e-5
1.42e-5	1.35e-5	1.29e-5	1.23e-5	1.17e-5	1.12e-5	1.06e-5	1.01e-5	9.64e-6	9.16e-6
8.70e-6	8.26e-6	7.83e-6	7.40e-6	6.99e-6	6.58e-6	6.16e-6	5.74e-6	5.27e-6	

Table 3.2: Eigenvalues for Cramer-von Mises statistics -  $U_d^2$

$U_d^2$									
2.54e-2	2.53e-2	6.39e-3	6.35e-3	2.86e-3	2.84e-3	1.63e-3	1.61e-3	1.06e-3	1.04e-3
7.51e-4	7.33e-4	5.67e-4	5.49e-4	4.51e-4	4.30e-4	3.72e-4	3.48e-4	3.12e-4	2.88e-4
2.63e-4	2.43e-4	2.23e-4	2.06e-4	1.90e-4	1.76e-4	1.63e-4	1.52e-4	1.41e-4	1.32e-4
1.23e-4	1.15e-4	1.07e-4	1.00e-4	9.41e-5	8.83e-5	8.30e-5	7.80e-5	7.34e-5	6.92e-5
6.52e-5	6.15e-5	5.81e-5	5.49e-5	5.19e-5	4.91e-5	4.65e-5	4.41e-5	4.18e-5	3.96e-5
3.76e-5	3.57e-5	3.39e-5	3.22e-5	3.06e-5	2.91e-5	2.77e-5	2.64e-5	2.51e-5	2.39e-5
2.28e-5	2.17e-5	2.07e-5	1.97e-5	1.88e-5	1.79e-5	1.71e-5	1.63e-5	1.56e-5	1.48e-5
1.42e-5	1.35e-5	1.29e-5	1.23e-5	1.17e-5	1.11e-5	1.06e-5	1.01e-5	9.62e-6	9.15e-6
8.69e-6	8.25e-6	7.81e-6	7.39e-6	6.98e-6	6.57e-6	6.16e-6	5.73e-6	5.26e-6	

Table 3.3: Eigenvalues for Cramer-von Mises statistics -  $A_d^2$ 

$A_d^2$									
5.00e-1	1.67e-1	8.31e-2	4.97e-2	3.31e-2	2.35e-2	1.76e-2	1.36e-2	1.08e-2	8.83e-3
7.32e-3	6.16e-3	5.25e-3	4.52e-3	3.93e-3	3.44e-3	3.04e-3	2.70e-3	2.42e-3	2.17e-3
1.96e-3	1.78e-3	1.62e-3	1.48e-3	1.35e-3	1.24e-3	1.15e-3	1.06e-3	9.82e-4	9.12e-4
8.49e-4	7.92e-4	7.40e-4	6.93e-4	6.49e-4	6.10e-4	5.74e-4	5.41e-4	5.10e-4	4.82e-4
4.56e-4	4.32e-4	4.09e-4	3.88e-4	3.69e-4	3.51e-4	3.34e-4	3.18e-4	3.04e-4	2.90e-4
2.77e-4	2.65e-4	2.53e-4	2.43e-4	2.33e-4	2.23e-4	2.14e-4	2.05e-4	1.97e-4	1.90e-4
1.83e-4	1.76e-4	1.69e-4	1.63e-4	1.57e-4	1.52e-4	1.46e-4	1.41e-4	1.36e-4	1.32e-4
1.28e-4	1.23e-4	1.19e-4	1.16e-4	1.12e-4	1.08e-4	1.05e-4	1.02e-4	9.88e-5	9.59e-5
9.30e-5	9.03e-5	8.77e-5	8.52e-5	8.28e-5	8.05e-5	7.83e-5	7.62e-5	7.41e-5	

Table 3.4: Asymptotic percentage points for Cramer-von Mises statistics

	0.5000	0.250	0.100	0.050	0.025	0.010
$W_d^2$ chi-square approx	0.112	0.209	0.352	0.467	0.585	0.744
$U_d^2$ chi-square approx	0.068	0.106	0.153	0.188	0.223	0.268
$A_d^2$ chi-square approx	0.713	1.219	1.94	2.51	3.084	3.857
$\chi^2$	88.334	97.599	106.469	112.022	116.989	122.942

Other than using Imhof's numerical method [22] which requires numerical integration in one dimension of a closed form expression to obtain the upper-tail probabilities for  $Q$  given  $\mu$ , we also can use a chi-square approximation which needs the first three cumulants of the statistics. Let  $\kappa_1, \kappa_2$ , and  $\kappa_3$  be the first three cumulants of a statistic,  $Q$ . Then approximately,  $Q \sim a + b\chi_{(p)}^2$  where  $b = \kappa_3/(4\kappa_2)$ ,  $p = 8\kappa_2^3/\kappa_3^2$ , and  $a = \kappa_1 - bp$ . Here  $\kappa_1 = \sum \lambda_i (1 + \mu_{y_i}^2)$ ,  $\kappa_2 = 2 \sum \lambda_i^2 (1 + 2\mu_{y_i}^2)$ , and  $\kappa_3 = 8 \sum \lambda_i^3 (1 + 3\mu_{y_i}^2)$  [22].

We use the results for contiguous alternatives above to derive approximate power curves for CVM statistics for given, fixed sample sizes. For example size  $n$  and for a specified alternative distribution, let  $E(\hat{\pi}) = \mathbf{p}(n) = \boldsymbol{\pi} + \boldsymbol{\mu}/\sqrt{n}$ , where  $\boldsymbol{\pi}$  is the vector of Benford probabilities. Solving for  $\boldsymbol{\mu} = \sqrt{n}[\mathbf{p}(n) - \boldsymbol{\pi}]$ , and substituting for  $\boldsymbol{\mu}$  in the asymptotic distribution for  $Q$  results in an approximation to the power of the CVM statistics.

The two key features of the testing problem concern discrete data and circular distribution. If the data are discrete, the issue of "ties" in the data arises while such

problem does not exist in the continuous distribution. To address the discrete nature of the data, we used modified tests in the paper. On the other hand, circular data exhibit the testing problem that most statistics will change with different starting points around a circle. To circumvent this situation, we used the statistic  $U_d^2$  because its value will not change with different starting point around a circle.

### 3.3 Simultaneous confidence intervals for multinomial probabilities

In the literature, there are several simultaneous confidence intervals for multinomial probabilities. It has been shown that Sison and Glaz simultaneous confidence interval is the best among others and that the coverage probability for Fitzpatrick and Scott simultaneous confidence interval is often too high [37]. Seven of the techniques will be considered in this section.

First, let  $n_1, n_2, \dots, n_k$  be the observed cell frequencies in a sample of size  $N$  from the multinomial distribution, i.e.  $P(n_1, n_2, \dots, n_k) = N! \pi_1^{n_1} \pi_2^{n_2} \dots \pi_k^{n_k} / n_1! n_2! \dots n_k!$ , where  $\pi_i \geq 0$ ,  $\sum_{i=1}^k \pi_i = 1$  and  $\sum_{i=1}^k n_i = N$ . Second, let  $\chi_{\alpha, v}^2$  be the upper  $\alpha^{th}$  quantile of the chi-square distribution with  $v$  degrees of freedom. Last, let  $Z_\alpha$  be the upper  $\alpha^{th}$  quantile of the standard normal distribution.

1. Quesenberry, C.P. and Hurst, D.C. [31]

$$\pi_i = \frac{A + 2n_i \pm \sqrt{A[A + 4n_i(N - n_i)/N]}}{2(N + A)}, i = 1, 2, \dots, k$$

where  $A = \chi_{\alpha, k-1}^2$

2. Goodman, L.A. [16]

$$\pi_i = \frac{B + 2n_i \pm \sqrt{B[B + 4n_i(N - n_i)/N]}}{2(N + B)}, i = 1, 2, \dots, k$$

where  $B = \chi_{\alpha/k, 1}^2$

3. Bailey, B.J.R. angular transformation [2]

$$\pi_i = \left\{ \sin \left[ \sin^{-1} \left( \sqrt{\frac{n_i + \frac{3}{8}}{N + \frac{3}{4}}} \right) \pm \sqrt{\frac{B}{4N + 2}} \right] \right\}^2, i = 1, 2, \dots, k$$

where  $B = \chi_{\alpha/k,1}^2$

4. Bailey, B.J.R. square root transformation [2]

$$\pi_i = \left\{ \frac{\sqrt{(n_i + \frac{3}{8})/(N + \frac{1}{8})} \pm \sqrt{C[C + 1 - (n_i + \frac{3}{8})/(N + \frac{1}{8})]}}{(C + 1)} \right\}^2, i = 1, 2, \dots, k$$

where  $C = B/(4N)$

5. Fitzpatrick, S. and Scott, A. [14]

$$\pi_i = \hat{\pi}_i \pm \frac{D}{2\sqrt{N}}, i = 1, 2, \dots, k$$

where  $D = Z_{\alpha/4}$

6. Sison, C.P. and Glaz, J. [37]

To compute this simultaneous confidence interval, a computer is needed because this procedure does not have a closed form. First, let  $V_i$  be independent Poisson random variables with mean  $n_i$ . Second, let  $Y_i$  be its truncation to  $[n_i - \tau, n_i + \tau]$ , where  $i = 1, 2, \dots, k$  and  $\tau$  is a constant. Third, let  $n_1^*, n_2^*, \dots, n_k^*$  have a multinomial distribution with  $N$  observations and cell probabilities

$\hat{\pi}_1, \hat{\pi}_2, \dots, \hat{\pi}_k$ . Define the central and factorial moments of  $Y_i$  as following.

$\mu_i = E(Y_i)$  and  $\sigma_i^2 = Var(Y_i)$ ,  $\mu_{(r)} = E[Y_i(Y_i - 1) \cdots (Y_i - r + 1)]$  and  $\mu_{r,i} = E(Y_i - \mu_i)^r$ . Set  $\gamma_1 = (1/k)(\sum_{i=1}^k \mu_{3,i})/\sqrt{k}[(1/k)(\sum_{i=1}^k \sigma_i^2)]^{3/2}$  and  $\gamma_2 = (1/k)(\sum_{i=1}^k \mu_{4,i} - 3\sigma_i^4)/\sqrt{k}[(1/k)(\sum_{i=1}^k \sigma_i^2)]^2$ . Then

$$v(\tau) = \frac{N!}{N^N e^{-N}} \left\{ \prod_{i=1}^k P(n_i - \tau \leq V_i \leq n_i + \tau) \right\} f_e \left( \frac{N - \sum_{i=1}^k \mu_i}{\sqrt{\sum_{i=1}^k \sigma_i^2}} \right) \frac{1}{\sqrt{\sum_{i=1}^k \sigma_i^2}}, \text{ where}$$

$$f_e(x) = \left( \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \right) \left\{ 1 + \frac{\gamma_1}{6} (x^3 - 3x) + \frac{\gamma_2}{24} (x^4 - 6x^2 + 3) + \frac{\gamma_1^2}{72} (x^6 - 15x^4 + 45x^2 - 15) \right\}.$$

$$\pi_i = \hat{\pi}_i - \frac{\tau}{N} \leq \pi_i \leq \hat{\pi}_i + \frac{\tau + 2\gamma}{N}, i = 1, 2, \dots, k$$

where the integer  $\tau$  satisfies the following two conditions:

$$v(\tau) < 1 - \alpha < v(\tau + 1) \text{ and } \gamma = [(1 - \alpha) - v(\tau)]/[v(\tau + 1) - v(\tau)]$$

## 7. Univariate approximate Binomial confidence intervals

$$\pi_i = \hat{\pi}_i \pm G \sqrt{\frac{\hat{\pi}_i(1 - \hat{\pi}_i)}{N}}, i = 1, 2, \dots, k$$

where  $G = Z_{\alpha/2}$



## Chapter 4

### 4. Numerical Results

In this section, we use simulation to examine the test statistics and confidence intervals. Since there are 90 numbers in digits 10 to 99, we need to use large sample sizes. Here, we use two sample sizes,  $n = 1000$  and  $n = 2000$ . Six alternative distributions were used to generate one thousand ( $N = 1000$ ) random multinomial samples. Table 4.1 summaries each of the distributions.

Table 4.1: Multinomial distribution used in simulation and numerical study

Distribution	Parameter Values	Notes
<b>Benford</b>		
<b>Uniform</b>	$\pi = 1/90$ for each digit	
<b>Contaminated (additive) Benford</b>	$\alpha = 0.02, 0.06$	1
<b>Contaminated (multiplicative) Benford</b>	$\alpha = 1.2, 1.5$	2
<b>Generalized Benford</b>	$\alpha = -1, -0.9, \dots, 0.9, 1$	
<b>Uniform/Benford mixture</b>	$\alpha = 0.1, 0.2, \dots, 0.5$	3
<b>Hill/Benford mixture</b>	$\alpha = 0.1, 0.2, \dots, 0.5$	4

<sup>1</sup> each  $\pi_i$  is increased in turn by  $\alpha$  and the remaining 89 digits are rescaled so that the sum of all  $\pi_i$  is one

<sup>2</sup> each  $\pi_i$  is set to  $\alpha\pi_i$  in turn and the remaining 89 digits are rescaled so that the sum of all  $\pi_i$  is one

<sup>3</sup>  $\alpha$  is the proportion Uniform distribution

<sup>4</sup>  $\alpha$  is the proportion Hill distribution

Many people will think that the frequencies of digits are uniformly distributed so one of the alternative distributions we consider here is the uniform distribution.

For the additive and multiplicative contaminated Benford distributions, we increased and multiplied the digit's Benford probability by  $\alpha$  respectively, then scaled the probabilities of the remaining digits to make the sum equal to one. These two alternative distributions are common in situations where the same transaction is processed many times. Note that the level of additive contamination is very large as compared with Benford probabilities, see Table 4.2.

Table 4.2: The relative frequencies of the 1st two digits of Benford's distribution

1 <sup>st</sup> digit	2 <sup>nd</sup> digit									
	0	1	2	3	4	5	6	7	8	9
<b>1</b>	0.0414	0.0378	0.0348	0.0322	0.0300	0.0280	0.0263	0.0248	0.0235	0.0223
<b>2</b>	0.0212	0.0202	0.0193	0.0185	0.0177	0.0170	0.0164	0.0158	0.0152	0.0147
<b>3</b>	0.0142	0.0138	0.0134	0.0130	0.0126	0.0122	0.0119	0.0116	0.0113	0.0110
<b>4</b>	0.0107	0.0105	0.0102	0.0100	0.0098	0.0095	0.0093	0.0091	0.0090	0.0088
<b>5</b>	0.0086	0.0084	0.0083	0.0081	0.0080	0.0078	0.0077	0.0076	0.0074	0.0073
<b>6</b>	0.0072	0.0071	0.0069	0.0068	0.0067	0.0066	0.0065	0.0064	0.0063	0.0062
<b>7</b>	0.0062	0.0061	0.0060	0.0059	0.0058	0.0058	0.0057	0.0056	0.0055	0.0055
<b>8</b>	0.0054	0.0053	0.0053	0.0052	0.0051	0.0051	0.0050	0.0050	0.0049	0.0049
<b>9</b>	0.0048	0.0047	0.0047	0.0046	0.0046	0.0045	0.0045	0.0045	0.0044	0.0044

We also consider two parametric alternative distributions, Generalized Benford and Uniform/Benford mixture and one empirical alternative distribution, the "Hill" distribution which was generated by Hill in 1988 [6, 19]. The relative frequencies of the first two digits of the "Hill" distribution are shown in Table 4.3.

Table 4.3: The relative frequencies of the 1st two digits of “Hill” distribution

2 <sup>nd</sup> digit										
1 <sup>st</sup> digit	0	1	2	3	4	5	6	7	8	9
1	0.0085	0.0156	0.0172	0.0160	0.0154	0.0147	0.0165	0.0188	0.0107	0.0135
2	0.0058	0.0106	0.0117	0.0109	0.0105	0.0100	0.0112	0.0128	0.0073	0.0092
3	0.0060	0.0110	0.0122	0.0113	0.0109	0.0104	0.0116	0.0133	0.0076	0.0096
4	0.0077	0.0141	0.0156	0.0145	0.0140	0.0133	0.0149	0.0170	0.0097	0.0122
5	0.0056	0.0103	0.0113	0.0106	0.0102	0.0097	0.0109	0.0124	0.0071	0.0089
6	0.0091	0.0166	0.0184	0.0171	0.0165	0.0157	0.0176	0.0201	0.0115	0.0144
7	0.0070	0.0127	0.0140	0.0131	0.0126	0.0120	0.0134	0.0154	0.0088	0.0110
8	0.0049	0.0089	0.0098	0.0092	0.0088	0.0084	0.0094	0.0108	0.0061	0.0077
9	0.0034	0.0061	0.0068	0.0063	0.0061	0.0058	0.0065	0.0074	0.0042	0.0053

When Benford is the generating distribution, the probabilities of rejection are close to 0.05. Table 4.4 shows the proportion of multinomial samples of sizes 1000 and 2000 rejected at the 0.05 level under Benford distribution. With  $N = 1000$  replications, the maximum standard error is 0.016 is attained for probability equal to 0.5 and is computed as follow:

$$s.e. = \sqrt{pq/N} = \sqrt{0.5(1 - 0.5)/1000} = 0.016$$

For both sample sizes of  $N = 1000$  replications, except for the LR-multinomial test statistics in which the Type I error rate is slightly over an acceptable rate, all of the remaining ten test statistics show acceptable size. For the remaining power calculations, the test statistics were not size-adjusted, that is adjusted for the deviation from 0.05 given in Table 4.4. Table 4.5 shows that all eleven test statistics have excellent power for detecting the error when the generating distribution is the Uniform distribution.

Table 4.4: Proportion of simulated data sets rejecting the null hypothesis of Benford's Law,  
N = 1000 replications

Test	n = 1000	n = 2000
<i>Benford Distribution simulated data</i>		
LR-multinomial	0.074	0.064
LR-decreasing	0.050	0.050
LR-generalized Benford	0.051	0.047
LR-Rodriguez	0.052	0.058
$W_d^2$ Imhof	0.046	0.042
$W_d^2$ chi-square approx.	0.044	0.041
$U_d^2$ Imhof	0.046	0.052
$U_d^2$ chi-square approx.	0.045	0.051
$A_d^2$ Imhof	0.045	0.047
$A_d^2$ chi-square approx.	0.042	0.045
Pearson's $\chi^2$	0.062	0.052

Table 4.5: Proportion of simulated data sets rejecting the null hypothesis when simulated data  
are from Uniform distribution, N = 1000 replications

Test	n = 1000	n = 2000
<i>Uniform Distribution simulated data</i>		
LR-multinomial	1.000	1.000
LR-decreasing	0.958	0.999
LR-generalized Benford	1.000	1.000
LR-Rodriguez	1.000	1.000
$W_d^2$ Imhof	0.805	0.609
$W_d^2$ chi-square approx.	1.000	1.000
$U_d^2$ Imhof	0.998	0.949
$U_d^2$ chi-square approx.	1.000	1.000
$A_d^2$ Imhof	0.916	0.762
$A_d^2$ chi-square approx.	1.000	1.000
Pearson's $\chi^2$	1.000	1.000

Since there are so many results, we only include some summaries in this section although we used simulations to investigate the empirical power for each of eleven test statistics and the remaining five alternative distributions. We include the rest of the results in appendix B.

Table 4.6: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from the contaminated additive Benford distribution for digit 10 with  $\alpha = 0.02$ ,  $N = 1000$  replications

Test	n = 1000	n = 2000
<b><i>Contaminated (additive) Benford Distribution for digit 10 (<math>\alpha = 0.02</math>)</i></b>		
LR-multinomial	0.203	0.364
LR-decreasing	0.440	0.705
LR-generalized Benford	0.191	0.355
LR-Rodriguez	0.260	0.453
$W_d^2$ Imhof	0.177	0.381
$W_d^2$ chi-square approx.	0.175	0.376
$U_d^2$ Imhof	0.146	0.292
$U_d^2$ chi-square approx.	0.143	0.292
$A_d^2$ Imhof	0.319	0.614
$A_d^2$ chi-square approx.	0.310	0.605
Pearson's $\chi^2$	0.166	0.368

Table 4.7: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from the contaminated additive Benford distribution for digit 10 with  $\alpha = 0.06$ ,  $N = 1000$  replications

Test	n = 1000	n = 2000
<b><i>Contaminated (additive) Benford Distribution for digit 10 (<math>\alpha = 0.06</math>)</i></b>		
LR-multinomial	0.981	1.000
LR-decreasing	0.999	1.000
LR-generalized Benford	0.880	0.996
LR-Rodriguez	0.960	0.998
$W_d^2$ Imhof	0.963	1.000
$W_d^2$ chi-square approx.	0.960	1.000
$U_d^2$ Imhof	0.931	0.999
$U_d^2$ chi-square approx.	0.930	0.999
$A_d^2$ Imhof	1.000	1.000
$A_d^2$ chi-square approx.	1.000	1.000
Pearson's $\chi^2$	0.988	1.000

Table 4.8: Proportion of simulated data set rejecting the null hypothesis when simulated data are from the contaminated multiplicative Benford distribution for digit 10 with  $\alpha = 1.2$ , N =1000 replications

Test	n = 1000	n = 2000
<b><i>Contaminated (multiplicative) Benford Distribution for digit 10 (<math>\alpha = 1.2</math>)</i></b>		
LR-multinomial	0.093	0.094
LR-decreasing	0.107	0.137
LR-generalized Benford	0.078	0.092
LR-Rodriguez	0.085	0.128
$W_d^2$ Imhof	0.073	0.088
$W_d^2$ chi-square approx.	0.070	0.085
$U_d^2$ Imhof	0.062	0.078
$U_d^2$ chi-square approx.	0.061	0.078
$A_d^2$ Imhof	0.094	0.135
$A_d^2$ chi-square approx.	0.091	0.126
Pearson's $\chi^2$	0.062	0.078

Table 4.9: Proportion of simulated data set rejecting the null hypothesis when simulated data are from the contaminated multiplicative Benford distribution for digit 10 with  $\alpha = 1.5$ , N =1000 replications

Test	n = 1000	n = 2000
<b><i>Contaminated (multiplicative) Benford Distribution for digit 10 (<math>\alpha = 1.5</math>)</i></b>		
LR-multinomial	0.226	0.392
LR-decreasing	0.470	0.735
LR-generalized Benford	0.205	0.371
LR-Rodriguez	0.275	0.481
$W_d^2$ Imhof	0.187	0.397
$W_d^2$ chi-square approx.	0.183	0.394
$U_d^2$ Imhof	0.158	0.307
$U_d^2$ chi-square approx.	0.155	0.304
$A_d^2$ Imhof	0.344	0.652
$A_d^2$ chi-square approx.	0.335	0.646
Pearson's $\chi^2$	0.180	0.392

Table 4.10: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Generalized Benford distribution with  $\alpha = -0.1$ , N = 1000 replications

Test	n = 1000	n = 2000
<i>Generalized Benford Distribution (<math>\alpha = -0.1</math>)</i>		
LR-multinomial	0.124	0.183
LR-decreasing	0.165	0.348
LR-generalized Benford	0.559	0.832
LR-Rodriguez	0.114	0.168
$W_d^2$ Imhof	0.555	0.813
$W_d^2$ chi-square approx.	0.554	0.809
$U_d^2$ Imhof	0.313	0.560
$U_d^2$ chi-square approx.	0.308	0.555
$A_d^2$ Imhof	0.548	0.827
$A_d^2$ chi-square approx.	0.538	0.819
Pearson's $\chi^2$	0.162	0.245

Table 4.11: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Generalized Benford distribution with  $\alpha = 0.1$ , N = 1000 replications

Test	n = 1000	n = 2000
<i>Generalized Benford Distribution (<math>\alpha = 0.1</math>)</i>		
LR-multinomial	0.143	0.170
LR-decreasing	0.260	0.475
LR-generalized Benford	0.547	0.848
LR-Rodriguez	0.065	0.112
$W_d^2$ Imhof	0.545	0.844
$W_d^2$ chi-square approx.	0.538	0.843
$U_d^2$ Imhof	0.286	0.576
$U_d^2$ chi-square approx.	0.285	0.574
$A_d^2$ Imhof	0.540	0.835
$A_d^2$ chi-square approx.	0.536	0.833
Pearson's $\chi^2$	0.047	0.092

Table 4.12: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Uniform/Benford Mixture distribution with  $\alpha = 0.1$ ,  $N = 1000$  replications

Test	n = 1000	n = 2000
<i>Uniform/Benford Mixture Distribution (<math>\alpha = 0.1</math>)</i>		
LR-multinomial	0.129	0.159
LR-decreasing	0.146	0.281
LR-generalized Benford	0.498	0.762
LR-Rodriguez	0.084	0.091
$W_d^2$ Imhof	0.487	0.762
$W_d^2$ chi-square approx.	0.481	0.761
$U_d^2$ Imhof	0.278	0.508
$U_d^2$ chi-square approx.	0.274	0.503
$A_d^2$ Imhof	0.495	0.772
$A_d^2$ chi-square approx.	0.487	0.763
Pearson's $\chi^2$	0.155	0.204

Table 4.13: Proportion of simulated data sets rejecting the null hypothesis when simulated data are from Hill/Benford Mixture distribution with  $\alpha = 0.1$ ,  $N = 1000$  replications

Test	n = 1000	n = 2000
<i>Hill/Benford Mixture Distribution (<math>\alpha = 0.1</math>)</i>		
LR-multinomial	0.129	0.164
LR-decreasing	0.170	0.291
LR-generalized Benford	0.371	0.625
LR-Rodriguez	0.099	0.106
$W_d^2$ Imhof	0.381	0.648
$W_d^2$ chi-square approx.	0.374	0.640
$U_d^2$ Imhof	0.279	0.492
$U_d^2$ chi-square approx.	0.278	0.490
$A_d^2$ Imhof	0.375	0.627
$A_d^2$ chi-square approx.	0.369	0.623
Pearson's $\chi^2$	0.141	0.198



From Figure 4.1 to Figure 4.4, we see that LR-decreasing is the best performing test statistic for the contaminated additive ( $\alpha = 0.02$ ) distribution when the first two significant digits are from 10 – 14. For the digits 15 or above, Pearson's  $\chi^2$  and LR-multinomial outperform other test statistics. On the other hand, the best performing test statistics for the contaminated additive ( $\alpha = 0.06$ ) distribution are LR-multinomial, Pearson's  $\chi^2$ , LR-decreasing and  $U_d^2$  for digits 10 – 99. The Rodriguez and generalized Benford's distributions do not accommodate for the additive contamination well.

Similar to the contaminated additive Benford distribution, Figure 4.5 to Figure 4.8 show that LR-decreasing is the best test statistic for the contaminated multiplicative Benford distribution when digits are small. However, LR-multinomial and Pearson's  $\chi^2$  are outperforming others for detecting larger digits.

From Figure 4.9 to Figure 4.12, LR-generalized Benford distribution is the best test statistic for all the alpha values from -1 to 1. In addition,  $W_d^2$  and  $A_d^2$  perform very well as the alpha value approaches zero.

We find that the best performing test statistics are LR-generalized Benford,  $W_d^2$  and  $A_d^2$  for both Uniform/Benford and Hill/Benford mixture distributions from Figure 4.13 to Figure 4.16.

Figure 4.1: Simulated power for  $n = 1000$  samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.02$ ,  $N = 1000$  replications, significance level 0.05.

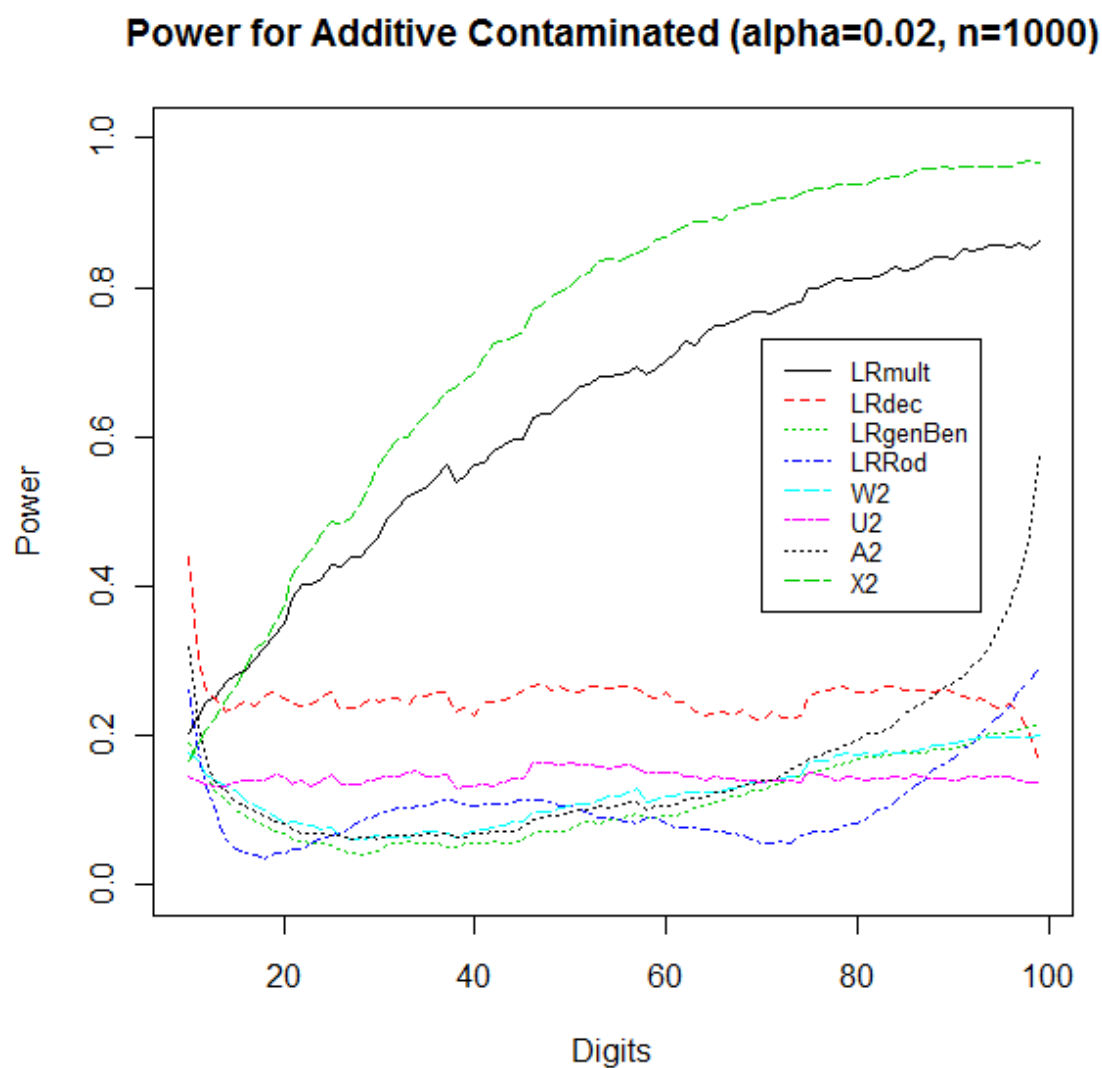


Figure 4.2: Simulated power for  $n = 2000$  samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.02$ ,  $N = 1000$  replications, significance level 0.05.

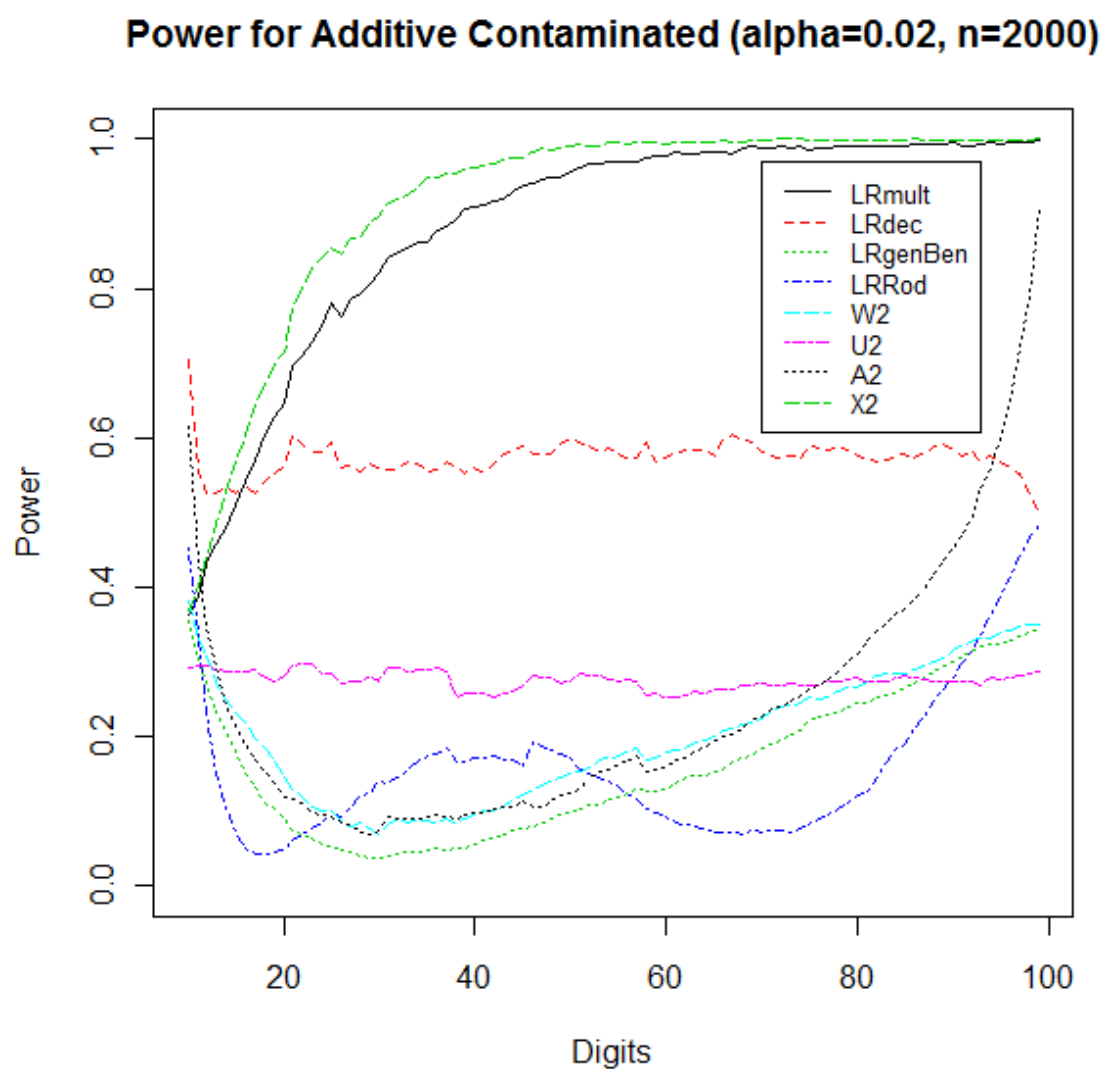


Figure 4.3: Simulated power for  $n = 1000$  samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.06$ ,  $N = 1000$  replications, significance level 0.05.

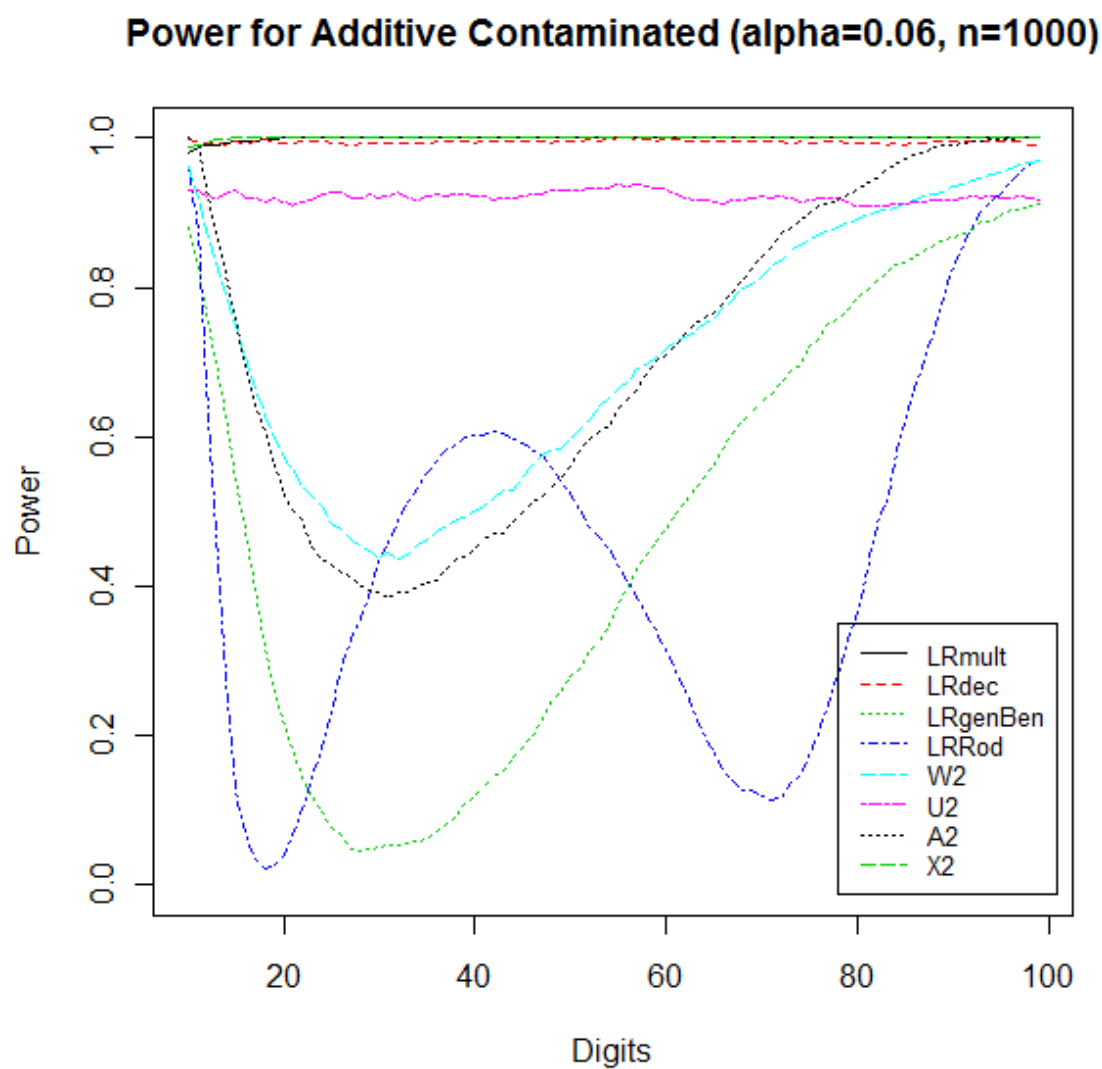


Figure 4.4: Simulated power for  $n = 2000$  samples generated under the contaminated additive Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.06$ ,  $N = 1000$  replications, significance level 0.05.

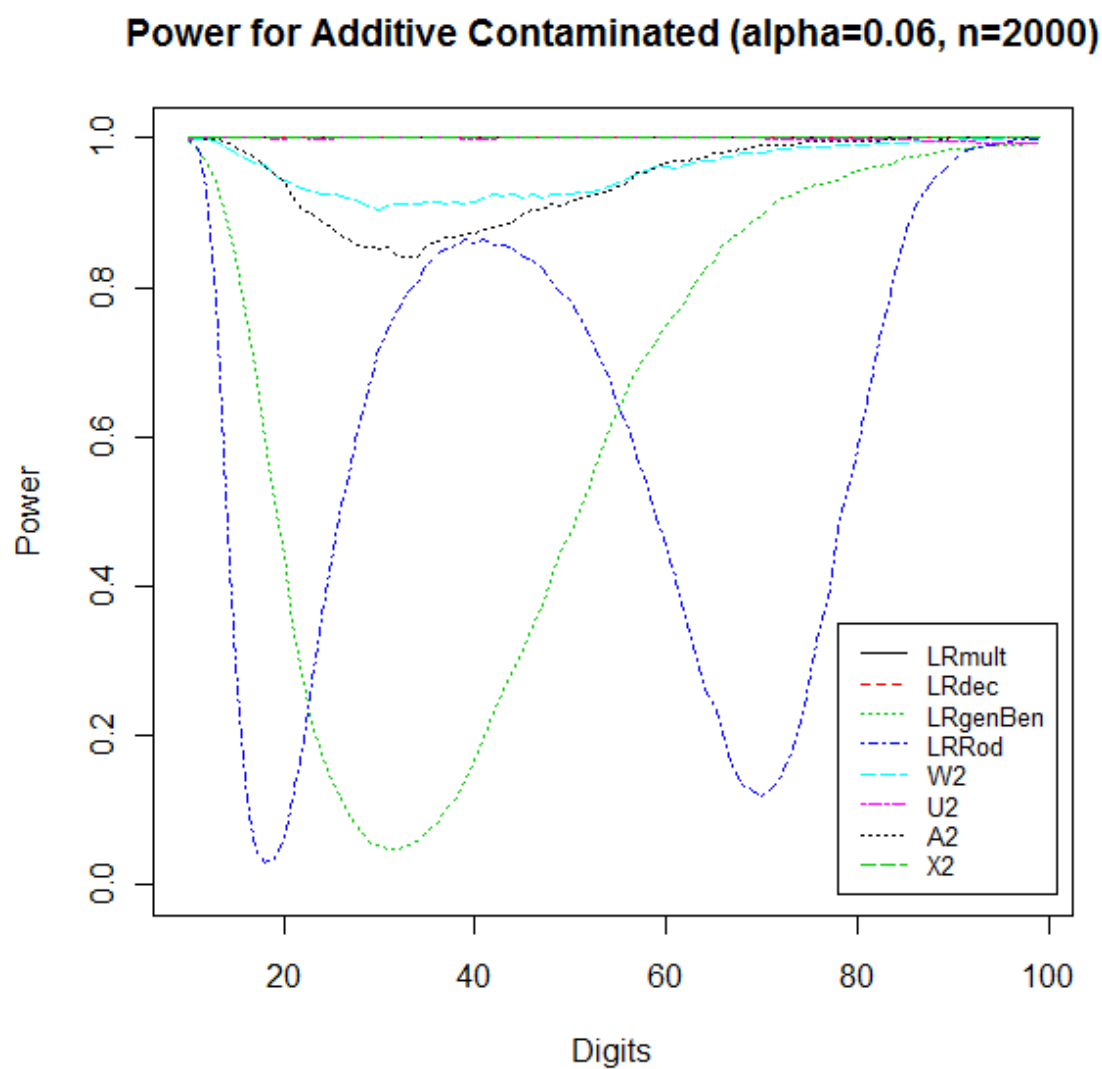


Figure 4.5: Simulated power for  $n = 1000$  samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 1.2$ ,  $N = 1000$  replications, significance level 0.05. Note y-axis scale is not 0 to 1.

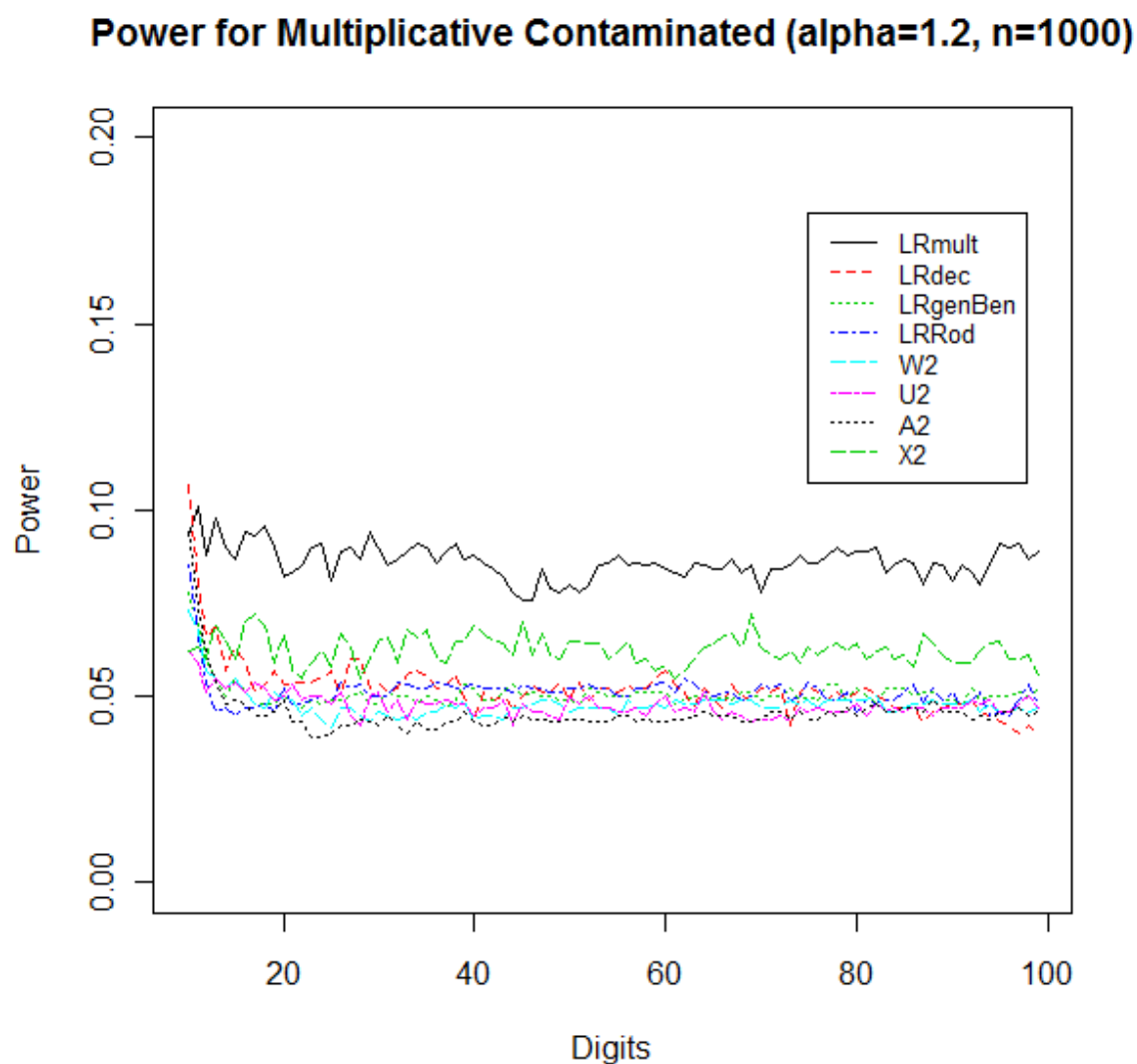




Figure 4.7: Simulated power for  $n = 1000$  samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 1.5$ ,  $N = 1000$  replications, significance level 0.05. Note y-axis scale is not 0 to 1.

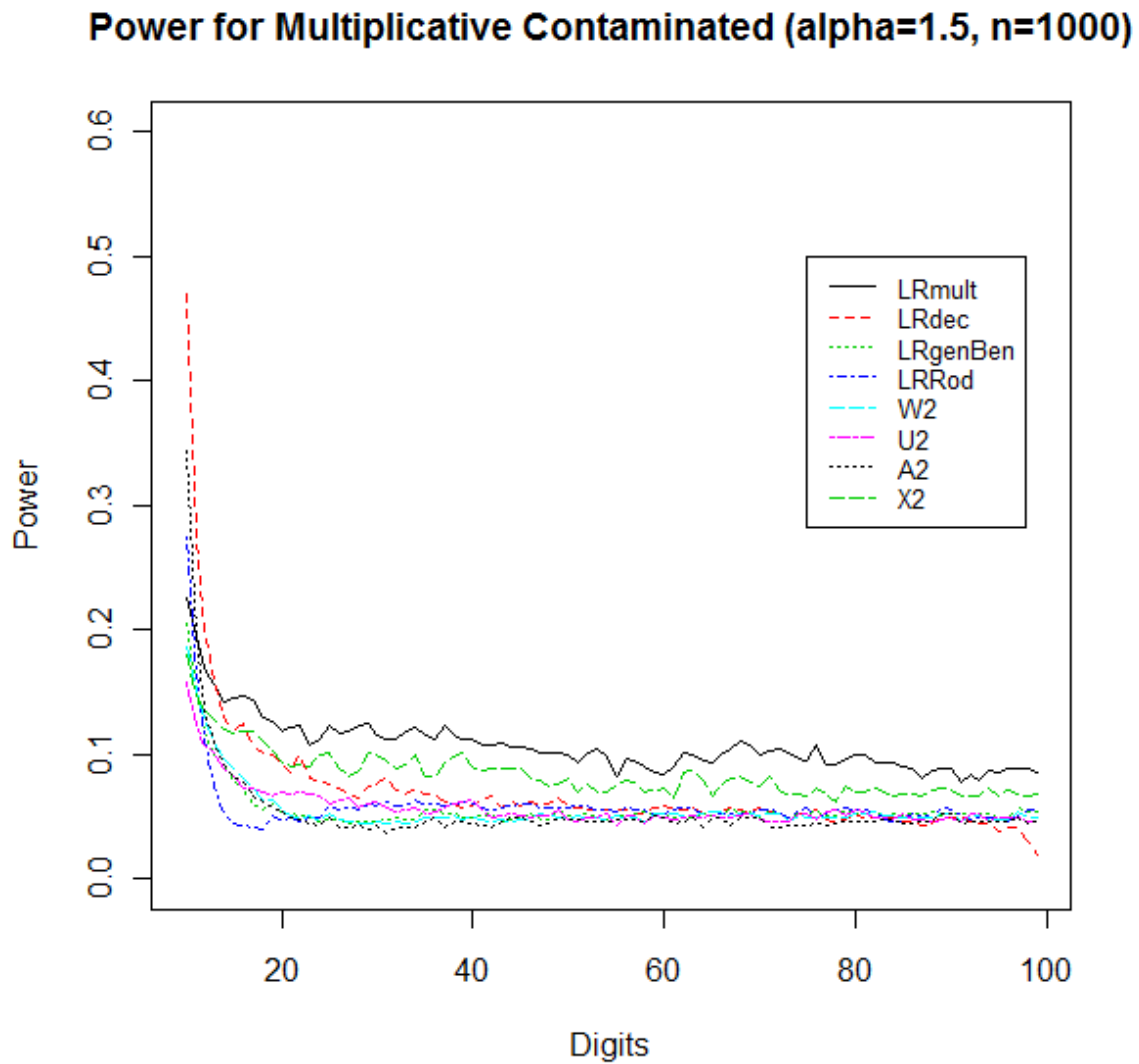




Figure 4.8: Simulated power for  $n = 2000$  samples generated under the contaminated multiplicative Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 1.5$ ,  $N = 1000$  replications, significance level 0.05. Note y-axis scale is not 0 to 1.

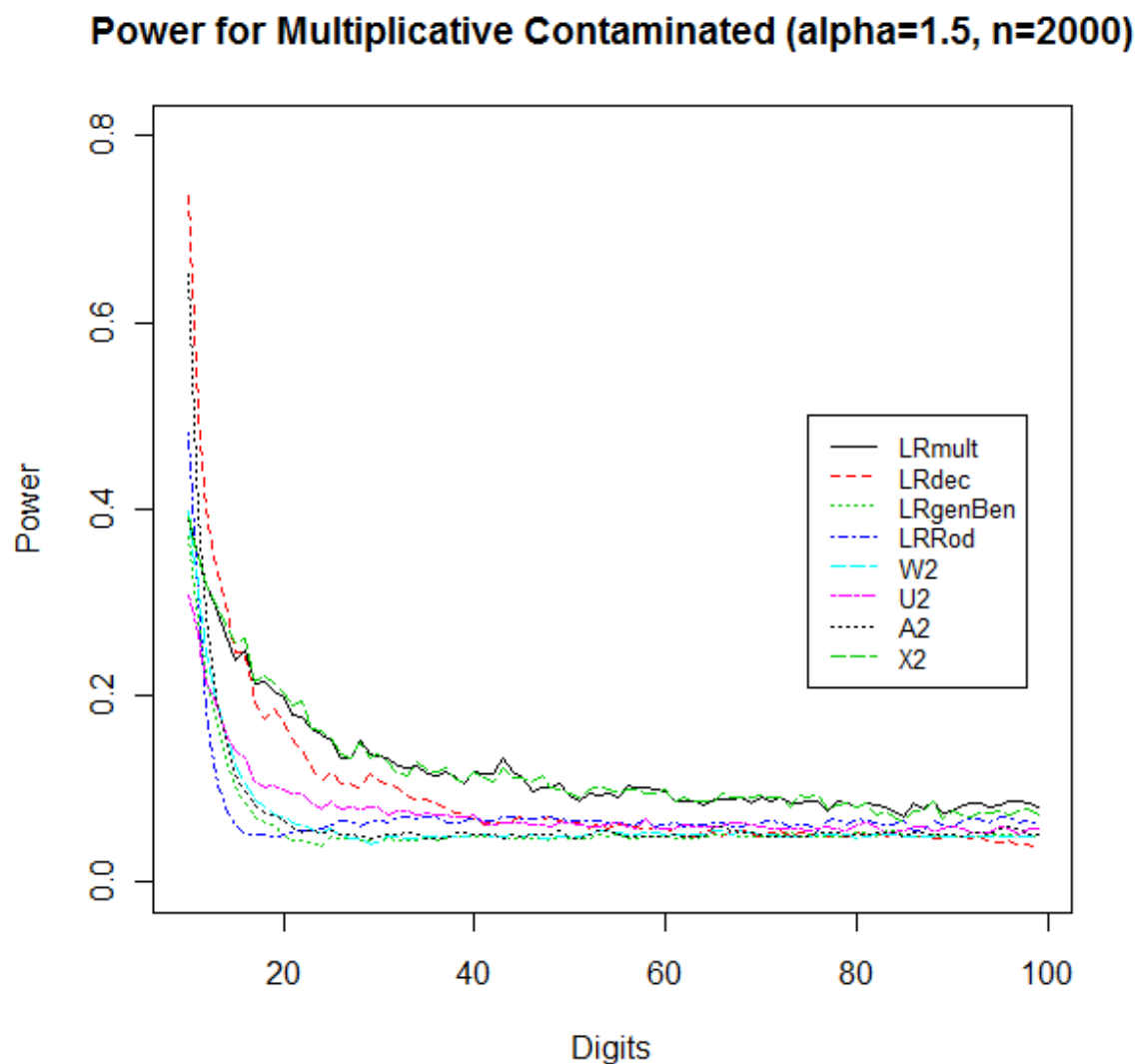


Figure 4.9: Simulated power for  $n = 1000$  samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = -1.0, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1$ ,  $N = 1000$  replications, significance level 0.05.

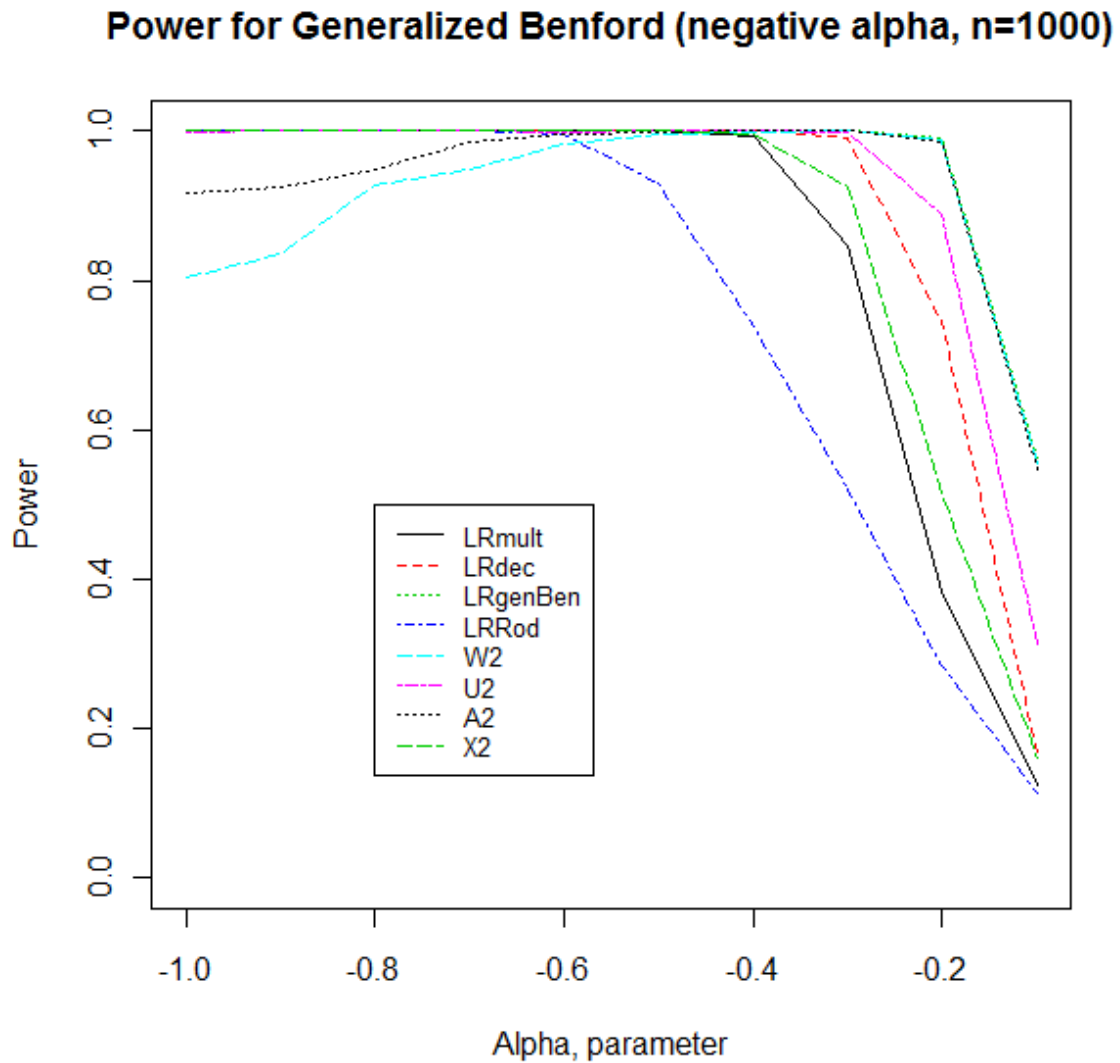


Figure 4.10: Simulated power for  $n = 2000$  samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = -1.0, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1$ ,  $N = 1000$  replications, significance level 0.05.

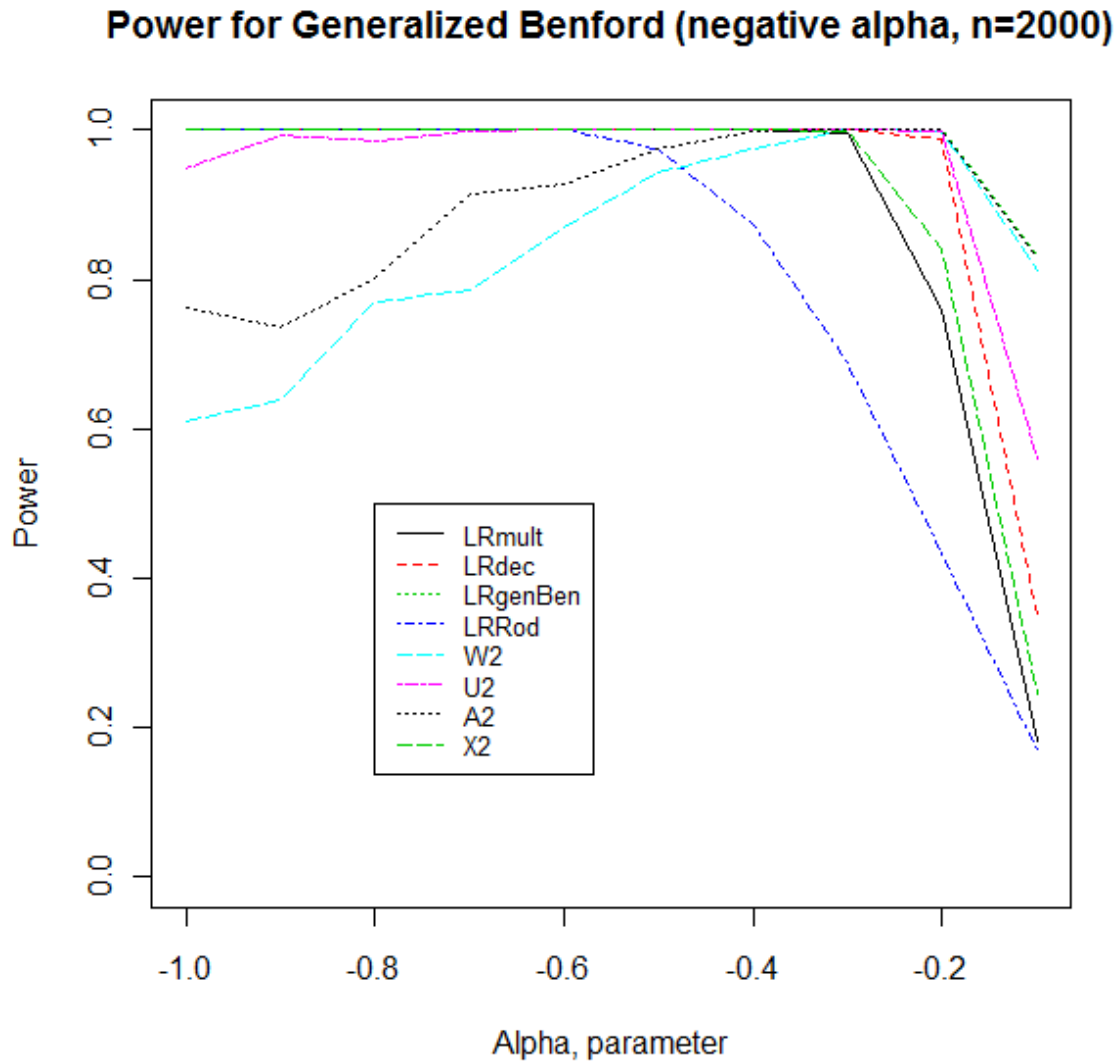


Figure 4.11: Simulated power for  $n = 1000$  samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0$ ,  $N = 1000$  replications, significance level 0.05.

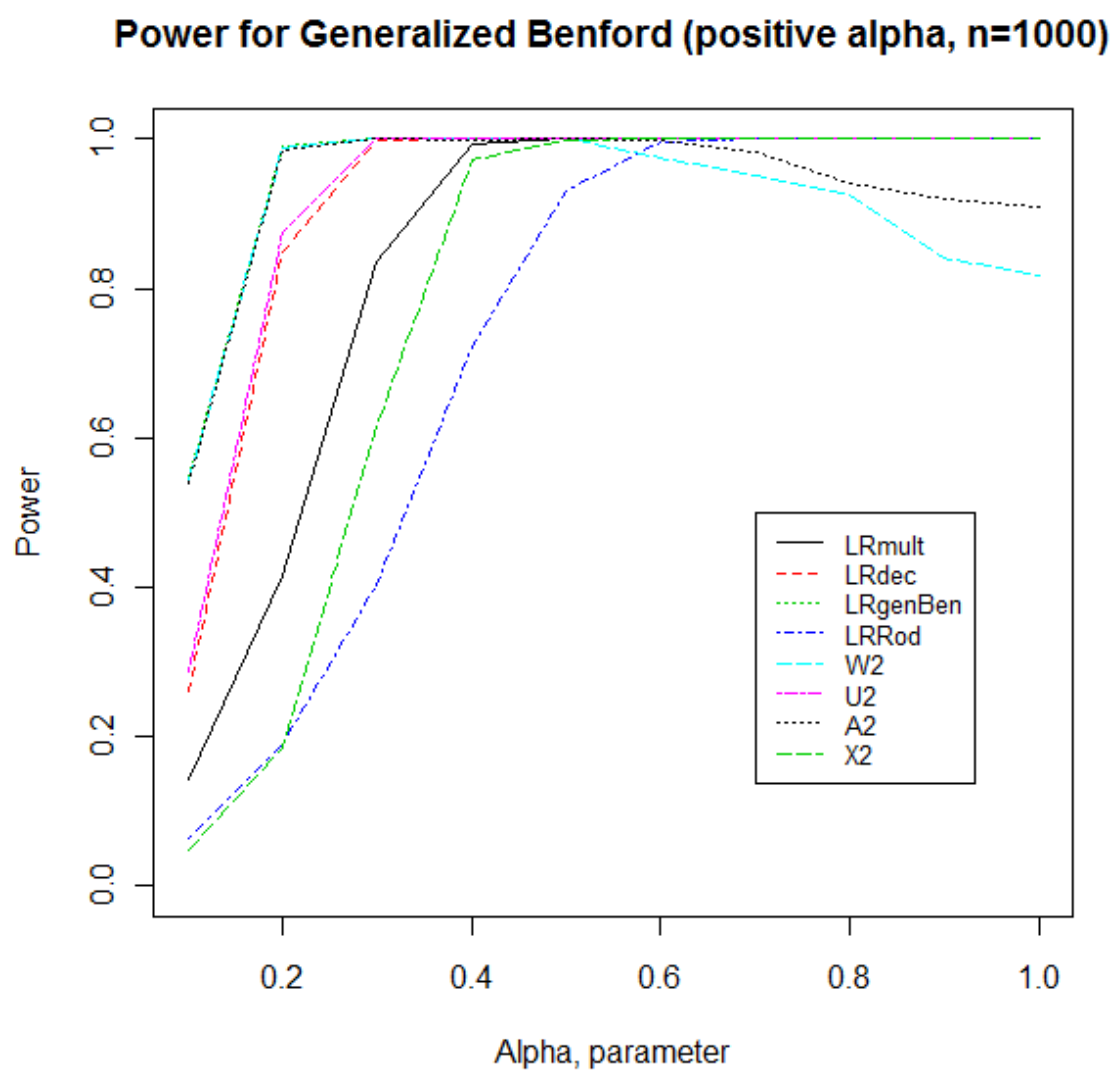


Figure 4.12: Simulated power for  $n = 2000$  samples generated under Generalized Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0$ ,  $N = 1000$  replications, significance level 0.05.

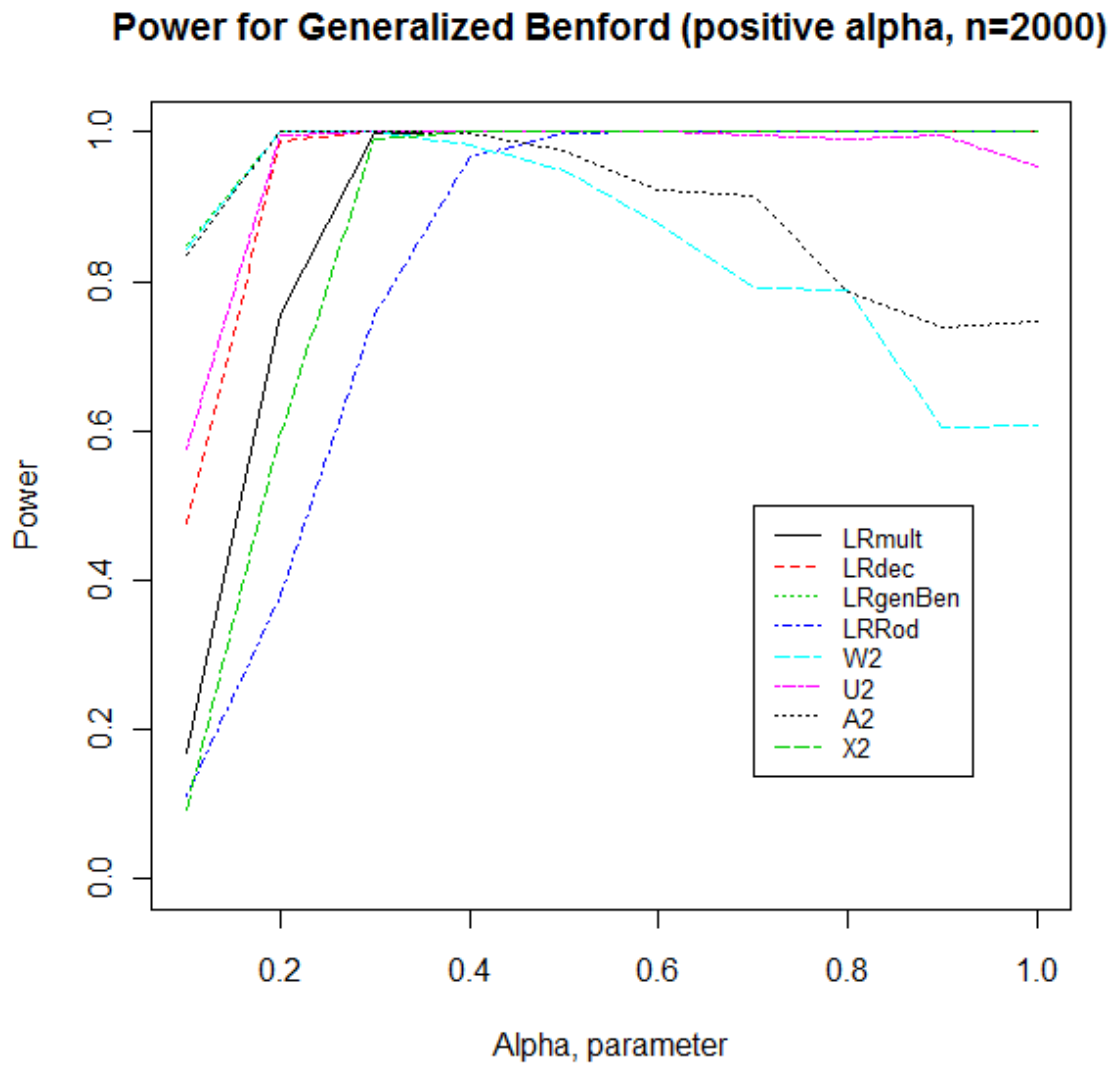


Figure 4.13: Simulated power for  $n = 1000$  samples generated under Mixed Uniform/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ,  $N = 1000$  replications, significance level 0.05.

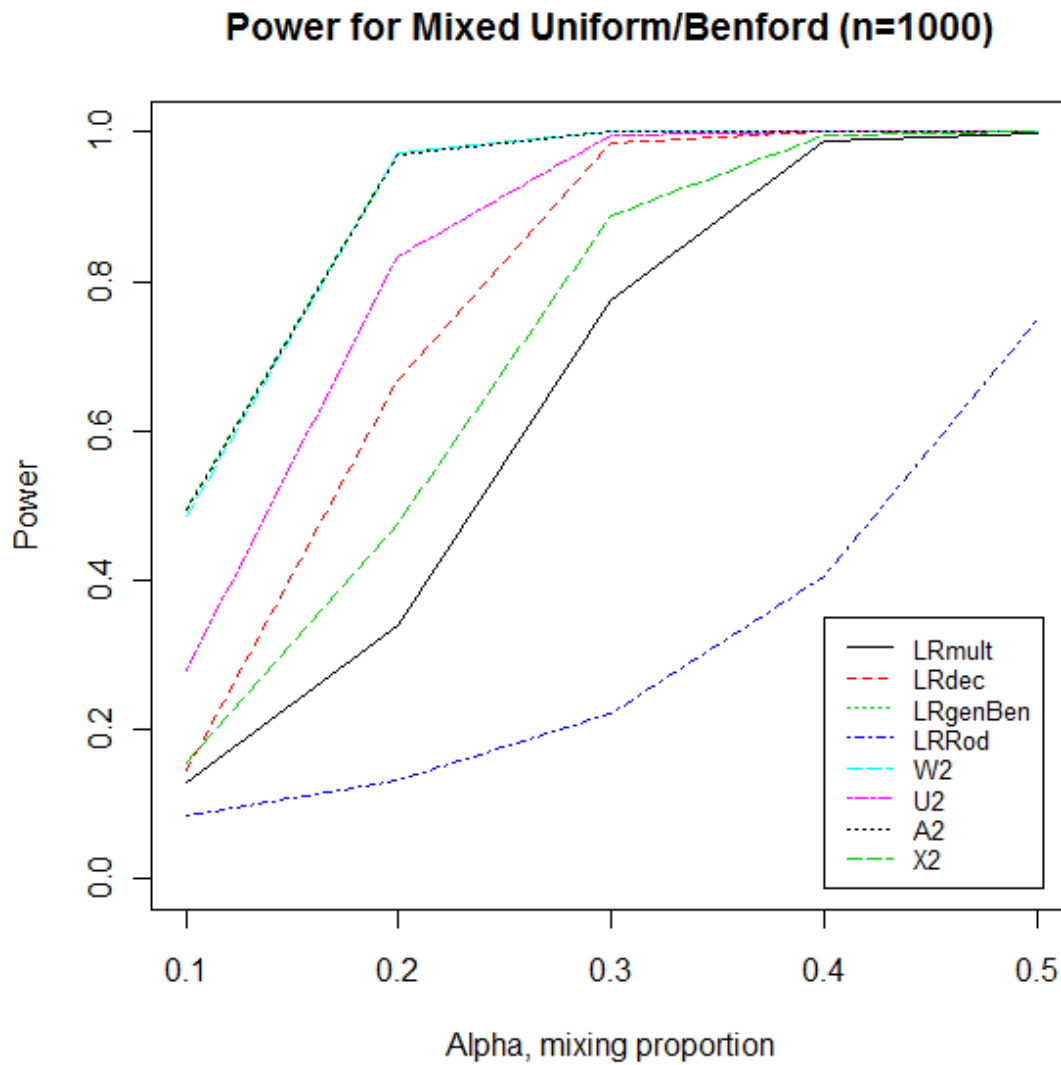


Figure 4.14: Simulated power for  $n = 2000$  samples generated under Mixed Uniform/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ,  $N = 1000$  replications, significance level 0.05.

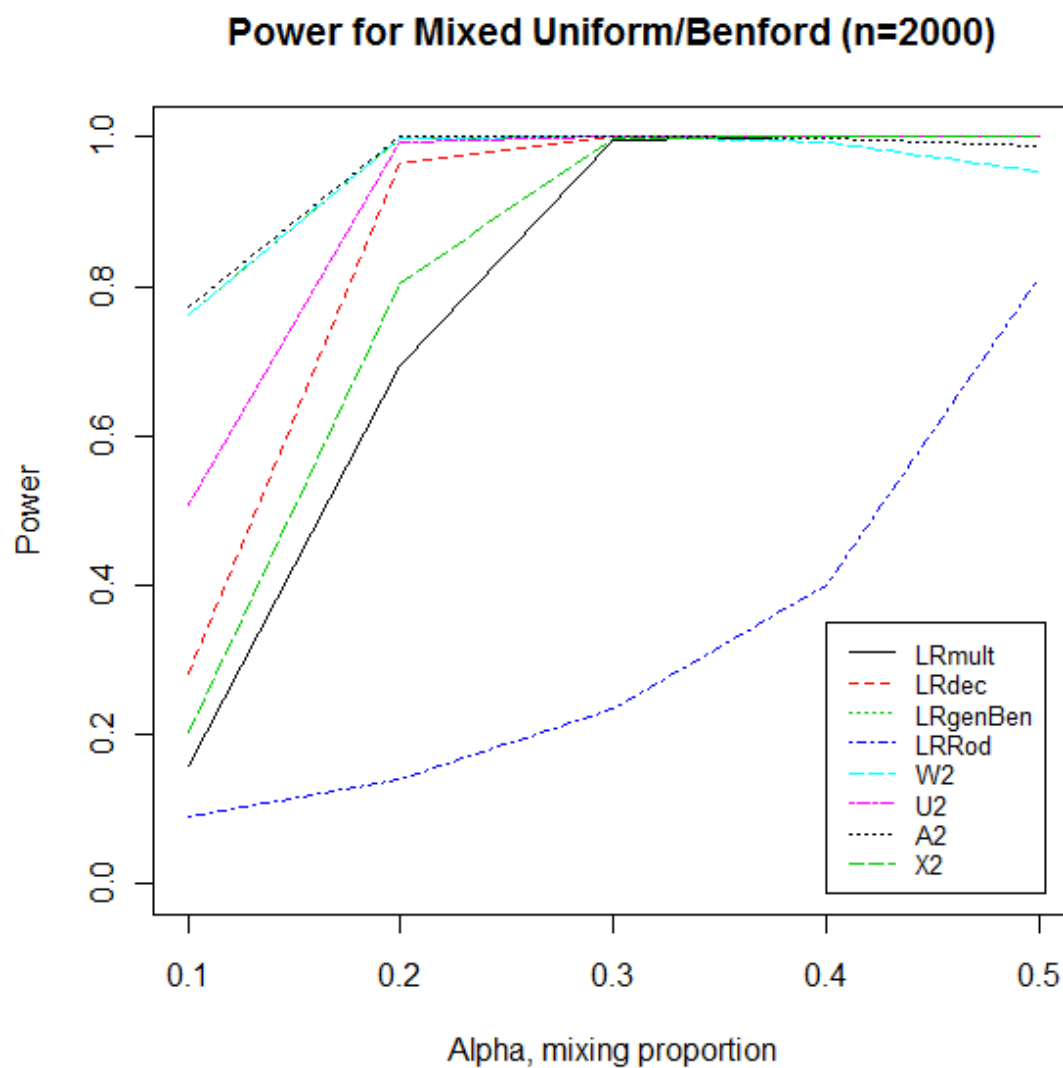


Figure 4.15: Simulated power for  $n = 1000$  samples generated under Mixed Hill/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ,  $N = 1000$  replications, significance level 0.05.

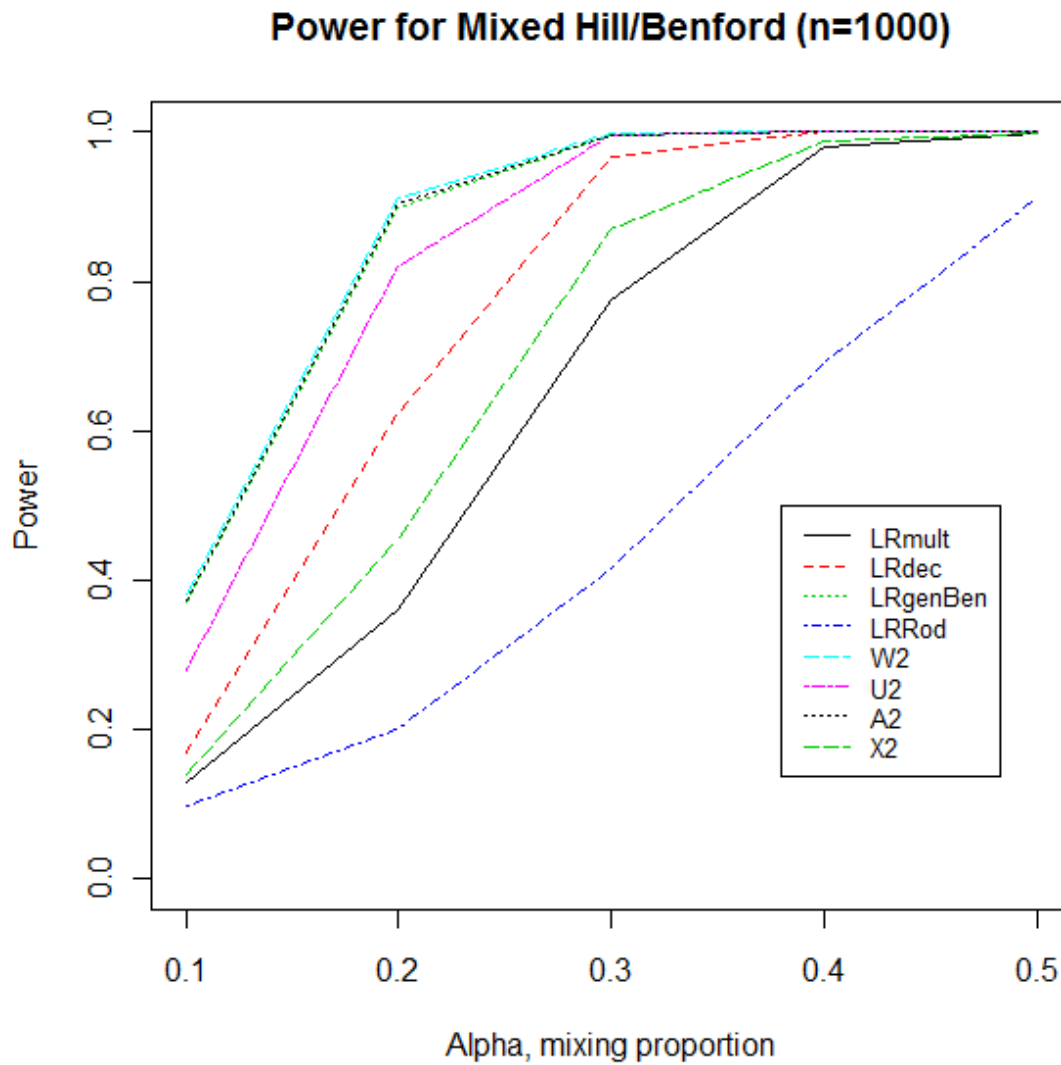
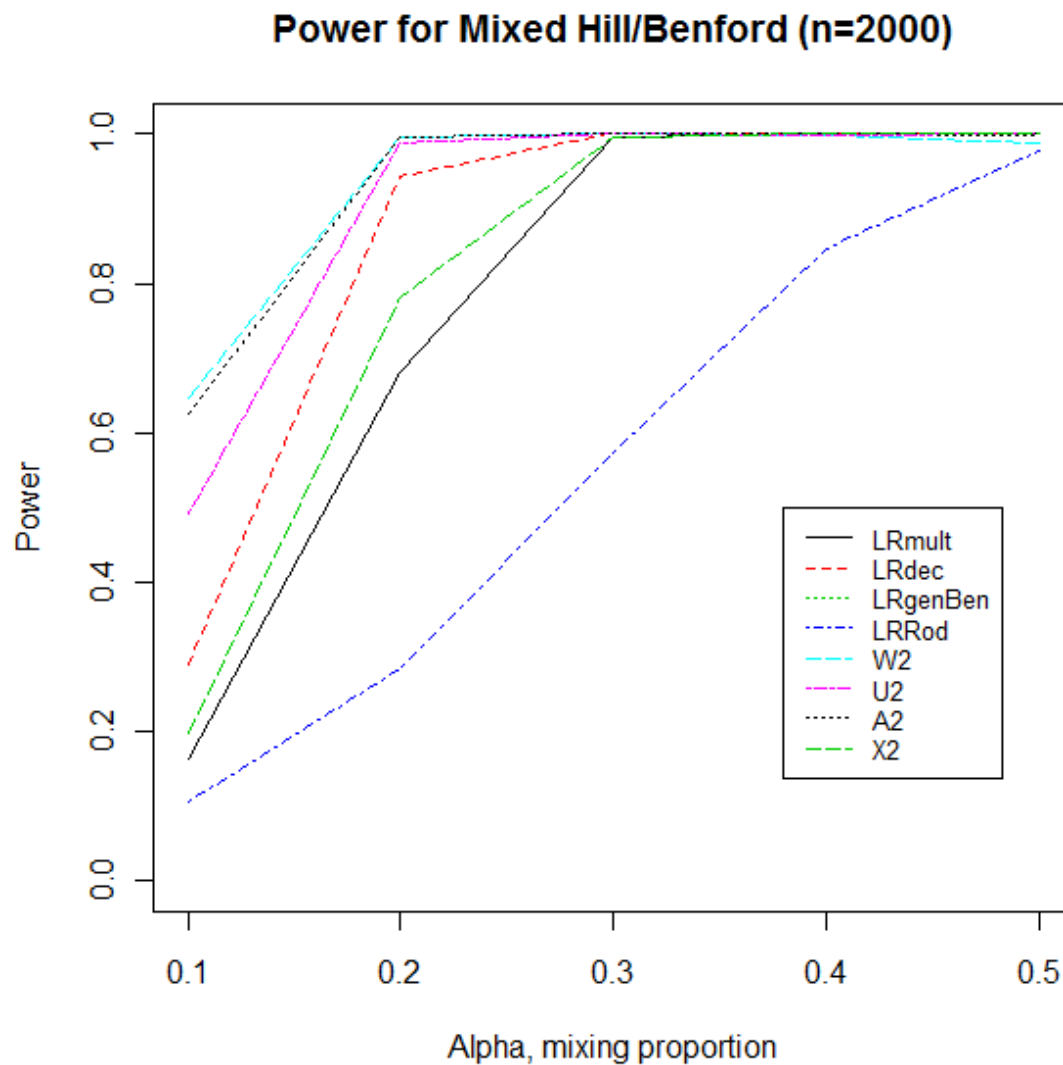




Figure 4.16: Simulated power for  $n = 2000$  samples generated under Mixed Hill/Benford distribution for statistics LR-mult, LR-dec, LR-genBen, LR-Rod,  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ,  $N = 1000$  replications, significance level 0.05.



From Figure 4.17 and 4.18, we show the approximate and simulated power for the Pearson's  $\chi^2$  when  $n = 1000$  and  $n = 2000$  at which sample data were generated using the contaminated additive Benford and contaminated multiplicative Benford distributions respectively. Power is plotted on y-axis and the contaminated parameter is on x-axis. For example, the plot 'digit 10' shows the approximate and simulated power when digit 10 is contaminated. We calculated the approximate 95% confidence intervals (vertical bars in the graphs) for the simulated power through the normal approximation to the binomial. Again, we only graph part of the results here because of huge volume of figures; however, the rest of the plots can be found in appendix B.

From Figure 4.19, 4.20 and Figure 4.21, 4.22 show the approximate and simulated power for two CVM statistics  $W_d^2$  and  $U_d^2$  when data were generated using the Uniform/Benford and Hill/Benford mixture distributions respectively. For example,  $\alpha = 0.1$  means 10% of the distribution comes from Uniform (Hill) distribution and 90% belongs to Benford distribution. For the graphs, we see that the approximate power agrees with the simulation results very well.

Figure 4.17: Comparison of approximate and simulated power for the contaminated additive Benford distribution ( $\alpha = 0.02, 0.06$ ) with digits 10 to 18,  $n = 1000$  (black solid line), 2000 (red dashed line), significance level 0.05.

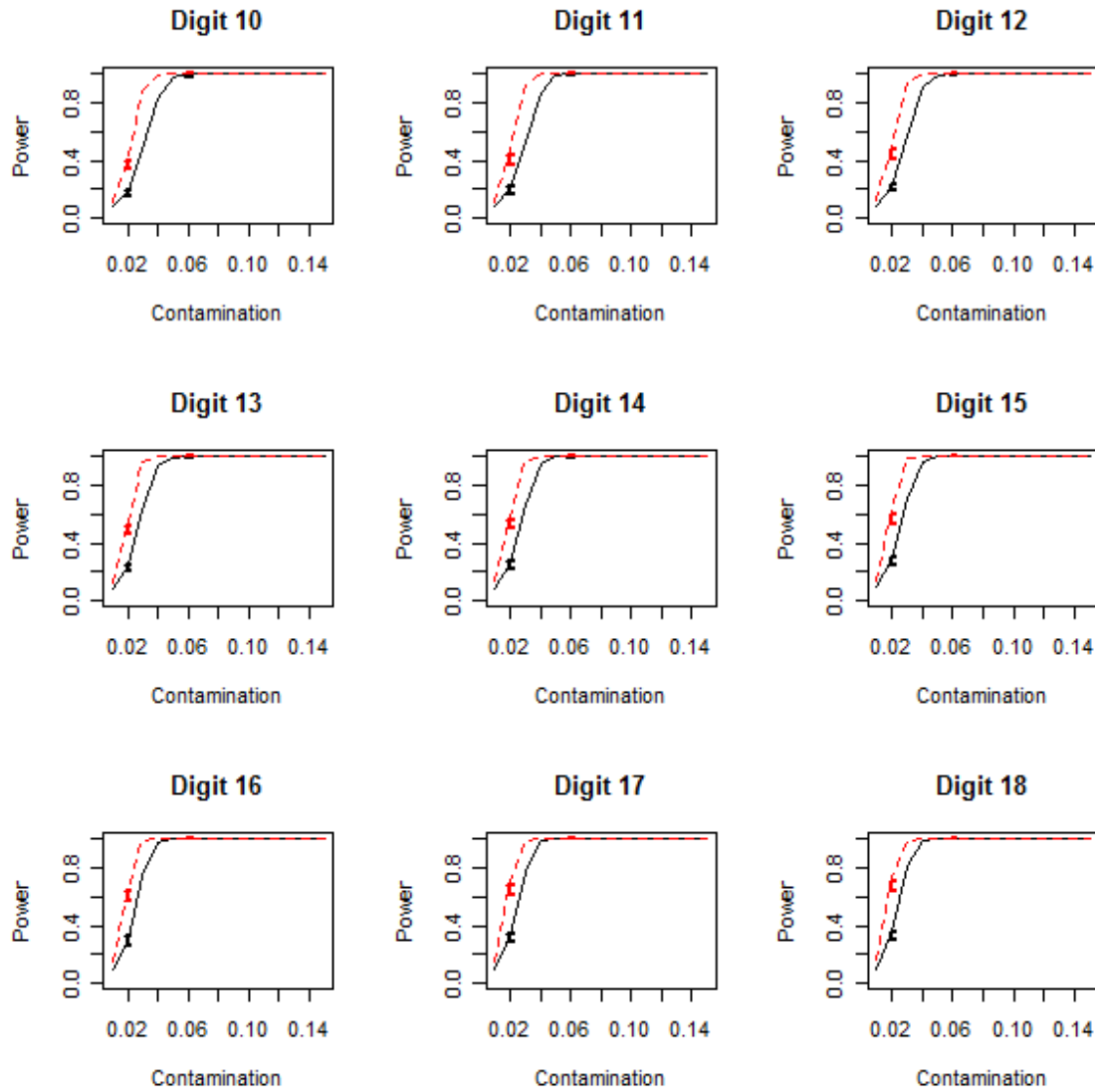


Figure 4.18: Comparison of approximate and simulated power for the contaminated multiplicative Benford distribution ( $\alpha = 1.2, 1.5$ ) with digits 10 to 18,  $n = 1000$  (black solid line), 2000 (red dashed line), significance level 0.05.

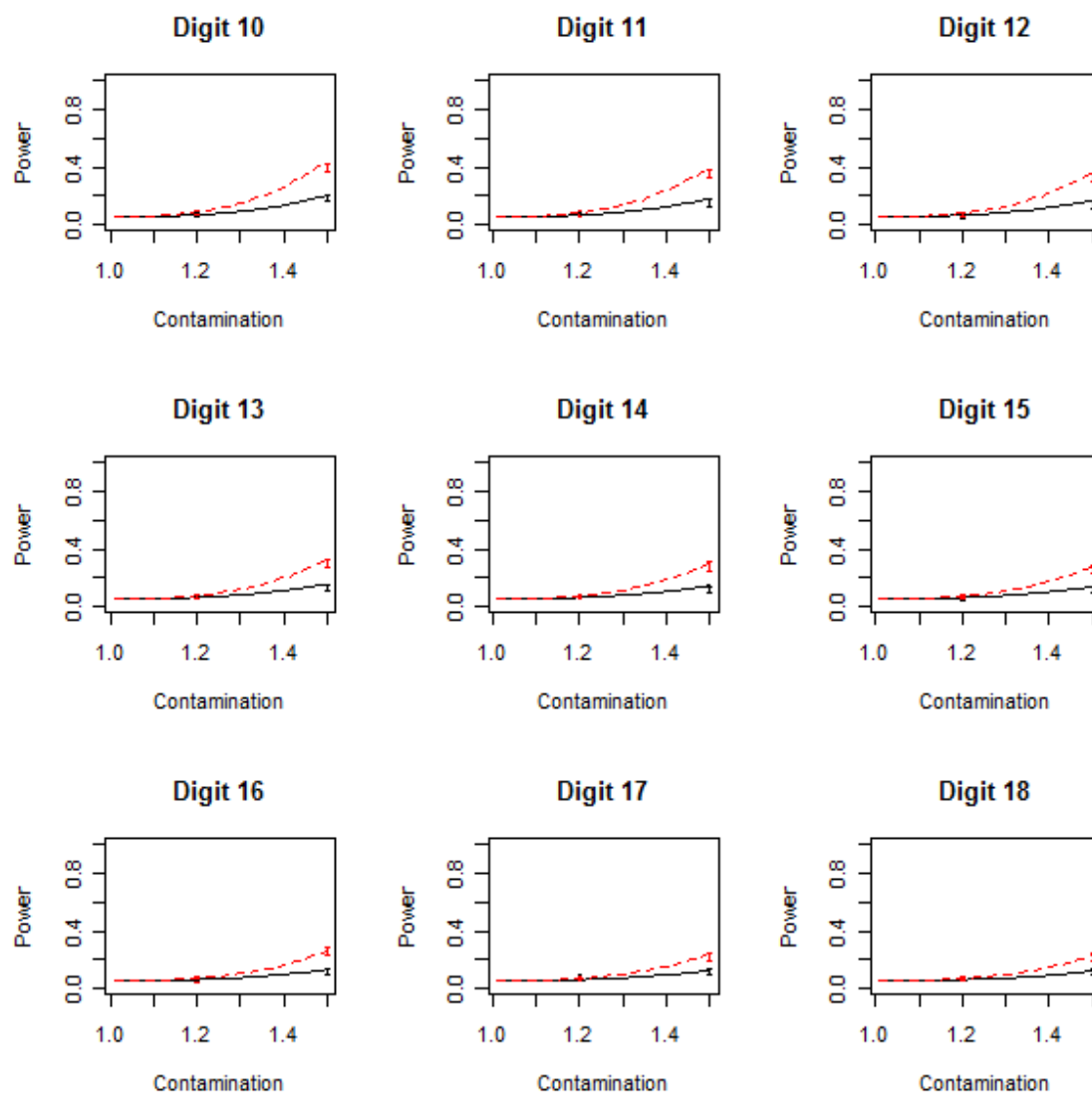


Figure 4.19: Comparison approximate and simulated power for  $n = 1000$  samples generated under Uniform/Benford mixture distribution for two CVM statistics,  $W_d^2$  and  $A_d^2$ , significance level 0.05.

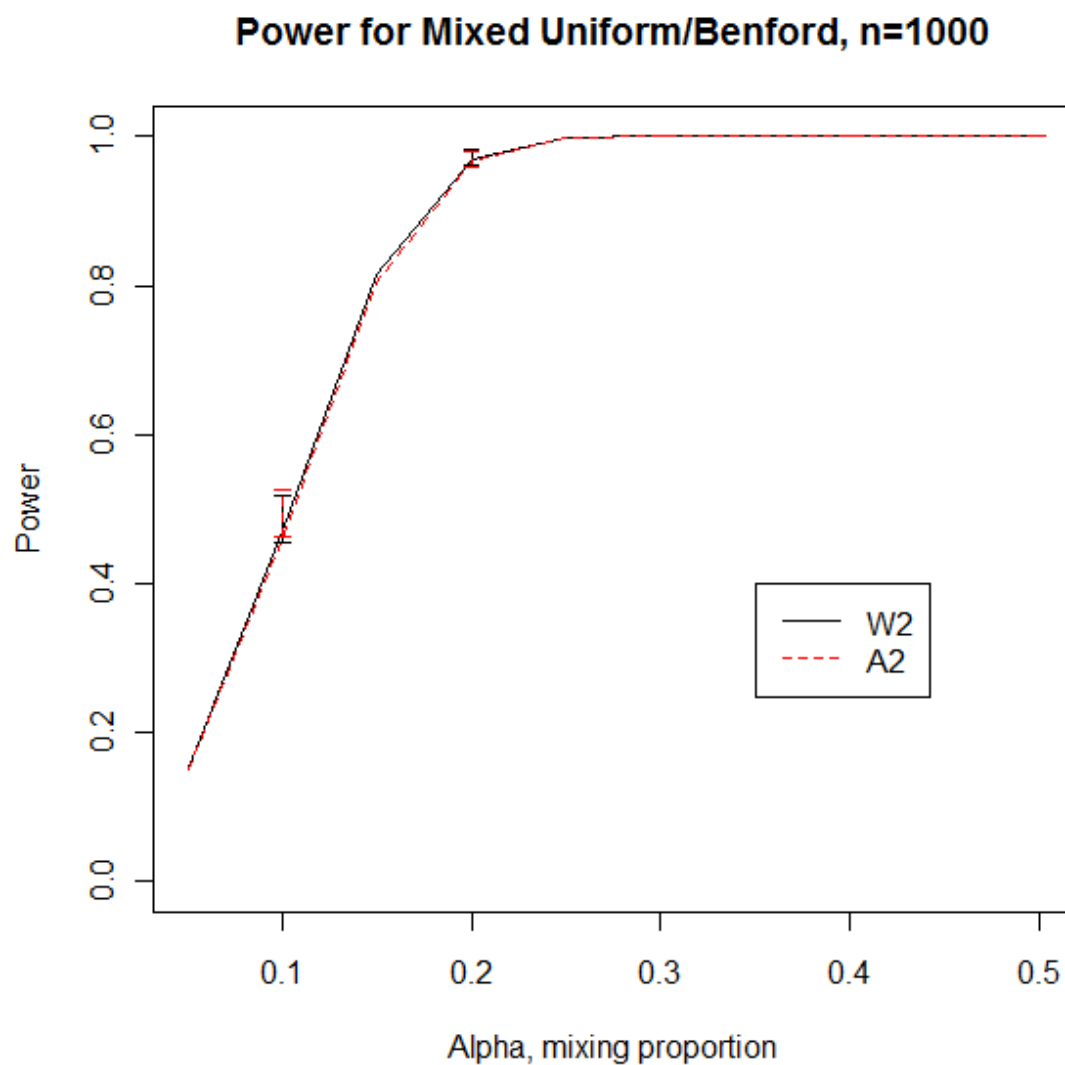


Figure 4.20: Comparison approximate and simulated power for  $n = 2000$  samples generated under Uniform/Benford mixture distribution for two CVM statistics,  $W_d^2$  and  $A_d^2$ , significance level 0.05.

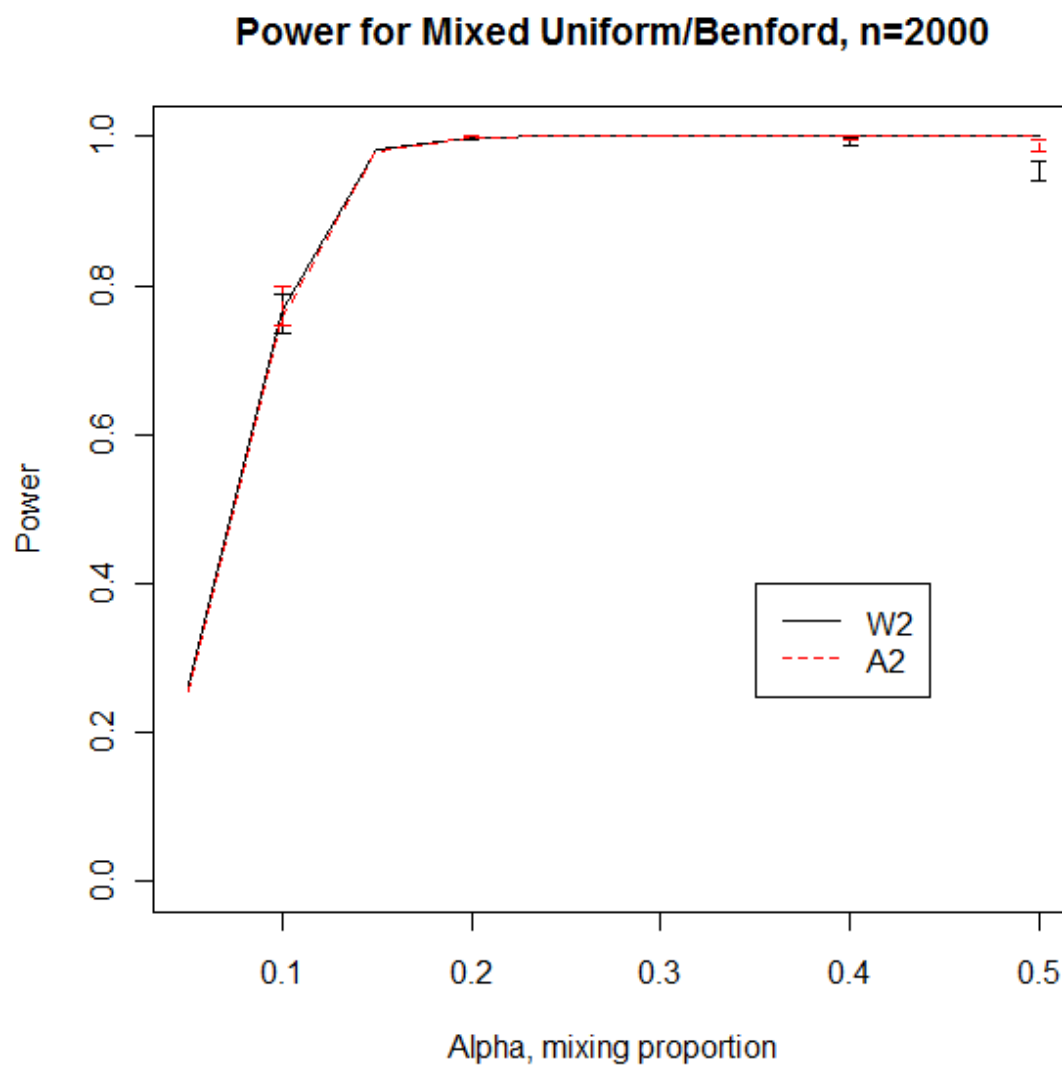


Figure 4.21: Comparison approximate and simulated power for  $n = 1000$  samples generated under Hill/Benford mixture distribution for two CVM statistics,  $W_d^2$  and  $A_d^2$ , significance level 0.05.

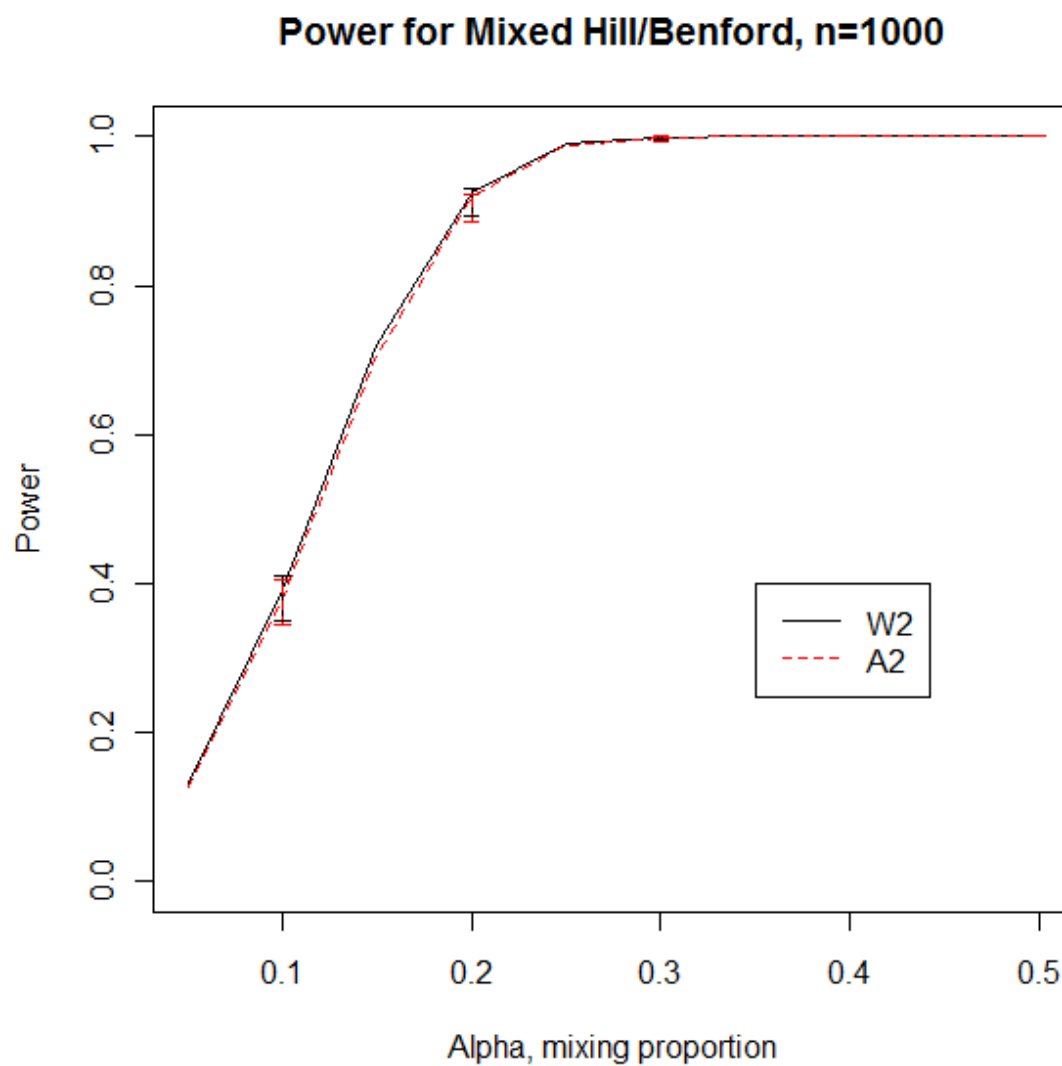
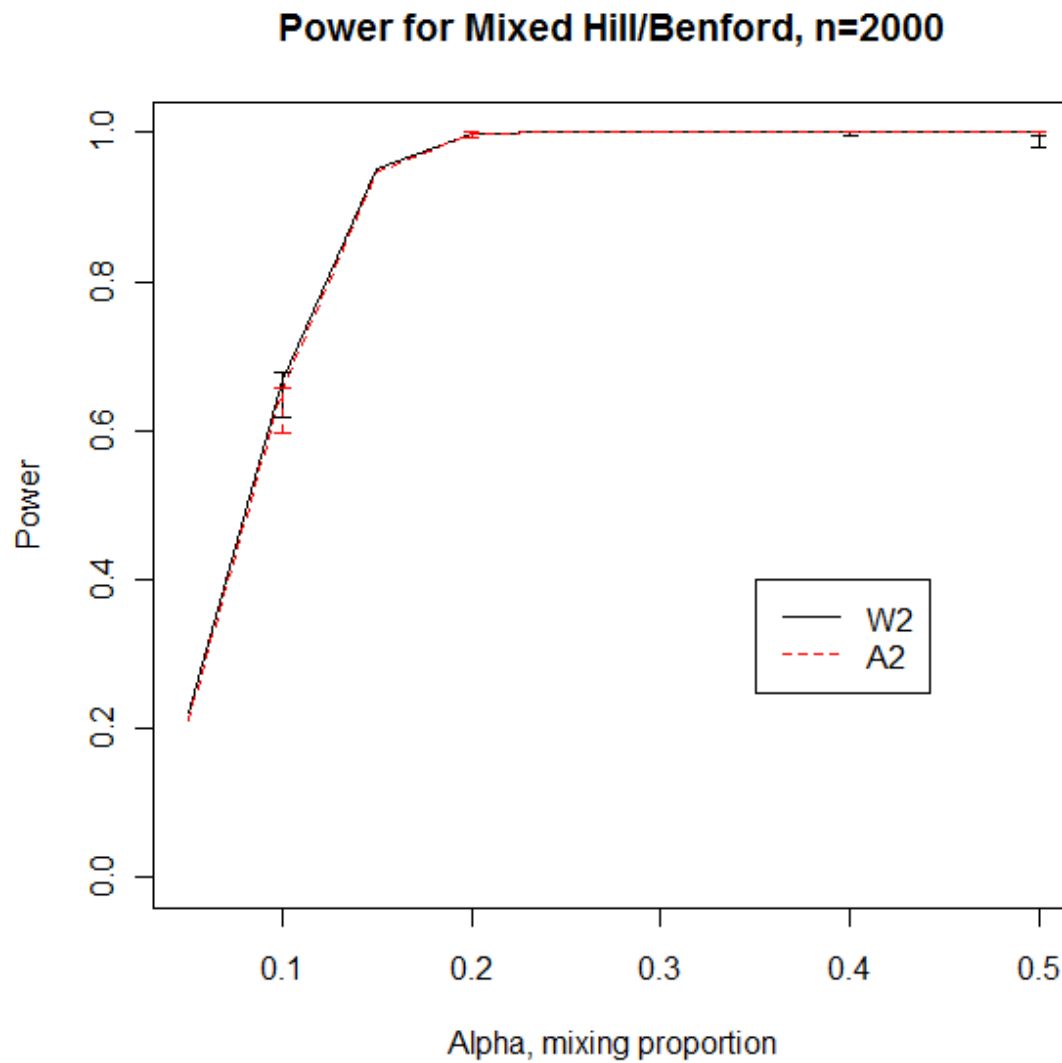


Figure 4.22: Comparison approximate and simulated power for  $n = 2000$  samples generated under Hill/Benford mixture distribution for two CVM statistics,  $W_d^2$  and  $A_d^2$ , significance level 0.05.





From Figure 4.23 to Figure 4.26, we show the relationship between the sample size and power. Here, we plot the approximate power curves for four test statistics  $W_d^2$ ,  $U_d^2$ ,  $A_d^2$ , and  $\chi^2$  with  $n = 100, 1000, 2000, 3000, 4000$ , and  $5000$  when the data generating distribution is the Hill/Benford mixture distribution. From this graph, we see that in order to get reasonable power for testing for Benford's Law, we need to have large sample sizes.

Figure 4.23: Approximate power for  $W_d^2$  for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05.

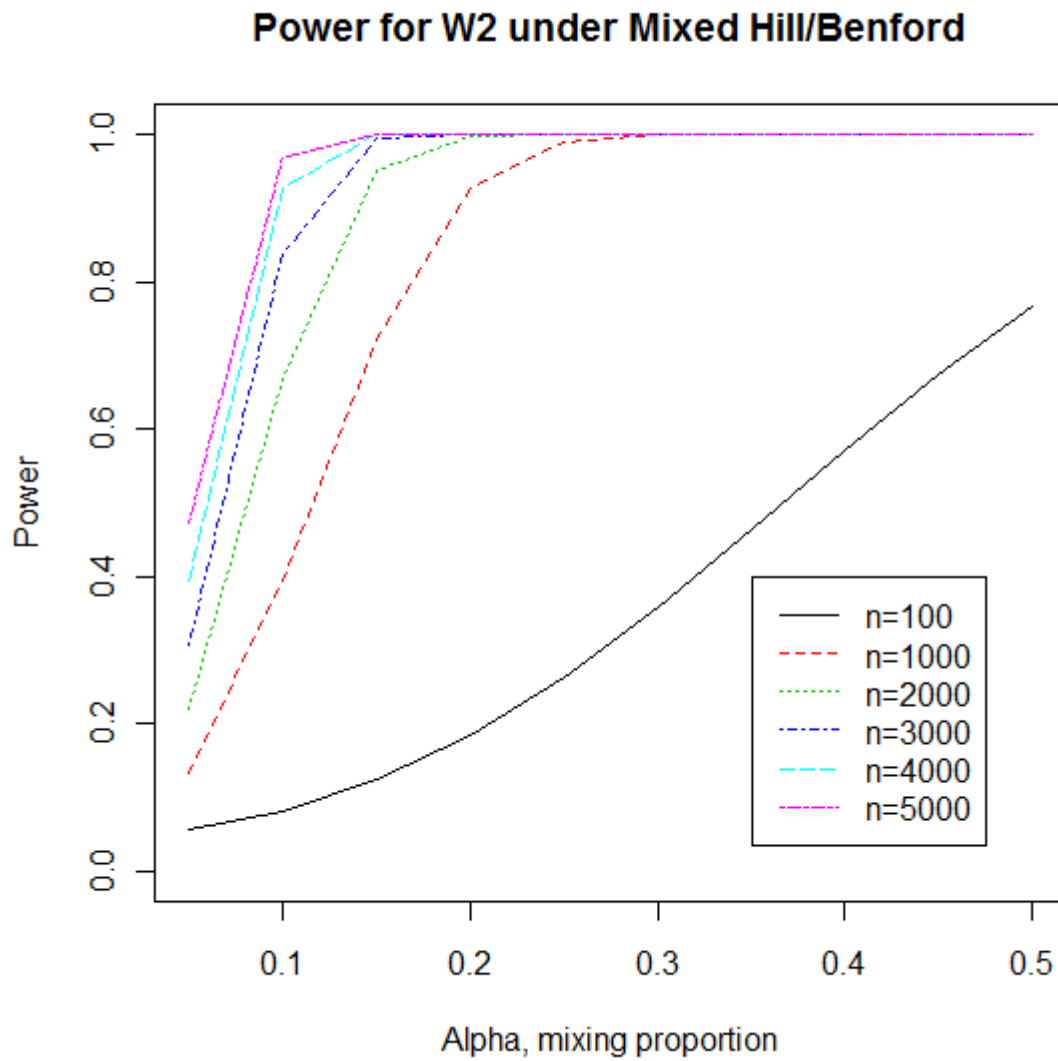


Figure 4.24: Approximate power for  $U_d^2$  for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05.

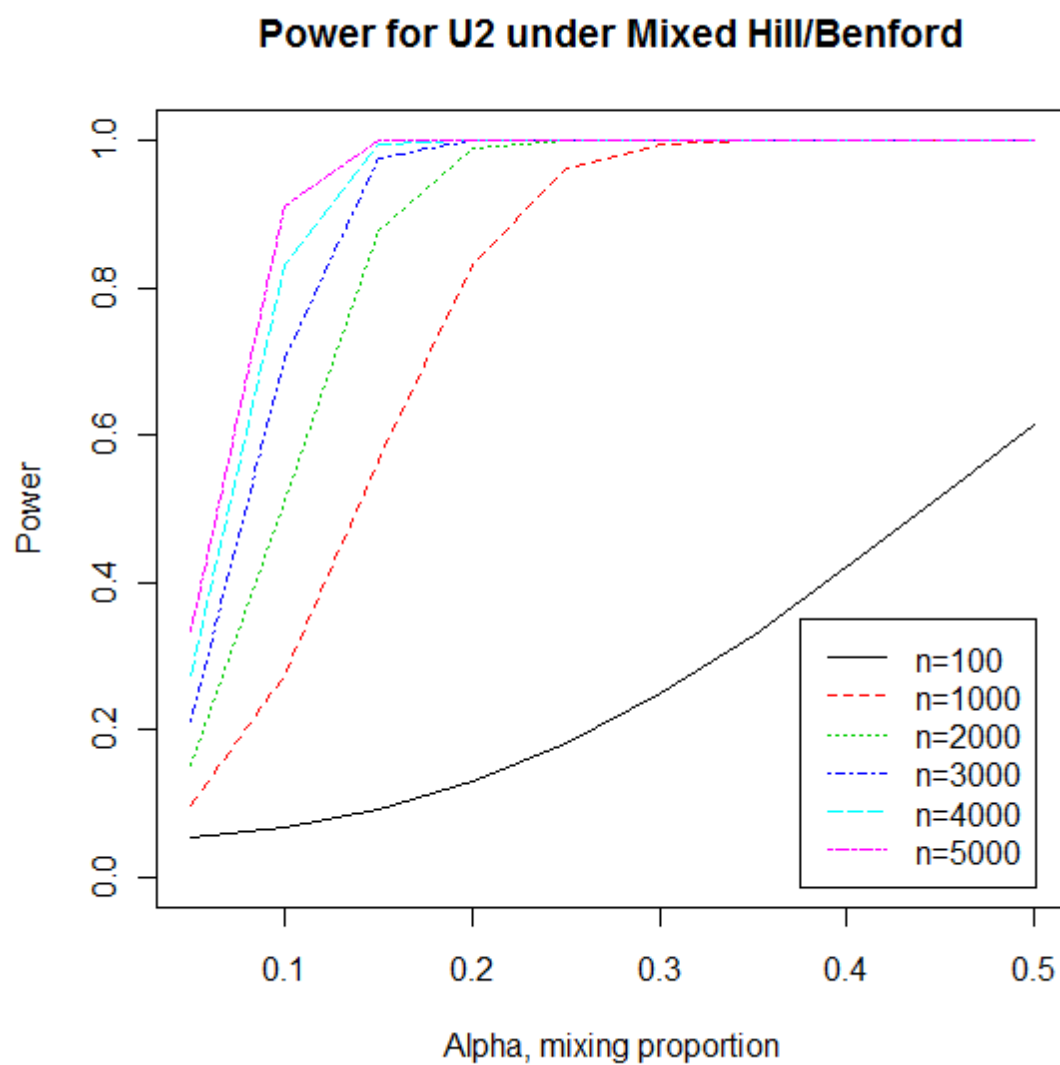


Figure 4.25: Approximate power for  $A_d^2$  for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05.

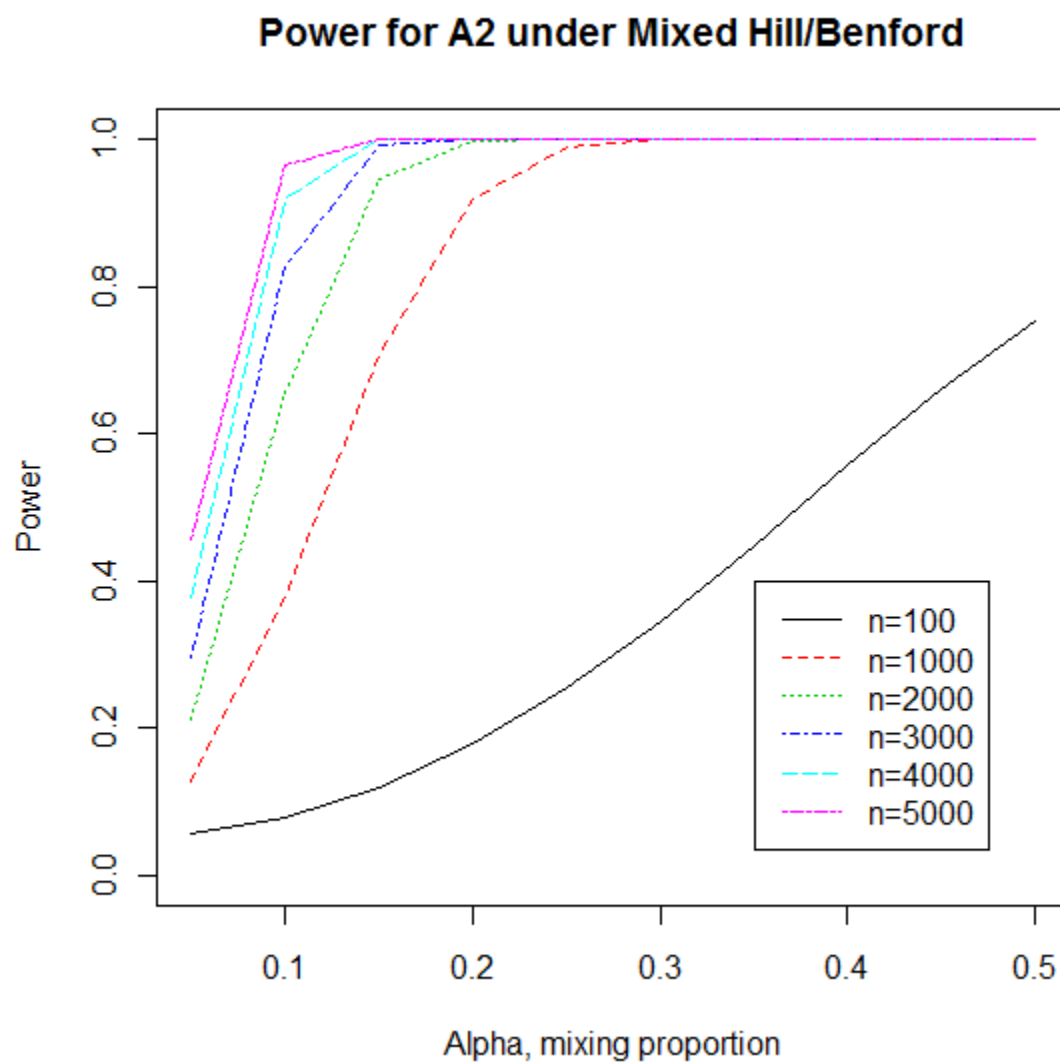


Figure 4.26: Approximate power for  $\chi^2$  for varying sample sizes generated under Hill/Benford mixture distribution, significance level 0.05.

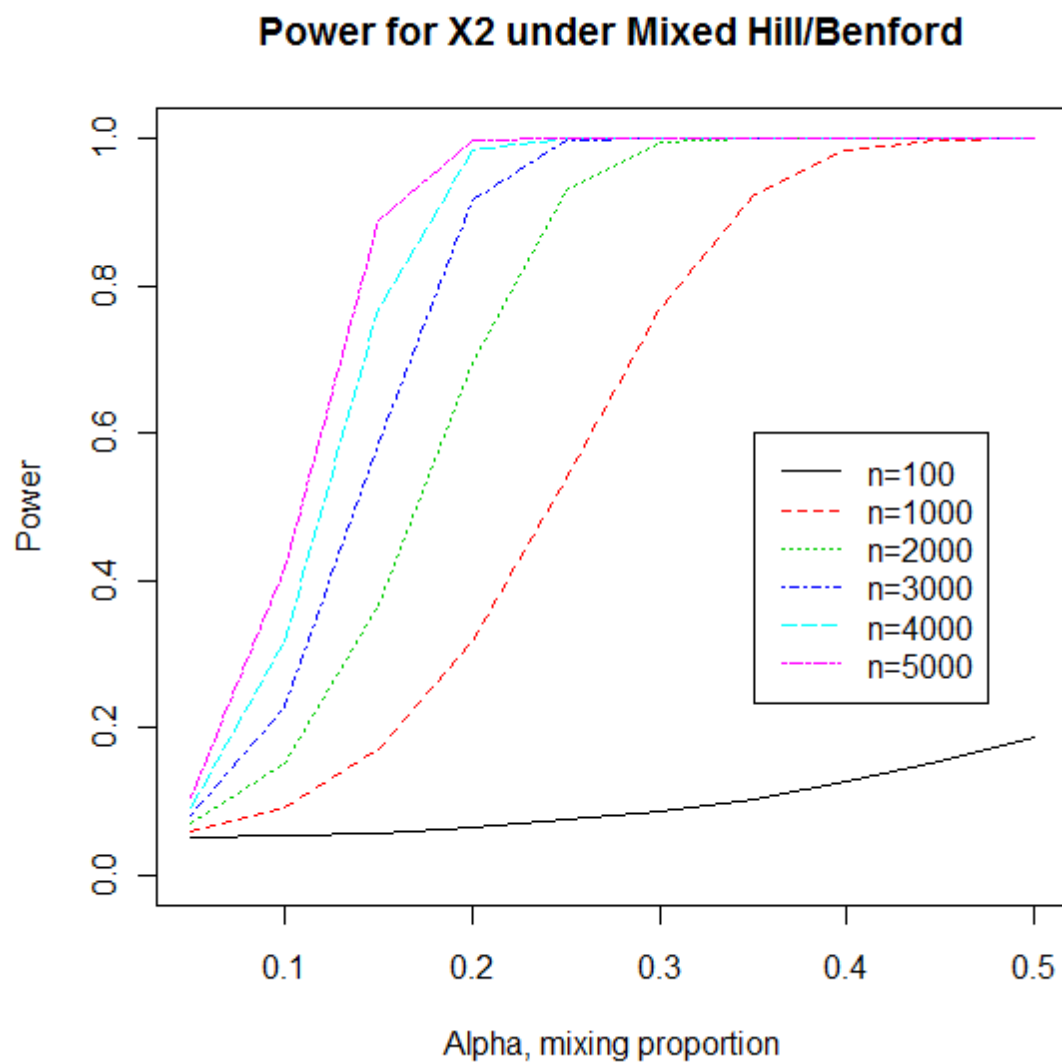


Table 4.14: Coverage proportions for 90%, 95% and 99% simultaneous confidence intervals for data generated using the Benford distribution

<i>Benford Distribution</i>	<b>n = 1000</b>			<b>n = 2000</b>		
	90% CI	95% CI	99% CI	90% CI	95% CI	99% CI
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.837	0.901	0.966	0.862	0.919	0.975
<b>Bailey</b>	0.866	0.923	0.981	0.888	0.940	0.983
<b>Angular</b>						
<b>Bailey</b>	0.865	0.921	0.980	0.888	0.940	0.983
<b>Square</b>						
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.889	0.947	0.993	0.897	0.961	0.993
<b>Univariate</b>	0.000	0.000	0.034	0.000	0.002	0.122
<b>Binomial</b>						

We evaluate the quality of 6 simultaneous confidence intervals – Quesenberry, Goodman, Bailey Angular, Bailey Square, Fitzpatrick and Sison when data are simulated using Benford’s Law. For each sample size,  $n = 1000$  and  $n = 2000$ , we generated  $N = 1000$  datasets and computed simultaneous confidence intervals for each dataset. We recorded the number of datasets for which Benford probabilities actually fall within the intervals. Table 4.14 contains the coverage proportions for  $N = 1000$  replications of 90%, 95%, and 99% simultaneous confidence intervals for data generated using the Benford distribution.

From these results, we see that the coverage probabilities of the Univariate Binomial confidence intervals are very small (0.000 – 0.122) under Benford distribution. On the other hand, the coverage probabilities of the Quesenberry and Fitzpatrick confidence intervals are 1.000 which is too conservative. The remaining 4 confidence intervals; Goodman, Bailey Angular, Bailey Square, and Sison have better coverage probabilities under Benford distribution. Within those, the Sison simultaneous confidence intervals perform the best, with coverage probabilities close to the nominal levels in all cases.

Since there are so many results, we summarize some of them in Tables 4.15 through 4.31 which show the proportions for  $N = 1000$  replications of simultaneous confidence

intervals for data generated from the Uniform distribution, contaminated additive Benford distribution, contaminated multiplicative Benford distribution, Generalized Benford distribution, Uniform/Benford mixture distribution and Hill/Benford mixture distribution. Since Univariate Binomial confidence intervals perform extremely poorly for simulated Benford data, we do not consider it seriously however the statistics are included for completeness. Also, we include the rest of the summaries in the appendix B.

Goodman, Bailey Angular, Bailey Square, and Sison perform very well with multinomial frequencies generated under the Uniform distribution; however, Quesenberry performs badly because the coverage probabilities are 1.000. The Bailey Angular and Bailey Square have similar performance over all of the distributions; they have small coverage probabilities under the contaminated additive Benford distribution.

Quesenberry and Fitzpatrick do not have a good performance because they almost totally cover the set of Benford probabilities under the 6 different distributions. They are somehow too conservative.

Goodman and Sison are the best simultaneous confidence intervals because they both have small coverage probabilities. Goodman confidence intervals perform better under mixture distributions such as Uniform/Benford and Hill/Benford distributions. On the other hand, Sison has a better performance under other distributions such as the contaminated additive Benford, contaminated multiplicative Benford, and Generalized Benford distributions.

Table 4.15: Coverage proportions for 90%, 95% and 99% simultaneous confidence intervals for data generated using the Uniform distribution

<i>Uniform Distribution</i>	<b>n = 1000</b>			<b>n = 2000</b>		
	90% CI	95% CI	99% CI	90% CI	95% CI	99% CI
<b>Quesenberry</b>	1.000	1.000	1.000	0.966	0.997	1.000
<b>Goodman</b>	0.000	0.000	0.000	0.000	0.000	0.000
<b>Bailey</b>	0.000	0.000	0.000	0.000	0.000	0.000
<b>Angular</b>						
<b>Bailey Square</b>	0.000	0.000	0.000	0.000	0.000	0.000
<b>Fitzpatrick</b>	0.532	0.966	1.000	0.000	0.003	0.702
<b>Sison</b>	0.000	0.000	0.000	0.000	0.000	0.000
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>						

Table 4.16: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.02$ ) with digits 10 to 14, n=1000

<i>Contaminated Additive Distribution (<math>\alpha = 0.02</math>)</i>	<b>n = 1000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.548	0.481	0.423	0.420	0.380
<b>Bailey</b>	0.646	0.597	0.596	0.523	0.485
<b>Angular</b>					
<b>Bailey Square</b>	0.646	0.634	0.595	0.522	0.545
<b>Fitzpatrick</b>	0.965	0.975	0.981	0.980	0.976
<b>Sison</b>	0.278	0.239	0.240	0.262	0.227
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.17: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.02$ ) with digits 10 to 14,  $n=2000$

<i>Contaminated Additive Distribution (<math>\alpha = 0.02</math>)</i>	<b>n =2000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.182	0.138	0.121	0.105	0.085
<b>Bailey</b>	0.237	0.203	0.182	0.145	0.125
<b>Angular</b>					
<b>Bailey Square</b>	0.237	0.203	0.182	0.169	0.125
<b>Fitzpatrick</b>	0.817	0.816	0.831	0.835	0.861
<b>Sison</b>	0.040	0.039	0.036	0.032	0.022
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.18: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.06$ ) with digits 10 to 14,  $n=1000$

<i>Contaminated Additive Distribution (<math>\alpha = 0.06</math>)</i>	<b>n =1000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	0.730	0.642	0.550	0.458	0.348
<b>Goodman</b>	0.000	0.000	0.000	0.000	0.000
<b>Bailey</b>	0.000	0.000	0.000	0.000	0.000
<b>Angular</b>					
<b>Bailey Square</b>	0.000	0.000	0.000	0.000	0.000
<b>Fitzpatrick</b>	0.003	0.004	0.004	0.002	0.002
<b>Sison</b>	0.000	0.000	0.000	0.000	0.000
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					



Table 4.19: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated additive Benford distribution ( $\alpha = 0.06$ ) with digits 10 to 14,  $n=2000$

<i>Contaminated Additive Distribution (<math>\alpha = 0.06</math>)</i>	<b>n =2000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	0.027	0.014	0.004	0.000	0.000
<b>Goodman</b>	0.000	0.000	0.000	0.000	0.000
<b>Bailey</b>	0.000	0.000	0.000	0.000	0.000
<b>Angular</b>					
<b>Bailey Square</b>	0.000	0.000	0.000	0.000	0.000
<b>Fitzpatrick</b>	0.000	0.000	0.000	0.000	0.000
<b>Sison</b>	0.000	0.000	0.000	0.000	0.000
<b>Univariate Binomial</b>	0.000	0.000	0.000	0.000	0.000

Table 4.20: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.2$ ) with digits 10 to 14,  $n=1000$

<i>Contaminated Multiplicative Distribution (<math>\alpha = 1.2</math>)</i>	<b>n =1000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.873	0.877	0.866	0.883	0.882
<b>Bailey</b>	0.895	0.905	0.917	0.912	0.910
<b>Angular</b>					
<b>Bailey Square</b>	0.891	0.908	0.914	0.910	0.910
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.833	0.844	0.858	0.872	0.896
<b>Univariate Binomial</b>	0.001	0.000	0.000	0.001	0.001

Table 4.21: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.2$ ) with digits 10 to 14,  $n=2000$

<i>Contaminated Multiplicative Distribution (<math>\alpha = 1.2</math>)</i>	<b>n =2000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.859	0.858	0.865	0.866	0.879
<b>Bailey</b>	0.872	0.884	0.891	0.899	0.907
<b>Angular</b>					
<b>Bailey Square</b>	0.872	0.885	0.891	0.908	0.907
<b>Fitzpatrick</b>	0.998	0.999	0.999	1.000	1.000
<b>Sison</b>	0.681	0.753	0.782	0.801	0.840
<b>Univariate</b>	0.001	0.000	0.001	0.002	0.000
<b>Binomial</b>					

Table 4.22: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.5$ ) with digits 10 to 14,  $n=1000$

<i>Contaminated Multiplicative Distribution (<math>\alpha = 1.5</math>)</i>	<b>n =1000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.514	0.527	0.575	0.638	0.646
<b>Bailey</b>	0.617	0.650	0.726	0.728	0.740
<b>Angular</b>					
<b>Bailey Square</b>	0.615	0.694	0.725	0.727	0.773
<b>Fitzpatrick</b>	0.959	0.984	0.990	0.996	1.000
<b>Sison</b>	0.251	0.297	0.370	0.468	0.472
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.23: Coverage proportions for 95% simultaneous confidence intervals for data generated using the contaminated multiplicative Benford distribution ( $\alpha = 1.5$ ) with digits 10 to 14,  $n=2000$

<i>Contaminated Multiplicative Distribution (<math>\alpha = 1.5</math>)</i>	<b>n =1000 (95% CI)</b>				
	10	11	12	13	14
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.155	0.189	0.245	0.293	0.341
<b>Bailey</b>	0.192	0.264	0.346	0.371	0.430
<b>Angular</b>					
<b>Bailey Square</b>	0.192	0.264	0.346	0.416	0.430
<b>Fitzpatrick</b>	0.775	0.877	0.935	0.959	0.981
<b>Sison</b>	0.029	0.059	0.103	0.141	0.171
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.001
<b>Binomial</b>					

Table 4.24: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = -0.5, -0.4, -0.3, -0.2, -0.1$ ),  $n=1000$

<i>Generalized Benford Distribution</i>	<b>n =1000 (95% CI)</b>				
	-0.5	-0.4	-0.3	-0.2	-0.1
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.093	0.262	0.478	0.665	0.811
<b>Bailey</b>	0.109	0.370	0.656	0.851	0.920
<b>Angular</b>					
<b>Bailey Square</b>	0.109	0.370	0.656	0.849	0.919
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.008	0.097	0.372	0.728	0.929
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.002
<b>Binomial</b>					

Table 4.25: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = -0.5, -0.4, -0.3, -0.2, -0.1$ ),  $n=2000$

<i>Generalized Benford Distribution</i>	<b>n =2000 (95% CI)</b>				
	-0.5	-0.4	-0.3	-0.2	-0.1
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.000	0.029	0.235	0.590	0.827
<b>Bailey</b>	0.000	0.025	0.277	0.703	0.911
<b>Angular</b>					
<b>Bailey Square</b>	0.000	0.025	0.281	0.706	0.913
<b>Fitzpatrick</b>	0.979	1.000	1.000	1.000	1.000
<b>Sison</b>	0.000	0.002	0.039	0.367	0.838
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.001
<b>Binomial</b>					

Table 4.26: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=1000$

<i>Generalized Benford Distribution</i>	<b>n =1000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.918	0.804	0.549	0.197	0.026
<b>Bailey</b>	0.899	0.807	0.598	0.282	0.048
<b>Angular</b>					
<b>Bailey Square</b>	0.893	0.804	0.597	0.308	0.059
<b>Fitzpatrick</b>	1.000	1.000	0.994	0.951	0.816
<b>Sison</b>	0.845	0.556	0.261	0.041	0.001
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.27: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Generalized Benford distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=2000$

<i>Generalized Benford Distribution</i>	<b>n =2000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.925	0.629	0.169	0.003	0.000
<b>Bailey</b>	0.862	0.632	0.195	0.008	0.000
<b>Angular</b>					
<b>Bailey Square</b>	0.865	0.640	0.204	0.009	0.000
<b>Fitzpatrick</b>	1.000	0.993	0.949	0.711	0.254
<b>Sison</b>	0.766	0.292	0.027	0.000	0.000
<b>Univariate</b>	0.000	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.28: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Uniform/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=1000$

<i>Uniform/Benford Distribution</i>	<b>n =1000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.829	0.670	0.464	0.253	0.081
<b>Bailey Angular</b>	0.925	0.879	0.753	0.551	0.286
<b>Bailey Square</b>	0.924	0.878	0.753	0.551	0.286
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.942	0.851	0.617	0.333	0.077
<b>Univariate</b>	0.001	0.000	0.000	0.000	0.000
<b>Binomial</b>					

Table 4.29: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Uniform/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=2000$

<i>Uniform/Benford Distribution</i>	<b>n =2000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.833	0.610	0.286	0.047	0.002
<b>Bailey Angular</b>	0.920	0.813	0.525	0.158	0.01
<b>Bailey Square</b>	0.919	0.816	0.528	0.162	0.01
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.899	0.628	0.228	0.023	0.001
<b>Univariate Binomial</b>	0.004	0.000	0.000	0.000	0.000

Table 4.30: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Hill/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=1000$

<i>Hill/Benford Distribution</i>	<b>n =1000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.810	0.677	0.453	0.247	0.077
<b>Bailey Angular</b>	0.926	0.884	0.746	0.520	0.275
<b>Bailey Square</b>	0.923	0.882	0.746	0.521	0.275
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	1.000
<b>Sison</b>	0.950	0.871	0.667	0.422	0.159
<b>Univariate Binomial</b>	0.000	0.000	0.001	0.000	0.000

Table 4.31: Coverage proportions for 95% simultaneous confidence intervals for data generated using the Hill/Benford mixture distributions ( $\alpha = 0.1, 0.2, 0.3, 0.4, 0.5$ ),  $n=2000$

<i>Hill/Benford</i> <i>Distribution</i>	<b>n =2000 (95% CI)</b>				
	0.1	0.2	0.3	0.4	0.5
<b>Quesenberry</b>	1.000	1.000	1.000	1.000	1.000
<b>Goodman</b>	0.831	0.569	0.213	0.028	0.002
<b>Bailey</b>	0.926	0.800	0.469	0.135	0.010
<b>Angular</b>					
<b>Bailey Square</b>	0.926	0.804	0.474	0.136	0.010
<b>Fitzpatrick</b>	1.000	1.000	1.000	1.000	0.998
<b>Sison</b>	0.911	0.703	0.303	0.061	0.002
<b>Univariate</b>	0.003	0.000	0.000	0.000	0.000
<b>Binomial</b>					

## Chapter 5

### 5. Conclusion

In this paper, among the eleven test statistics we studied, four of them are likelihood ratio test statistics: LR-multinomial, LR-decreasing, LR-generalized Benford, and LR-Rodriguez. Three of them are Cramèr-von Mises  $W_d^2$ ,  $U_d^2$ , and  $A_d^2$  and Pearson's  $\chi^2$ . We found that the test statistics LR-generalized Benford,  $W_d^2$  and  $A_d^2$  performed well when detecting compliance with Benford's Law for Generalized Benford distribution, Uniform/Benford mixture distribution and Hill/Benford mixture distribution. On the other hand, Pearson's  $\chi^2$  and LR-multinomial test statistics are good for detecting compliance with Benford's Law for the contaminated additive/multiplicative distribution.

Also, we investigated six simultaneous confidence intervals – Quesenberry, Goodman, Bailey Angular, Bailey Square, Fitzpatrick and Sison. Since both the Goodman and Sison simultaneous confidence intervals have better performance, we recommend their use to detect whether data deviate from Benford's Law.

To get an adequate power to test the validity of Benford's Law, we need to have large sample sizes. Since it requires a huge demand on computer time, we used two different sample sizes of  $n = 1000$  and  $n = 2000$  only. Large sample sizes such as  $n = 3000$ ,  $n = 4000$ , etc. will increase the rejection rate for the alternative distribution even for distributions that are close to the null distribution – Benford's Law.

In order to show the extent of the deviation from Benford's Law, we recommend computing and graphing Sison simultaneous confidence interval. In this way, a data examiner can obtain a clear picture of where any departure from Benford's Law lies, and



thereby, better understand the precision inherent in the data. Another important point is the fact that there are 90 digits in the first two digits. Therefore, if the examiner chooses a small sample size such as 100, some of the digits will not contain any data and this fails to provide any valid information. In other words, a reasonably large sample size such as 1000 must be selected when testing conformity with Benford's Law with the first two significant digits.

## Bibliography

- [1] Aggarwal, R., Lucey, B.M. (2007). Psychological Barriers in Gold Prices?. *Review of Financial Economics*; 16: 217-30.
- [2] Bailey, B.J.R. (1980). Large Sample Simultaneous Confidence Intervals for the Multinomial Probabilities Based on Transformations of the Cell Frequencies. *Technometrics*; 22(4): 583-589.
- [3] Benford, F. (1938). The Law of Anomalous Numbers. *Proceedings of the American Philosophical Society*; 78(4): 551-572.
- [4] Brambach, G. (2002). Preispolitik des Handels im Zuge der Euro-Einführung [Retail Pricing Policy During Introduction of the Euro].
- [5] Brown, R.J.C. (2005). Benford's Law and Screening of Analytical Data: the Case of Pollutant Concentrations in Ambient air. *Analyst*; 130(9): 1280-1285.
- [6] Busta B., Weinberg, R., (1998). Using Benford's Law and Neural Networks as a Review Procedure. *Managerial Auditing Journal*; 13(6): 356-366.
- [7] Canessa, E. (2003). Theory of Analogous Force in Number Sets. *Physica A*; 328: 44-52.
- [8] Cho, W.K.T., Gaines, B.J. (2007). Breaking the (Benford) Law: Statistical Fraud Detection in Campaign Finance. *The American Statistician*; 61(3): 218-223.
- [9] Choulakian, V., Lockhart, R.A., and Stephens, M.A. (1994). Cramér-Von Mises Statistics for Discrete Distributions. *The Canadian Journal of Statistics*; 22(1): 125-137.
- [10] De Ceuster, M.K.J., Dhaene, G., and Schatteman, T. (1998). On the Hypothesis of Psychological Barriers in Stock Markets and Benford's Law. *Journal of Empirical Finance*; 5(3): 263-79.
- [11] Duncan, R.L. (1969). A Note on the Initial Digit Problem. *Fibonacci Quarterly*; 7: 474-475.
- [12] Durtschi, C., Hillison, W., and Pacini, C. (2004). The Effective Use of Benford's Law to Assist in Detecting Fraud in Accounting Data. *Journal of Forensic Accounting*; V: 17-34.
- [13] Enron Scandal, Wikipedia. [http://en.wikipedia.org/wiki/Enron\\_scandal](http://en.wikipedia.org/wiki/Enron_scandal). Date Accessed: August 31, 2010.

- [14] Fitzpatrick, S., Scott, A. (1987). Quick Simultaneous Confidence Intervals for Multinomial Proportions. *Journal of the American Statistical Association*; 82(399): 875-878.
- [15] Giles, D.E. (2007). Benford's Law and Naturally Occurring Prices in Certain eBay Auctions. *Applied Economics Letters*; 14(3): 157-161.
- [16] Goodman, L.A. (1965). On Simultaneous Confidence Intervals for Multinomial Proportion. *Technometrics*; 7(2): 247-254.
- [17] Hill, T.P. (1995). A Statistical Derivation of the Significant-Digit Law. *Statistical Science*; 10(4): 354-363.
- [18] Hill, T.P. (1995). The Significant-Digit Phenomenon. *The American Mathematical Monthly*; 102(4): 322-327.
- [19] Hill, T.P. (1988). Random Number Guessing and the First Digit Phenomenon. *Psychological Reports*; 62: 967-971.
- [20] Hill, T.P. (1998). The First Digit Phenomenon. *American Scientist*; 86(4): 358-363.
- [21] Hill, T.P. (1999). The Difficulty of Faking Data. *Chance*; 13(3): 27-31.
- [22] Imhof, J.P. (1961). Computing the Distribution of Quadratic Forms in Normal Variables. *Biometrika*; 48(3/4): 419-426.
- [23] Judge, G., Schechter, L. (2009). Detecting Problems in Survey Data Using Benford's Law. *Journal of Human Resources*. 44(1): 1-24.
- [24] Kuiper, N.H. (1962). Tests Concerning Random Points on a Circle. *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen*; Series A 63: 38-47
- [25] Lesperance, M., Reed, W.J., Stephens, M.A., Wilton, B., and Cartwright, A. (2006). Testing for Benford's Law and Possible Fraud Detection.
- [26] Ley, E. (1996). On the Peculiar Distribution of the U.S. Stock Indexes' Digits. *The American Statistician*; 50(4): 311-313.
- [27] Lockhart, R.A., Spinelli, J.J., and Stephens, M.A. (2007). Cramér-von Mises Statistics for Discrete Distributions with Unknown Parameters. *The Canadian Journal of Statistics*; 35(1): 125-133.
- [28] Lu, O.F., Giles, D.E. (2010). Benford's Law and Psychological Barriers in Certain eBay Auctions. *Applied Economics Letters*; 17(10): 1005-1008.

- [29] Newcomb, S. (1881). Note on the Frequency of Use of the Different Digits in Natural Numbers. *American Journal of Mathematics*; 4(1): 39-40.
- [30] Pietronero, L., Tosatti, E., Tosatti, V., and Vespignani, A. (2001). Explaining the Uneven Distribution of Numbers in Nature: the Laws of Benford and Zipf. *Physica A*; 293: 297-304.
- [31] Quesenberry, C.P., Hurst, D.C. (1964). Large Sample Simultaneous Confidence Intervals for Multinomial Proportions. *Technometrics*; 6(2): 191-195.
- [32] Rodriguez, R.J. (2004). First Significant Digit Patterns from Mixtures of Uniform Distributions. *The American Statistician*; 58(1): 64-71.
- [33] Scheffé, H. (1959). *The Analysis of Variance*. Wiley, New York
- [34] Schindler, R.M., Kirby, P.N. (1997). Patterns of Rightmost Digits used in Advertised Prices: Implications for Nine-Ending Effects. *Journal of Consumer Research*; 24(2): 192-201
- [35] Sehity, T., Hoelzl, E., and Kirchler, E. (2005). Price Developments After a Nominal Shock: Benford's Law and Psychological Pricing After the Euro Introduction. *International Journal of Research in Marketing*; 22(4): 471-480.
- [36] Shao, L., Ma, B-Q. (2010). The Significant Digit Law in Statistical Physics. *Physica A*. 389: 3109-3116.
- [37] Sison, C.P., Glaz, J. (1995). Simultaneous Confidence Intervals and Sample Size Determination for Multinomial Proportions. *Journal of the American Statistical Association*; 90(429): 366-369.
- [38] Stiving, M., Winter, R.S. (1997). An Empirical Analysis of Price Endings with Scanner Data. *Journal of Consumer Research*; 24(1): 57-67.
- [39] Stephens, M.A. (1970). Use of the Kolmogorov-Smirnov, Cramér-Von Mises and Related Statistics Without Extensive Tables. *Journal of the Royal Statistical Association*; 32(1): 115-122.
- [40] WorldCom Scandal: A Look Back at One of the Biggest Corporate Scandals in U.S. History, Associated Content.  
[http://www.associatedcontent.com/article/162656/worldcom\\_scandal\\_a\\_look\\_back\\_at\\_one.html?cat=3](http://www.associatedcontent.com/article/162656/worldcom_scandal_a_look_back_at_one.html?cat=3). Date Accessed: August 31, 2010.

## Appendix A

### BenMain.R

```
#####
# Benford paper simulations
# For 2 digits (10:99) we should run (little) n at least 1000 (do 1000, 2000)
#####

# saveWhere
mdir <- "F:/Stanley/Thesis/Results/2 digits/"

# test significant digits
digits <- 10:99

# number of replications
N <- 1000

# load BenSimus.R, BenSimfcns.R and Simultaneousfcns.R codes
source("F:/Stanley/Thesis/Codes/SW_BenSimfcns.R")
source("F:/Stanley/Thesis/Codes/SW_BenSimus.R")
source("F:/Stanley/Thesis/Codes/SW_Simultaneousfcns.R")

#####
# Set up:
# bign = length of null vector
# litn = multinomial sample size
# digits = 10:99
#####

# LR test stat for testing H_0: Ben vs. H_1: decreasing p
LR.dec.null.n100 <- LR.dec.null.f(bign=10000, litn=100, digits)
LR.dec.null.n1000 <- LR.dec.null.f(bign=10000, litn=1000, digits)
LR.dec.null.n2000 <- LR.dec.null.f(bign=10000, litn=2000, digits)

# Compute eigenvalues required for approx to SL's for CVM stats
# Eigenvalues do not depend on sample sizes
CVME <- CVMEigen(n=100, digits)
CR.eig.W <- CVME$eig.W
CR.eig.U <- CVME$eig.U
CR.eig.A <- CVME$eig.A

# Run Simulations:
# 1. Benford; 2. Uniform; 3. Multiplicative Contaminated Benford;
# 4. Additive Contaminated Benford; 5. Generalized Benford; 6. Uniform/Benford;
# 7. Hill/Benford mixture models

for (n in c(100, 1000, 2000))
{
  #####
  # Simulate under null = Benford
```

```
#####

Sfreq <- MultSim(Ben.ps(digits), n=n, N=N, digits)
fname <- paste(mdir, "RBen/RBenford.n", n, ".N", N, ".txt", sep="")
print(fname)
BenSimus(n, N, fname, Sfreq, digits)

#####
# Simulate under Uniform
#####

Sfreq <- MultSim(rep(1/length(digits), length(digits)), n=n, N=N, digits)
fname <- paste(mdir, "RBen/RUniform.n", n, ".N", N, ".txt", sep="")
print(fname)
BenSimus(n, N, fname, Sfreq, digits)

#####
# Simulate under Multiplicative Contaminated Benford
# Contaminate each of the digits 10:99 with low=1.20 and high=1.50
#####

for (index in 1:length(digits))
{
  for (contam in c(1.20, 1.50))
  {
    cps <- Ben.ps(digits)
    cps[index] <- min(1, cps[index]*contam)
    cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
    Sfreq <- MultSim(cps, n=n, N=N, digits)
    fname <- paste(mdir, "RBen/RConMultiBen", index+length(digits)/9-1, "i",
                  contam, ".n", n, ".N", N, ".txt", sep="")
    print(fname)
    BenSimus(n, N, fname, Sfreq, digits)
  }
}

#####
# Simulate under Additive Contaminated Benford
# Contaminate each of the digits 10:99 with low=0.02 and high=0.06
#####

for (index in 1:length(digits))
{
  for (contam in c(.02, .06))
  {
    cps <- Ben.ps(digits)
    cps[index] <- min(1, cps[index]+contam)
    cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
    Sfreq <- MultSim(cps, n=n, N=N, digits)
    fname <- paste(mdir, "RBen/RConAddBen", index+length(digits)/9-1, "i",
                  contam, ".n", n, ".N", N, ".txt", sep="")
    print(fname)
    BenSimus(n, N, fname, Sfreq, digits)
  }
}

```

```
#####
# Simulate under Generalized Benford
#####

for (alpha in c(seq(-1, -.1, .1), seq(.1, 1, .1)))
{
  Sfreq <- MultSim(genben.ps(alpha,digits), n=n, N=N, digits)
  fname <- paste(mdir, "RBen/RGenBen", alpha, ".n", n, ".N", N, ".txt", sep=")
  print(fname)
  BenSimus(n, N, fname, Sfreq, digits)
}

#####
# Simulate under Mixture Uniform/Benford
#####

for (alpha in seq(.1, .5, .1))
{
  ps <- alpha/length(digits) + (1-alpha)*Ben.ps(digits)
  Sfreq <- MultSim(ps, n=n, N=N, digits)
  fname <- paste(mdir, "RBen/RUniBen", alpha, ".n", n, ".N", N, ".txt", sep=")
  print(fname)
  BenSimus(n, N, fname, Sfreq, digits)
}

#####
# Simulate under Mixture Hill/Benford
#####

if(length(digits)==9)
{
  Hill <- c(.147, .100, .104, .133, .097, .157, .120, .084, .058)
} else {
  Hill <- c(0.0085, 0.0156, 0.0172, 0.0160, 0.0154, 0.0147, 0.0165, 0.0188, 0.0107,
    0.0135, 0.0058, 0.0106, 0.0117, 0.0109, 0.0105, 0.0100, 0.0112, 0.0128,
    0.0073, 0.0092, 0.0060, 0.0110, 0.0122, 0.0113, 0.0109, 0.0104, 0.0116,
    0.0133, 0.0076, 0.0096, 0.0077, 0.0141, 0.0156, 0.0145, 0.0140, 0.0133,
    0.0149, 0.0170, 0.0097, 0.0122, 0.0056, 0.0103, 0.0113, 0.0106, 0.0102,
    0.0097, 0.0109, 0.0124, 0.0071, 0.0089, 0.0091, 0.0166, 0.0184, 0.0171,
    0.0165, 0.0157, 0.0176, 0.0201, 0.0115, 0.0144, 0.0070, 0.0127, 0.0140,
    0.0131, 0.0126, 0.0120, 0.0134, 0.0154, 0.0088, 0.0110, 0.0049, 0.0089,
    0.0098, 0.0092, 0.0088, 0.0084, 0.0094, 0.0108, 0.0061, 0.0077, 0.0034,
    0.0061, 0.0068, 0.0063, 0.0061, 0.0058, 0.0065, 0.0074, 0.0042, 0.0053)
}

for (alpha in seq(.1, .5, .1))
{
  ps <- alpha*Hill + (1-alpha)*Ben.ps(digits)
  Sfreq <- MultSim(ps, n=n, N=N, digits)
  fname <- paste(mdir, "RBen/RHilBen", alpha, ".n", n, ".N", N, ".txt", sep=")
  print(fname)
  BenSimus(n, N, fname, Sfreq, digits)
}
}
```

## BenNonCentral.R

```
# load required libraries
library(MASS)
library(mgcv)

Ben.ps <- function(digits)
{
  # return Benford probabilities for digits=10:99
  return(log10(1+1/digits))
}

Div <- function(a,b)
{
  # returns a/b; zero when b=0
  return((b!=0)*a/(b+(b==0)))
}

CVMPower <- function(digits)
{
  #####
  # Computations for power of Cramer-von Mises type
  # statistics for testing H_0: Benford vs H_1: multinomial p's
  # W2 Cramer-von Mises; U2 Watson; A2 Anderson-Darling; X2 Pearson chi-square
  # digits = first significant digits - 10:99
  # Uses R libraries mgcv, mass
  #####

  k <- length(digits)
  ps <- Ben.ps(digits)
  Tt <- cumsum(ps)
  H <- Tt
  tj <- .5*(ps + c(ps[-1],ps[1]))
  Amat <- matrix(0,nrow=k,ncol=k)
  Amat[row(Amat)>=col(Amat)] <- 1
  Emat <- diag(tj)
  Dmat <- diag(ps)
  Kmat <- diag(Div(1,H*(1-H)))
  Kmat[k,k] <- 0
  Sigma.0 <- (Dmat-(ps%o%ps))
  Sigma.y <- Amat%*%Sigma.0%*%t(Amat)
  Sigma.y[,k] <- 0
  Sigma.y[k,] <- 0
  B <- mroot(Sigma.y,rank=k-1,method='svd')
  Binv <- ginv(B)
  BinvA <- Binv%*%Amat
  eigv.W <- eigen(t(B)%*%Emat%*%B)$vectors
  Utem <- diag(1,k)-(Dmat%*%(rep(1,k)%o%rep(1,k)))
  eigv.U <- eigen(t(B)%*%Utem%*%Emat%*%t(Utem)%*%B)$vectors
  eigv.A <- eigen(t(B)%*%Emat%*%Kmat%*%B)$vectors
  return(list(eigv.W=eigv.W, eigv.U=eigv.U, eigv.A=eigv.A, BinvA=BinvA))
}

Chi.approx <- function(x, eigvals)
{

```



```

# Pearson's 3 moment chi-square approximation to distribution of test stat
k1 <- sum(eigvals)
k2 <- 2*sum(eigvals^2)
k3 <- 8*sum(eigvals^3)
b <- k3/4/k2
p <- 8*k2^3/k3^2
a <- k1-b*p
return(1-pchisq((x-a)/b,p))
}

Chinv.approx <- function(alpha, eigvals)
{
#####
# Pearson's 3 moment chi-square approximation to distribution of test stat
# alpha is lower tail area
# returns approximate quantile
#####

k1 <- sum(eigvals)
k2 <- 2*sum(eigvals^2)
k3 <- 8*sum(eigvals^3)
b <- k3/4/k2
p <- 8*k2^3/k3^2
a <- k1-b*p
return(a+b*qchisq(alpha,p))
}

Chinc.approx <- function(x, eigvals, noncen)
{
#####
# Pearson's 3 moment chi-square approximation to distribution of test stat
# eigvals vector
# noncen vector same length as eigvals
#####

k1 <- sum(eigvals*(1+noncen))
k2 <- 2*sum(eigvals^2*(1+2*noncen))
k3 <- 8*sum(eigvals^3*(1+3*noncen))
b <- k3/4/k2
p <- 8*k2^3/k3^2
a <- k1-b*p
return(1-pchisq((x-a)/b,p))
}

lmhofnc <- function(x, eigvals, noncen, UPPER=NA, subdiv=100, eps1=.0001)
{
#####
# function to compute P(Q>x) for Q a quadratic form in normal variables
# ref: "Computing the distribution of quadratic forms in normal variables"
# Biometrika (1961), 48, 419-426
# where Q=t(x+mu)A(x+mu), x~N(0,Sigma)
# eigvals - vector of eigenvalues of A%%Sigma **there should be no zeroes!
# noncen - vector noncentrality parameters, linear combinations of mu
# length(eigvals)==length(noncen)
#####

```

```

klam <- length(eigvals)*.5
if (is.na(UPPER))
{
    slam <- .5*sum(log(abs(eigvals)))
    UB <- exp(-(log(eps1*pi*klam)+ slam +.5*sum(noncen))/klam)
    sdet <- .5*sum((eigvals*UB)^2*noncen/(1+(eigvals*UB)^2))
    TU <- exp(-(log(pi*klam)+klam*log(UB)+sdet+slam))
} else {
    UB<-UPPER
}
res <- integrate(dlmhofnc, lower=0, upper=UB, xcrit=x, lambda=eigvals,
delta2=noncen,subdivisions=subdiv,
stop.on.error=FALSE)
if(res$message!="OK") print(res$message)
return(.5+res$value/pi)
}

dlmhofnc <- function(u, xcrit, lambda, delta2)
{
#####
# integrand for the lmhof function
# ref: "Computing the distribution of quadratic forms in normal variables"
# Biometrika (1961), 48, 419-426
# u = evaluate the integrand at u
# xcrit = x critical P(Q>x)
# lambda = vector of eigenvalues of A%%Sigma
# delta2 = noncentrality vector
#####

ulambda <- u%o%lambda
thsum <- (u%o%(delta2*lambda))/(1+ulambda^2)
theta <- .5*as.vector(apply(atan(ulambda)+thsum,1,sum))-.5*xcrit*u
rsum <- .25*log(1+ulambda^2)+.5*(u%o%(sqrt(delta2)*lambda))^2/(1+ulambda^2)
rho <- exp(as.vector(apply(rsum,1,sum)))
return(sin(theta)/u/rho)
}

```

## BenSimfcns.R

```

# load required library for CVMEigen
library(mgcv)

MultSim <- function(p, n, N, digits, myseed=524896987)
{
#####
# Generate N multinomial (n,p) frequencies in N*n matrix
# p = multinomial probabilities for parent distn -
# n = multinomial sample sizes
# N = number of simulation replications
# digits = significant digits vector - 10:99
#####

set.seed(myseed)

```

```

k <- max(digits)
m <- min(digits)
sU <- sample(digits, n*N, replace=TRUE, prob=p)
sU <- matrix(sU, nrow=N, ncol=n)
if(m==1)
{
  sF <- t(apply(sU, 1, tabulate, k))
} else {
  sF <- t(apply(sU, 1, tabulate, k))[, -c(1:(m-1))]
}
return(sF)
}

Chi.square <- function(o, e, df)
{
  #####
  # o = vector or matrix of observed frequencies with samples in columns
  # e = vector or matrix of expected frequencies with samples in columns
  # Need to have dim(as.matrix(o))==dim(as.matrix(e))
  #####

  X <- apply(as.matrix(Div((o-e)^2, e)), 2, sum)
  return(1-pchisq(X, df))
}

Ben.ps <- function(digits)
{
  # return Benford probabilities for digits=10:99
  return(log10(1+1/digits))
}

aLoga <- function(a) # used for LR.mat.mult
{
  # function to return a*log(a) - returns zero if a is zero
  return(a*log(a+(a==0)))
}

bLoga <- function(b, a)
{
  # function to return b*log(a) - returns zero if a and b are zero
  return(b*log(a+(a==0 & b==0)))
}

Div <- function(a, b)
{
  # returns a/b; zero when b=0
  return((b!=0)*a/(b+(b==0)))
}

LR.mat.mult <- function(freq, digits)
{
  #####
  # Likelihood ratio test of Benford versus general multinomial
  # freq - matrix with multinomial samples in the rows
  # digits = significant digits vector, 10:99
  #####

```

```

log.l1 <- apply(aLoga(freq), 1, sum) - aLoga(apply(freq, 1, sum))
log.l0 <- apply(t(freq)*log(Ben.ps(digits)), 2, sum)
LR <- 2*(log.l1-log.l0)
return(1 - pchisq(LR, length(digits)-1))
}

LR.dec <- function(frequencies, LR.dec.null=c(0), digits)
{
#####
# LR test stat (-2 log(lambda)) for testing H_0: Ben vs. H_1: decreasing p
# frequencies - multinomial frequency vector for digits
# digits = significant digits vector, 10:99
# LR.null = large vector of simulated values of LR.dec under H_0
# returns LR (to generate LR.null) and Sig Level
# convergence = 0 means successful convergence from nlminb
#####

assign('freq', frequencies, 1)
k <- length(digits)
log.l1 <- nlminb(objective=dec.objf, start = rep(1, k), lower = rep(0, k),
freq = freq)
log.l0 <- sum(freq * log(Ben.ps(digits)))
LR <- 2 * (-log.l1$objective - log.l0)
return(list(LR=LR, SL=sum(LR.dec.null >= LR)/length(LR.dec.null),
conv=log.l1$convergence))
}

dec.objf <- function(zs, freq)
{
#####
# Multinomial loglikelihood as a function of zs
# Objective (-log.lik) for finding MLEs of first significant digit probs
# assuming p1 >= p2 >= p3....
# z[i] = ps[i]-ps[i+1]
# freq = vector of multinomial frequencies
#####

ps <- rev.cumsum(zs)
ps <- ps/sum(ps)
return(- sum(bLoga(freq, ps)))
}

rev.cumsum <- function(v)
{
# v is numeric vector; returns reverse cumulative sum
return(rev(cumsum(rev(v))))
}

LR.dec.null.f <- function(bign, litn, digits)
{
# LR.dec - Generate large null vector to compute p-values for LR.dec
LR <- conv <- vector("numeric")
data.sim <- MultSim(Ben.ps(digits), n=litn, N=bign, digits)
for (i in 1:bign)
{

```

```

        LR.dec.run <- LR.dec(data.sim[i,], LR.dec.null=c(0), digits)
        LR <- c(LR, LR.dec.run$LR)
        conv <- c(conv, LR.dec.run$conv)
    }
    print(paste('Number of calls to nlmnb not converging ', sum(conv!=0)))
    return(list(LR=LR, conv=conv))
}

LR.genben <- function(frequencies, digits)
{
    #####
    # LR test stat (-2 log(lambda)) for testing H_0:Ben vs. H_1: gen.Ben
    # freq - multinomial frequency data for digits
    # digits - significant digits vector, 1:9, 10:99, etc.
    #####

    assign('freq', frequencies, 1)
    assign('digs', digits, 1)
    log.l1 <- nlmnb(objective=genben.ml.objf, start = 0.2, freq = freq, digits=digs )
    log.l0 <- sum(freq * log(Ben.ps(digits)))
    LR <- 2 * (-log.l1$objective - log.l0)
    #print(LR)
    return(list(SL=1 - pchisq(LR, 1),conv=log.l1$conv))
}

genben.ml.objf <- function(alpha, freq, digits)
{
    # Computes -log.lik for data freqs using gen.ben with parameter alpha
    return(- sum(bLoga(freq,genben.ps(alpha,digits))))
}

genben.ps <- function(alpha, digits)
{
    # Computes generalized Benford probabilities
    # alpha - parameter
    # digits - significant digits vector, 1:9, 10:99, etc.
    return((digits^(-alpha)-(digits+1)^(-alpha))/(min(digits)^(-alpha)-max(digits + 1)^(-alpha)))
}

LR.rod <- function(frequencies, digits)
{
    #####
    # LR test stat (-2 log(lambda)) for testing H_0:Ben vs. H_1:
    # Rodriguez gen.Ben
    # frequencies - multinomial frequencies for digits
    # digits - significant digits vector, eg. 1:9, 10:99, etc.
    #####

    assign('freq', frequencies, 1)
    assign('digs', digits, 1)
    log.l1 <- nlmnb(objective=rodben.ml.objf, start = -0.5, freq = freq, digits=digs,
                    control=list(trace=1))

    if(log.l1$par>40)
    {
        log.l1$objective <- -(sum(aLoga(freq))-aLoga(sum(freq)))
    }
}

```

```

        log.l1$convergence<-2
    }
    log.l0 <- sum(freq * log(Ben.ps(digits)))
    LR <- 2 * (-log.l1$objective - log.l0)
    return(list(SL=1 - pchisq(LR, 1),conv=log.l1$convergence))
}

rodben.ml.objf <- function(beta, freq, digits)
{
    #####
    # Computes -log.lik for data freqs using Rodriguez's generalized Benford with beta
    #####

    return( - sum(bLoga(freq, rodben.ps(beta, digits))))
}

rodben.ps <- function(beta, digits)
{
    #####
    # Computes probabilities using Rodriguez's generalization of Benford.
    # beta - parameter
    # digits - significant digits vector - 10:99
    #####

    return((beta + 1)/(length(digits) * beta) - ((digits + 1)^(beta + 1) - digits^(beta + 1))/
            (beta * (max(digits + 1)^(beta + 1) - min(digits)^(beta + 1))))
}

CVMStats <- function(freq, digits)
{
    #####
    # Computes Cramer-von Mises type statistics for testing
    # H_0: Benford vs H_1: multinomial p's
    # W2 Cramer-von Mises; U2 Watson; A2 Anderson-Darling; X2 Pearson chi-square;
    # Kuiper's (Vstar) test (see Stephens 1970; Giles working paper, EWP0505)
    # freq - vector of multinomial frequencies
    # digits = significant digits vector - 10:99
    # uses constants for digits 10:99; CR.eig.W CR.eigen.U CR.eig.A CR.eig.X
    #####

    k <- length(freq)
    n <- sum(freq)
    ps <- Ben.ps(digits)
    e <- n*ps #expected under Bendford
    S <- cumsum(freq)
    Tt <- cumsum(e)
    Z <- S-Tt
    H <- Tt/n
    tj <- .5*(ps + c(ps[-1],ps[1]))
    Zbar <- sum(tj*Z)
    W2 <- sum(tj*Z^2)/n
    W <- Imhof(W2, CR.eig.W)
    if (W<0) {W <- Imhof(W2, CR.eig.W, UPPER=Inf, subdiv=500)}
    Wap <- Chi.approx(W2, CR.eig.W)
    U2 <- sum(tj*(Z-Zbar)^2)/n
    U <- Imhof(U2, CR.eig.U)
}

```

```

    if (U<0) {U <- lmfhof(U2, CR.eig.U, UPPER=Inf, subdiv=500)}
    Uap <- Chi.approx(U2, CR.eig.U)
    A2 <- sum(Div(tj*Z^2,H*(1-H)))/n
    A <- lmfhof(A2, CR.eig.A)
    if (A<0) {A <- lmfhof(A2, CR.eig.A, UPPER=Inf, subdiv=500)}
    Aap <- Chi.approx(A2, CR.eig.A)
    X2 <- sum(Div((freq-e)^2, e))
    X <- 1-pchisq(X2, k-1)
    Vstar <- (max(Z)+max(-Z))*(sqrt(n)+.155+.24/sqrt(n))/n
    #print(c(W2, W, U2, U, A2, A, X2, X, Vstar))
    return(list(W=W, Wap=Wap, U=U, Uap=Uap, A=A, Aap=Aap, X=X, Vstar=Vstar))
}

CVMEigen <- function(n, digits)
{
  #####
  # Computes eigenvalues for computation of significance levels of Cramer-von Mises type
  # statistics for testing H_0: Benford vs H_1: multinomial p's
  # W2 Cramer-von Mises; U2 Watson; A2 Anderson-Darling; X2 Pearson chi-square
  # n = sum of multinomial frequencies (sample size)
  # digits = significant digits vector - 10:99
  # Uses mroot function from library mgcv
  #####

  library(mgcv)

  k <- length(digits)
  ps <- Ben.ps(digits)
  e <- n*ps # expected under Benford
  Tt <- cumsum(e)
  H <- Tt/n
  tj <- .5*(ps + c(ps[-1],ps[1]))
  Amat <- matrix(0,nrow=k,ncol=k)
  Amat[row(Amat)>=col(Amat)] <- 1
  Emat <- diag(tj)
  Dmat <- diag(ps)
  Kmat <- diag(Div(1,H*(1-H)))
  Kmat[k,k] <- 0
  Sigma.0 <- (Dmat-(ps%o%ps))
  Sigma.y <- Amat%*%Sigma.0%*%t(Amat)
  Sigma.y[,k] <- 0
  Sigma.y[k,] <- 0
  B <- mroot(Sigma.y,rank=k-1,method='svd')
  eig.W <- eigen(t(B)%*%Emat%*%B)$values
  Utem <- diag(1,k)-(Emat%*%(rep(1,k)%o%rep(1,k)))
  eig.U <- eigen(t(B)%*%Utem%*%Emat%*%t(Utem)%*%B)$values
  eig.A <- eigen(t(B)%*%Emat%*%Kmat%*%B)$values
  return(list(eig.W=eig.W, eig.U=eig.U, eig.A=eig.A))
}

lmhof <- function(x, eigvals, UPPER=NA, subdiv=100, eps1=.0001)
{
  #####
  # function to compute P(Q>x) for Q a quadratic form in normal variables
  # ref: "Computing the distribution of quadratic forms in normal variables"
  # Biometrika (1961), 48, 419-426

```

```

# where Q=t(x)Ax, x~N(0,Sigma)
# eigvals - eigenvalues of A%%Sigma **there should be no zeroes!
#####

klam <- length(eigvals)*.5
if (is.na(UPPER))
{
  UB <- exp(-(log(eps1*pi*klam)+.5*sum(log(abs(eigvals)))))/klam)
} else {
  UB <- UPPER
}
res <- integrate(dlmhof, lower=0, upper=UB, xcrit=x, lambda=eigvals,
  subdivisions=subdiv,
  stop.on.error=FALSE)
if(res$message!="OK") print(res$message)
return(.5+res$value/pi)
}

dlmhof <- function(u, xcrit, lambda)
{
  #####
  # integrand for the lmhof function
  # ref: "Computing the distribution of quadratic forms in normal variables"
  # Biometrika (1961), 48, 419-426
  # u = evaluate the integrand at u
  # xcrit = x critical P(Q>x)
  # lambda = eigenvalues of A%%Sigma
  #####

  ulambda <- u%o%lambda
  theta <- .5*as.vector(apply(atan(ulambda), 1, sum))-.5*xcrit*u
  rho <- exp(.25*as.vector(apply(log(1+ulambda^2), 1, sum)))
  return(sin(theta)/u/rho)
}

Chi.approx <- function(x,eigvals)
{
  # Pearson's 3 moment chi-square approximation to distribution of test stat
  k1 <- sum(eigvals)
  k2 <- 2*sum(eigvals^2)
  k3 <- 8*sum(eigvals^3)
  b <- k3/4/k2
  p <- 8*k2^3/k3^2
  a <- k1-b*p
  return(1-pchisq((x-a)/b,p))
}

SimultBenp <- function(freq, alpha=0.05, digits)
{
  #####
  # function to compare multinomial confidence intervals
  # with Benford probabilities of digits
  # returns indicator vector of length 7 for the 7 CIs in SimultConf
  # where 0=at least one Benford p outside of interval; 1=all B. p's in interval
  # freq = multinomial frequencies of digits
  # digits = significant digits vector - 10:99

```



```

# length(freq)==length(digits)
#####

n <- sum(freq)
ps <- matrix(Ben.ps(digits), nrow=length(digits), ncol=7)
CI <- SimultConf(freq, alpha)
Ind <- (CI$Lower<= ps) & (CI$Upper >= ps)
return(apply(Ind, 2, prod))
}

```

## BenSimus.R

```

BenSimus <- function(n, N, fname, Sfreq, digits)
{
  #####
  # All simulations use global eigenvalues CR2.eig.W, CR2.eig.U, CR2.eig.A
  # n = multinomial sample sizes
  # N = number of simulation replications
  # fname = file's name
  # Sfreq = multinomial frequencies of the digits
  # digits = significant digits vector - 10:99
  #####

  write(paste("mult dec deconv gen genconv rod rodconv W Wap U Uap A Aap X Vstar",
    "Ques.10 Good.10 Bang.10 Bsqrt.10 Fitz.10 Sison.10 UniBin.10",
    "Ques.05 Good.05 Bang.05 Bsqrt.05 Fitz.05 Sison.05 UniBin.05",
    "Ques.01 Good.01 Bang.01 Bsqrt.01 Fitz.01 Sison.01 UniBin.01"), fname)

  # vector of length N of SL's
  mult.sl <- LR.mat.mult(Sfreq, digits)

  #####
  # returns SL's for multinomial, decreasing(+conv), generalized(+conv), rodrig(+conv), W,
  # U, A, X
  # returns test stat for Kuiper's V (crit values for Vstar 1.620, 1.747, 2.001) (.1, .05, .01)
  # returns indicators of concordance with Benford for 7 CI's
  # 1=Benford in interval; 0=Benford not in interval
  #####

  for(i in 1:N)
  {
    write(i, paste(mdir, "Iterations.txt"), append=TRUE)
    print(paste('iteration ', i))

    # null vector for LR.dec depends on sample size n
    if(n==100)
    {
      dec <- LR.dec(Sfreq[i,], LR.dec.null.n100$LR, digits)
    } else if(n==1000) {
      dec <- LR.dec(Sfreq[i,], LR.dec.null.n1000$LR, digits)
    } else if(n==2000) {
      dec <- LR.dec(Sfreq[i,], LR.dec.null.n2000$LR, digits)
    }
  }
}

```

```

        gen <- LR.genben(Sfreq[i,], digits)
        rod <- LR.rod(Sfreq[i,], digits)
        CVM <- CVMStats(Sfreq[i,], digits)
        SimultBenp.10 <- SimultBenp(Sfreq[i,], alpha=.10, digits)
        SimultBenp.05 <- SimultBenp(Sfreq[i,], alpha=.05, digits)
        SimultBenp.01 <- SimultBenp(Sfreq[i,], alpha=.01, digits)
        write(round(c(mult.sl[i], dec$SL, dec$conv, gen$SL, gen$conv, rod$SL, rod$conv,
                    CVM$W, CVM$Wap, CVM$U, CVM$Uap, CVM$A, CVM$Aap, CVM$X,
                    CVM$Vstar, SimultBenp.10, SimultBenp.05, SimultBenp.01), 4), fname,
              ncolumns=36, append=TRUE)
    }
}

```

## BenSummary.R

```

BenSummary <- function(fnames, shnames, plotnames, N=1000, alphacrit=.05)
{
    #####
    # alpha = .05
    # fnames = file names vector of character entries
    # shnames = shorts names for column headings
    #####

    SLs <- c(1, 2, 4, 6, 8, 9, 10, 11, 12, 13, 14)
    Convs <- c(3, 5, 7)
    CIs <- 16:36
    Vst <- 15
    Vstar.crit <- c(1.620, 1.747, 2.001)
    num <- length(fnames)
    res <- matrix(0, nrow=19, ncol=0)

    for (i in 1:num)
    {
        RBen <- read.table(fnames[i], header=TRUE)
        rej <- apply(RBen[,SLs] < .05, 2, mean, na.rm=TRUE)
        rej <- c(rej, mean(RBen[,Vst] > Vstar.crit[2], na.rm=TRUE))
        rej <- c(rej, 1-apply(RBen[,CIs[8:14]], 2, mean, na.rm=TRUE))
        res <- cbind(res, rej)
    }

    rownames(res) <- names(RBen)[c(c(1, 2, 4, 6), 8:15, 23:29)]
    colnames(res) <- shnames
    return(res)
}

```

## MyErrorBar.R

```

# modification of errbar from library Hmisc
myerrbar <- function(x, y, yplus, yminus, cap = 0.015, xlab = as.character(substitute(x)),
                    ylab = if (is.factor(x) || is.character(x)) "" else as.character(substitute(y)),

```

```

        add = FALSE, lty = 1, ylim, lwd = 1, Type = rep(1, length(y)), ...)
{
  if (missing(ylim))
  {
    ylim <- range(y[Type == 1], yplus[Type == 1], yminus[Type == 1], na.rm = TRUE)
  }

  if (is.factor(x) || is.character(x))
  {
    x <- as.character(x)
    n <- length(x)
    t1 <- Type == 1
    t2 <- Type == 2
    n1 <- sum(t1)
    n2 <- sum(t2)
    omai <- par("mai")
    mai <- omai
    mai[2] <- max(strwidth(x, "inches")) + 0.25 * .R.
    par(mai = mai)
    on.exit(par(mai = omai))
    plot(0, 0, xlab = ylab, ylab = "", xlim = ylim, ylim = c(1, n + 1), axes = FALSE)
    axis(1)
    w <- if (any(t2))
      n1 + (1:n2) + 1
    else numeric(0)
    axis(2, at = c(1:n1, w), labels = c(x[t1], x[t2]), las = 1, adj = 1)
    points(y[t1], 1:n1, pch = 16)
    segments(yplus[t1], 1:n1, yminus[t1], 1:n1)
    if (any(Type == 2))
    {
      abline(h = n1 + 1, lty = 2)
      offset <- mean(y[t1]) - mean(y[t2])
      if (min(yminus[t2]) < 0 & max(yplus[t2]) > 0)
      lines(c(0, 0) + offset, c(n1 + 1, par("usr")[4]), lty = 2)
      points(y[t2] + offset, w, pch = 16)
      segments(yminus[t2] + offset, w, yplus[t2] + offset, w)
      at <- pretty(range(y[t2], yplus[t2], yminus[t2]))
      axis(3, at = at + offset, label = format(round(at, 6)))
    }
    return(invisible())
  }
}

if (!add)
{
  plot(x, y, ylim = ylim, xlab = xlab, ylab = ylab, ...)
  xcoord <- par()$usr[1:2]
  segments(x, yminus, x, yplus, lty = lty, lwd = lwd, ...)
  smidge <- cap * (xcoord[2] - xcoord[1])/2
  segments(x - smidge, yminus, x + smidge, yminus, lwd = lwd, ...)
  segments(x - smidge, yplus, x + smidge, yplus, lwd = lwd, ...)
  invisible()
}
}

```

## Simultaneousfcns.R

```

SimultConf <- function(N, alpha=.05)
{
#####
# Author: Bree Wilton; Modifications: Mary Lesperance
# This function takes a vector of observed cell frequencies from a multinomial distribution
# and returns simultaneous confidence intervals for proportions using six different
# methods given a specified significance level, alpha.
# ref: Hou,C.D., Chiang,J.T. and Tai,J.J. (2003) "A family of simultaneous confidence
# intervals for multinomial proportions," Computational Statistics and Data Analysis, 43,
# 29-45.
# N is the vector of observed frequencies
# k is the length of the vector N
# A is the upper 100*alpha percentile of the chi-square distribution with k-1 d.f.
# B is the upper 100*(alpha/k) percentile of the chi-square distribution with one d.f.
# C is B/(4*n)
# D is the upper 100*(alpha/4) percentile of the standard normal distribution
# E is the upper 100*(alpha/2) percentile of the standard normal distribution
# requires function v() defined below
#####

n <- sum(N)
k <- length(N)
A <- qchisq(1-alpha, k-1)
B <- qchisq(1-(alpha/k), 1)
C <- B/(4*n)
D <- qnorm(1-alpha/4)
E <- qnorm(1-alpha/2)
phat <- N/n

### Quesenberry and Hurst's method
Lower1 <- pmax(0, round((A+2*N-sqrt(A*(A+4*(N*(n-N)/n))))/(2*(n+A)), 4))
Upper1 <- pmin(1, round((A+2*N+sqrt(A*(A+4*(N*(n-N)/n))))/(2*(n+A)), 4))

### Goodman's method
Lower2 <- pmax(0, round((B+2*N-sqrt(B*(B+4*(N*(n-N)/n))))/(2*(n+B)), 4))
Upper2 <- pmin(1, round((B+2*N+sqrt(B*(B+4*(N*(n-N)/n))))/(2*(n+B)), 4))

### Bailey's angular transformation method
Lower3 <- pmax(0, round((sin(asin(sqrt((N+3/8)/(n+3/4)))-sqrt(B/(4*n+2))))^2, 4))
Upper3 <- pmin(1, round((sin(asin(sqrt((N+3/8)/(n+3/4)))+sqrt(B/(4*n+2))))^2, 4))

### Bailey's square root transformation method
Lower4 <- pmax(0, round((sqrt((N+3/8)/(n+1/8))-sqrt(C*(C+1-
(N+3/8)/(n+1/8))))^2/(C+1)^2, 4))
Upper4 <- pmin(1, round((sqrt((N+3/8)/(n+1/8))+sqrt(C*(C+1-
(N+3/8)/(n+1/8))))^2/(C+1)^2, 4))

### Fitzpatrick and Scott's method
Lower5 <- pmax(0, round((phat)-D/(2*sqrt(n)), 4))
Upper5 <- pmin(1, round((phat)+D/(2*sqrt(n)), 4))

### Sison and Glaz's method
# requires function v() which is defined below

```

```

tau <- 1
vtau <- v(N, tau)
vtaupl <- v(N, tau+1)

while(vtaupl < (1-alpha))
{
    tau <- tau+1
    vtau <- vtaupl
    vtaupl <- v(N, tau+1)
}
gamma <- ((1-alpha)-vtau)/(vtaupl-vtau)

Lower6 <- pmax(0, round((phat)-(tau/n), 4))
Upper6 <- pmin(1, round((phat)+(tau+2*gamma)/n, 4))

### Simple univariate Normal approx to Binomial intervals
Lower7 <- pmax(0, round(phat-E*sqrt(phat*(1-phat)/n), 4))
Upper7 <- pmin(1, round(phat+E*sqrt(phat*(1-phat)/n), 4))

tnames <- c("Ques", "Good", "Bang", "Bsqr", "Fitz", "Sison", "UniBin")
Lower <- cbind(Lower1, Lower2, Lower3, Lower4, Lower5, Lower6, Lower7)
Upper <- cbind(Upper1, Upper2, Upper3, Upper4, Upper5, Upper6, Upper7)
dimnames(Lower)[[2]] <- tnames
dimnames(Upper)[[2]] <- tnames
return(list(Lower=Lower, Upper=Upper))
}

v <- function(N,tau)
{
    # function required by SimultConf to compute Sison & Glaz's method
    n <- sum(N)
    k <- length(N)
    lambda <- N
    a <- N+tau
    b <- N-tau

    # factorial moments for the truncated Poisson random variable
    mu1 <- lambda^1*(1+(((ppois(b-1,N)-ppois(b-2,N))-(ppois(a,N)-ppois(a-1,N)))/(ppois(a,N)-ppois(b-1,N))))
    mu2 <- lambda^2*(1+(((ppois(b-1,N)-ppois(b-3,N))-(ppois(a,N)-ppois(a-2,N)))/(ppois(a,N)-ppois(b-1,N))))
    mu3 <- lambda^3*(1+(((ppois(b-1,N)-ppois(b-4,N))-(ppois(a,N)-ppois(a-3,N)))/(ppois(a,N)-ppois(b-1,N))))
    mu4 <- lambda^4*(1+(((ppois(b-1,N)-ppois(b-5,N))-(ppois(a,N)-ppois(a-4,N)))/(ppois(a,N)-ppois(b-1,N))))

    # central moments in terms of factorial moments
    sigma <- sqrt(mu2+mu1-mu1^2)
    mu3i <- mu3+3*mu2-3*mu2*mu1+mu1-3*mu1^2+2*mu1^3
    mu4i <- mu4+6*mu3+7*mu2+mu1-4*mu1*mu3-12*mu1*mu2-4*mu1^2+6*mu1^2*mu2+6*mu1^3-3*mu1^4
    gamma1 <- sum(mu3i)/sum(sigma^2)^(3/2)
    gamma2 <- sum(mu4i-3*sigma^4)/(sum(sigma^2)^2)

    A <- 1/(dpois(n,n))
    S <- (n-sum(mu1))/(sqrt(sum(sigma^2)))

```

```

        vtau <- A*prod(ppois(a,N)-ppois(b-1,N))*f(S,gamma1,gamma2)/sqrt(sum(sigma^2))

    return(vtau)
}

f <- function(x, gamma1, gamma2)
{
    return((exp(-x^2/2)/sqrt(2*pi))*(1+(gamma1/6*(x^3-3*x))+(gamma2/24*(x^4-
        6*x^2+3))+(gamma1^2/72*(x^6-15*x^4+45*x^2-15))))
}

```

## PowerSampleSize.R

```

#####
# Approximate Power vs Sample Sizes
#####

# files directory
mdir <- "F:/Stanley/Thesis/Results/2digits/"

# length of multinomial vector
digits <- 10:99

# Number of replications
N <- 1000

# Number of sample sizes for each replication
n <- 1000

# load BenNonCentral, BenSimfcns and BenSummary2 codes
source("F:/Stanley/Thesis/Codes/SW_BenSimus.R")
source("F:/Stanley/Thesis/Codes/SW_BenNonCentral.R")
source("F:/Stanley/Thesis/Codes/SW_BenSimfcns.R")
source("F:/Stanley/Thesis/Codes/SW_BenSummary2.R")

# Compute eigenvalues required for approx to SL's for CVM stats
CVME <- CVMEigen(n, digits)
CR.eig.W <- CVME$eig.W
CR.eig.U <- CVME$eig.U
CR.eig.A <- CVME$eig.A

# approximate critical values for CVM tests
Wcrit <- Chinv.approx(.95, CR.eig.W)
Ucrit <- Chinv.approx(.95, CR.eig.U)
Acrit <- Chinv.approx(.95, CR.eig.A)
Xcrit <- qchisq(.95, length(digits)-1)

noncen <- CVMPower(digits)
eigv.W <- noncen$eigv.W
eigv.U <- noncen$eigv.U
eigv.A <- noncen$eigv.A
Binva <- noncen$Binva

```

```
#####
# Hill/Benford #
#####

alpha <- seq(.05, .5, .05)
nlist <- 1000*c(0.1, 1:5)
Power.HB <- matrix(0, nrow=length(alpha), ncol=length(nlist))
Hill <- c(0.0085, 0.0156, 0.0172, 0.0160, 0.0154, 0.0147, 0.0165, 0.0188, 0.0107, 0.0135,
          0.0058, 0.0106, 0.0117, 0.0109, 0.0105, 0.0100, 0.0112, 0.0128, 0.0073, 0.0092,
          0.0060, 0.0110, 0.0122, 0.0113, 0.0109, 0.0104, 0.0116, 0.0133, 0.0076, 0.0096,
          0.0077, 0.0141, 0.0156, 0.0145, 0.0140, 0.0133, 0.0149, 0.0170, 0.0097, 0.0122,
          0.0056, 0.0103, 0.0113, 0.0106, 0.0102, 0.0097, 0.0109, 0.0124, 0.0071, 0.0089,
          0.0091, 0.0166, 0.0184, 0.0171, 0.0165, 0.0157, 0.0176, 0.0201, 0.0115, 0.0144,
          0.0070, 0.0127, 0.0140, 0.0131, 0.0126, 0.0120, 0.0134, 0.0154, 0.0088, 0.0110,
          0.0049, 0.0089, 0.0098, 0.0092, 0.0088, 0.0084, 0.0094, 0.0108, 0.0061, 0.0077,
          0.0034, 0.0061, 0.0068, 0.0063, 0.0061, 0.0058, 0.0065, 0.0074, 0.0042, 0.0053)

for (nin in 1:length(nlist))
{
  for (ai in (1:length(alpha)))
  {
    ps <- alpha[ai]*Hill + (1-alpha[ai])*Ben.ps(digits)
    mu <- sqrt(nlist[nin])*(Ben.ps(digits)-ps)
    BinvAmu <- as.vector(BinvA%%mu)

    #mu2.W <- as.vector(eigv.W%%BinvAmu)^2
    #Power.HB[ai, nin] <- lmhofnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

    #mu2.U <- as.vector(eigv.U%%BinvAmu)^2
    #Power.HB[ai,nin] <- lmhofnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

    mu2.A <- as.vector(eigv.A%%BinvAmu)^2
    Power.HB[ai,nin] <- lmhofnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

    #psi2 <- mu%%diag(1/Ben.ps(digits))%%mu
    #Power.HB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

    dimnames(Power.HB)[[2]]<-paste("n=", nlist, sep=")
  }
}

nt <- 1:length(nlist)
matplot(alpha, Power.HB, type='l', lty=nt, col=nt, ylim=c(0,1), ylab='Power', xlab='Alpha, mixing
proportion')

#title('Power for W2 under Mixed Hill/Benford')
#legend(0.35, 0.40, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

#title('Power for U2 under Mixed Hill/Benford')
#legend(0.375, 0.35, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

title('Power for A2 under Mixed Hill/Benford')
legend(0.35, 0.40, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

#title('Power for X2 under Mixed Hill/Benford')
```

```

#legend(0.35, 0.60, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

#####
# Uniform/Benford #
#####

alpha <- seq(.05, .5, .05)
nlist <- 1000*c(0.1, 1:5)
Power.UB <- matrix(0, nrow=length(alpha), ncol=length(nlist))

for (nin in 1:length(nlist))
{
  for (ai in (1:length(alpha)))
  {
    ps <- alpha[ai]/length(digits) + (1-alpha[ai])*Ben.ps(digits)
    mu <- sqrt(nlist[nin])*(Ben.ps(digits)-ps)
    BinvAmu <- as.vector(BinvA%%mu)

    mu2.W <- as.vector(eigv.W%%BinvAmu)^2
    Power.UB[ai, nin] <- Imhofnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

    mu2.U <- as.vector(eigv.U%%BinvAmu)^2
    Power.UB[ai,nin] <- Imhofnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

    mu2.A <- as.vector(eigv.A%%BinvAmu)^2
    Power.UB[ai,nin] <- Imhofnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

    psi2 <- mu%%diag(1/Ben.ps(digits))%%mu
    Power.UB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

    dimnames(Power.UB)[[2]]<-paste("n=", nlist, sep="")
  }
}

nt <- 1:length(nlist)
matplot(alpha, Power.UB, type='l', lty=nt, col=nt, ylim=c(0,1),
  ylab='Power', xlab='Alpha, mixing proportion')

title('Power for W2 under Mixed Uniform/Benford')
legend(0.35, 0.40, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

#title('Power for U2 under Mixed Uniform/Benford')
#legend(0.375, 0.35, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

#title('Power for A2 under Mixed Uniform/Benford')
#legend(0.35, 0.40, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

#title('Power for X2 under Mixed Uniform/Benford')
#legend(0.35, 0.60, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

#####
# Additive Contaminated #
#####

contam <- seq(.01, .15, .01)
nlist <- 1000*c(0.1, 1:5)

```



```

nt <- 1:length(nlist)
Power.AC <- matrix(0, nrow=length(contam), ncol=length(nlist))

par(mfrow=c(3,3))

for (index in 82:90)
{
  for (nin in 1:length(nlist))
  {
    for (conin in (1:length(contam)))
    {
      cps <- Ben.ps(digits)
      cps[index] <- cps[index]+contam[conin]
      cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
      mu <- sqrt(nlist[nin])*(Ben.ps(digits)-cps)
      psi2 <- mu%*%diag(1/Ben.ps(digits))%*%mu
      Power.AC[conin, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)
    }
  }
  #jpeg(paste(mdir,"RBen/Benplots/AddCon-dig", index+9, ".jpeg", sep=""))
  matplot(contam, Power.AC, type='l', lty=nt, col=nt, ylim=c(0,1), ylab='Power',
xlab='Contamination')
  #legend(.1, .4, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)
  title(paste('Digit ', index+9, sep=""))
  #dev.off()
}

#####
# Multiplicative Contaminated #
#####

contam <- seq(1.01, 1.50, .01)
nlist <- 1000*c(0.1, 1:5)
nt <- 1:length(nlist)
Power.MC <- matrix(0, nrow=length(contam), ncol=length(nlist))

par(mfrow=c(3,3))

for (index in 82:90)
{
  for (nin in 1:length(nlist))
  {
    for (conin in (1:length(contam)))
    {
      cps <- Ben.ps(digits)
      cps[index] <- min(1, cps[index]*contam[conin])
      cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
      mu <- sqrt(nlist[nin])*(Ben.ps(digits)-cps)
      psi2 <- mu%*%diag(1/Ben.ps(digits))%*%mu
      Power.MC[conin, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)
    }
  }
  #jpeg(paste(mdir,"RBen/Benplots/MulCon-dig", index+9, ".jpeg", sep=""))
  matplot(contam, Power.MC, type='l', lty=nt, col=nt, ylim=c(0,1),
ylab='Power', xlab='Contamination')
  #legend(.1, .4, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)

```

```

        title(paste('Digit ', index+9, sep=""))
        #dev.off()
    }

#####
# Generalized Benford #
#####

alpha <- c(seq(-1, -.1, .1), seq(.1, 1, .1))
nlist <- 1000*c(0.1, 1:5)
Power.GB <- matrix(0, nrow=length(alpha), ncol=length(nlist))

for (nin in 1:length(nlist))
{
    for (ai in (1:length(alpha)))
    {
        ps <- genben.ps(alpha, digits)
        mu <- sqrt(nlist[nin])*(Ben.ps(digits)-ps)
        BinvAmu <- as.vector(BinvA%%mu)

        mu2.W <- as.vector(eigv.W%%BinvAmu)^2
        Power.GB[ai, nin] <- Imhofnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

        #mu2.U <- as.vector(eigv.U%%BinvAmu)^2
        #Power.GB[ai,nin] <- Imhofnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

        #mu2.A <- as.vector(eigv.A%%BinvAmu)^2
        #Power.GB[ai,nin] <- Imhofnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

        #psi2 <- mu%%diag(1/Ben.ps(digits))%%mu
        #Power.GB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

        dimnames(Power.GB)[[2]]<-paste("n=", nlist, sep=")
    }
}

nt <- 1:length(nlist)
matplot(alpha, Power.GB, type='l', lty=nt, col=nt, ylim=c(0,1),
        ylab='Power', xlab='Alpha, parameter')

title('Power for W2 under Generalized Benford')
legend(0.35, 0.40, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

#title('Power for U2 under Generalized Benford')
#legend(0.375, 0.35, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

#title('Power for A2 under Generalized Benford')
#legend(0.35, 0.40, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

#title('Power for X2 under Generalized Benford')
#legend(0.35, 0.60, legend=paste("n=", nlist, sep="), lty=nt, col=nt)

```

## ApproxSimulated.R

```
#####
# Approximated Power vs Simulated Power
#####

# files directory
mdir <- "F:/Stanley/Thesis/Results/2digits/"

# length of multinomial vector
digits <- 10:99

# Number of replications
N <- 1000

# Number of sample sizes for each replication
n <- 1000

# load BenNonCentral, BenSimfcns and BenSummary2 codes
source("F:/Stanley/Thesis/Codes/SW_BenSimus.R")
source("F:/Stanley/Thesis/Codes/SW_BenNonCentral.R")
source("F:/Stanley/Thesis/Codes/SW_BenSimfcns.R")
source("F:/Stanley/Thesis/Codes/SW_BenSummary2.R")
source("F:/Stanley/Thesis/Codes/SW_MyErrorBar.R")

# Compute eigenvalues required for approx to SL's for CVM stats
CVME <- CVMEigen(n, digits)
CR.eig.W <- CVME$eig.W
CR.eig.U <- CVME$eig.U
CR.eig.A <- CVME$eig.A

# approximate critical values for CVM tests
Wcrit <- Chinv.approx(.95, CR.eig.W)
Ucrit <- Chinv.approx(.95, CR.eig.U)
Acrit <- Chinv.approx(.95, CR.eig.A)
Xcrit <- qchisq(.95, length(digits)-1)

noncen <- CVMPower(digits)
eigv.W <- noncen$eigv.W
eigv.U <- noncen$eigv.U
eigv.A <- noncen$eigv.A
BinvA <- noncen$BinvA

#####
# Hill/Benford #
#####

# Simulated Power Under Mixture Hill/Benford

fnames <- shnames <- plotnames <- vector("character", 0)
for(alpha in c(seq(.1, .5, .1)))
{
  fnames <- c(fnames, paste(mdir, "RBen/RHilBen", alpha, ".n", n, ".N", N, ".txt", sep=""))
  shnames <- c(shnames, paste("HilBen", alpha, sep=""))
}
```

```

      plotnames <- c(plotnames, paste(mdir, "RBen/RHilBen", alpha, ".n", n, ".N", N, ".jpeg",
sep=""))
}
RES.HB <- BenSummary2(fnames, shnames, plotnames)
### Extract only W2 and A2 for Mixtures ###
RES.HB.CVM <- as.vector(RES.HB[c(5,9),])
### width of power proportions (2*sqrt(margin of errors)) ###
RES.HB.w <- 2*sqrt(RES.HB.CVM*(1-RES.HB.CVM)/N)

# Approximated Power Under Mixture Hill/Benford

alpha <- seq(.05, .5, .05)
Power.HB <- matrix(0, nrow=length(alpha), ncol=2)
Hill <- c(0.0085, 0.0156, 0.0172, 0.0160, 0.0154, 0.0147, 0.0165, 0.0188, 0.0107, 0.0135,
0.0058, 0.0106, 0.0117, 0.0109, 0.0105, 0.0100, 0.0112, 0.0128, 0.0073, 0.0092,
0.0060, 0.0110, 0.0122, 0.0113, 0.0109, 0.0104, 0.0116, 0.0133, 0.0076, 0.0096,
0.0077, 0.0141, 0.0156, 0.0145, 0.0140, 0.0133, 0.0149, 0.0170, 0.0097, 0.0122,
0.0056, 0.0103, 0.0113, 0.0106, 0.0102, 0.0097, 0.0109, 0.0124, 0.0071, 0.0089,
0.0091, 0.0166, 0.0184, 0.0171, 0.0165, 0.0157, 0.0176, 0.0201, 0.0115, 0.0144,
0.0070, 0.0127, 0.0140, 0.0131, 0.0126, 0.0120, 0.0134, 0.0154, 0.0088, 0.0110,
0.0049, 0.0089, 0.0098, 0.0092, 0.0088, 0.0084, 0.0094, 0.0108, 0.0061, 0.0077,
0.0034, 0.0061, 0.0068, 0.0063, 0.0061, 0.0058, 0.0065, 0.0074, 0.0042, 0.0053)

for (ai in (1:length(alpha)))
{
  ps <- alpha[ai]*Hill + (1-alpha[ai])*Ben.ps(digits)
  mu <- sqrt(n)*(Ben.ps(digits)-ps)
  BinvAmu <- as.vector(BinvA%*mu)

  mu2.W <- as.vector(eigv.W%*BinvAmu)^2
  Power.HB[ai, 1] <- lmhofnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

  #mu2.U <- as.vector(eigv.U%*BinvAmu)^2
  #Power.HB[ai, nin] <- lmhofnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

  mu2.A <- as.vector(eigv.A%*BinvAmu)^2
  Power.HB[ai, 2] <- lmhofnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

  #psi2 <- mu%*diag(1/Ben.ps(digits))%*mu
  #Power.HB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

  dimnames(Power.HB)[[2]] <- c('W2', 'A2')
}

alphaS <- seq(.1, .5, .1)
matplot(alpha, Power.HB, type='l', lty=1:2, col=1:2, ylim=c(0,1),
  ylab='Power', xlab='Alpha, mixing proportion')

myerrbar(x=rep(seq(.1, .5, .1), rep(2, 5)),
  y=RES.HB.CVM, yplus=RES.HB.CVM+RES.HB.w, yminus=RES.HB.CVM-RES.HB.w,
  add=TRUE,
  lty=1:2,col=1:2)

#myerrbar(x=(rep(alphaS, rep(2,length(alphaS)))+rep(c(0,0.005), length(alphaS))),
# y=RES.HB.CVM, yplus=RES.HB.CVM+RES.HB.w, yminus=RES.HB.CVM-RES.HB.w,
# add=TRUE,

```

```

# lty=1:2,col=1:2)

title(paste('Power for Mixed Hill/Benford, n=', n, sep=''))
legend(0.35, 0.40, legend=c('W2', 'A2'), lty=1:2, col=1:2)

#####
# Uniform/Benford #
#####

# Simulated Power Under Mixture Uniform/Benford

fnames <- shnames <- plotnames <- vector("character", 0)
for(alpha in c(seq(.1, .5, .1)))
{
  fnames <- c(fnames, paste(mdir, "RBen/RUniBen", alpha, ".n", n, ".N", N, ".txt", sep=''))
  shnames <- c(shnames, paste("RUniBen", alpha, sep=''))
  plotnames <- c(plotnames, paste(mdir, "RBen/RUniBen", alpha, ".n", n, ".N", N, ".jpeg",
sep=''))
}
RES.UB <- BenSummary2(fnames, shnames, plotnames)
### Extract only W2 and A2 for Mixtures ###
RES.UB.CVM <- as.vector(RES.UB[c(5,9),])
### width of power proportions (2*sqrt(margin of errors)) ###
RES.UB.w <- 2*sqrt(RES.UB.CVM*(1-RES.UB.CVM)/N)

# Approximated Power Under Mixture Uniform/Benford

alpha <- seq(.05, .5, .05)
Power.UB <- matrix(0, nrow=length(alpha), ncol=2)

for (ai in (1:length(alpha)))
{
  ps <- alpha[ai]/length(digits) + (1-alpha[ai])*Ben.ps(digits)
  mu <- sqrt(n)*(Ben.ps(digits)-ps)
  BinvAmu <- as.vector(BinvA%*%mu)

  mu2.W <- as.vector(eigv.W%*%BinvAmu)^2
  Power.UB[ai, 1] <- lmfnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

  #mu2.U <- as.vector(eigv.U%*%BinvAmu)^2
  #Power.UB[ai, nin] <- lmfnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

  mu2.A <- as.vector(eigv.A%*%BinvAmu)^2
  Power.UB[ai, 2] <- lmfnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

  #psi2 <- mu%*%diag(1/Ben.ps(digits))%*%mu
  #Power.UB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

  dimnames(Power.UB)[[2]] <- c('W2', 'A2')
}

matplot(alpha, Power.UB, type='l', lty=1:2, col=1:2, ylim=c(0,1),
  ylab='Power', xlab='Alpha, mixing proportion')

myerrbar(x=rep(seq(.1, .5, .1), rep(2, 5)),

```

```

y=RES.UB.CVM, yplus=RES.UB.CVM+RES.UB.w, yminus=RES.UB.CVM-RES.UB.w,
add=TRUE,
lty=1:2,col=1:2)

#myerrbar(x=(rep(alphaS, rep(2,length(alphaS)))+rep(c(0,0.005), length(alphaS))),
# y=RES.HB.CVM, yplus=RES.HB.CVM+RES.HB.w, yminus=RES.HB.CVM-RES.HB.w,
add=TRUE,
# lty=1:2,col=1:2)

title(paste('Power for Mixed Uniform/Benford, n=', n, sep=""))
legend(0.35, 0.40, legend=c('W2', 'A2'), lty=1:2, col=1:2)

#####
# Additive Contaminated #
#####

# Simulated Power Under Additive Contaminated

contam <- seq(.01, .15, .01)
contam.list <- c(.02, .06)
nlist <- 1000*c(1, 2)
nt <- 1:length(nlist)
Power.AC <- matrix(0,nrow=length(contam), ncol=length(nlist))

par(mfrow=c(3,3))

for (index in 82:90)
{
  RES.AC.CVM <- matrix(0, nrow=2, ncol=length(contam.list))
  for (nin in 1:length(nlist))
  {
    fnames <- shnames <- plotnames <- vector("character", 0)
    for (i in 1:length(contam.list))
    {
      fnames <-
c(fnames,paste(mdir,"RBen/RConAddBen",index+length(digits)/9-1,"i",contam.list[i], ".n",
nlist[nin], ".N",N,".txt",sep="))
      shnames <- c(shnames,paste("RConAddBen",index+length(digits)/9-
1,"i",contam.list[i],sep="))
      plotnames <-
c(plotnames,paste(mdir,"RBen/RConAddBen",index+length(digits)/9-
1,"i",contam.list[i], ".n",nlist[nin], ".N",N,".jpeg",sep="))
    }
    RES.AC <- BenSummary2(fnames,shnames,plotnames)
    # Extract only chi-square statistic ##
    RES.AC.CVM[nin,] <- as.vector(RES.AC[11,])
  }
  RES.AC.CVM <- as.vector(RES.AC.CVM)
  RES.AC.w <- 2*sqrt(RES.AC.CVM*(1-RES.AC.CVM)/N)

  for (nin in 1:length(nlist))
  {
    for (conin in (1:length(contam)))
    {
      cps <- Ben.ps(digits)

```

```

      cps[index]      <- cps[index]+contam[conin]
      cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
      mu <- sqrt(nlist[nin])*(Ben.ps(digits)-cps)
      psi2 <- mu%*%diag(1/Ben.ps(digits))%*%mu
      Power.AC[conin, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)
    }
  }
  #jpeg(paste(mdir,"RBen/Benplots/AddCon-dig", index+9, ".jpeg", sep="))
  matplot(contam, Power.AC, type='l', lty=nt, col=nt, ylim=c(0,1),
  ylab='Power', xlab='Contamination')
  #legend(.1, .4, legend=paste("n=", nlist, sep=""), lty=nt, col=nt)
  title(paste('Digit ', index+9, sep="))
  myerrbar(x=rep(contam.list,rep(2,length(contam.list))),
  y=RES.AC.CVM, yplus=RES.AC.CVM+RES.AC.w, yminus=RES.AC.CVM-
RES.AC.w, add=TRUE,
  cap=.03, lty=1,col=1:2, lwd=2)
  #dev.off()
}

#####
# Multiplicative Contaminated #
#####

# Simulated Power Under Additive Contaminated

contam <- seq(1.01, 1.50, .01)
contam.list<-c(1.20,1.50)
nlist <- 1000*c(1:2)
nt <- 1:length(nlist)
Power.MC <- matrix(0, nrow=length(contam), ncol=length(nlist))

par(mfrow=c(1,1))

for (index in 82:90)
{
  RES.MC.CVM <- matrix(0, nrow=2, ncol=length(contam.list))
  for (nin in 1:length(nlist))
  {
    fnames <- shnames <- plotnames <- vector("character", 0)
    for (i in 1:length(contam.list))
    {
      fnames <- c(fnames, paste(mdir, "RBen/RConMultiBen",
index+length(digits)/9-1, "i", contam.list[i], ".n", nlist[nin], ".N", N, ".txt", sep="))
      shnames <- c(shnames, paste("RConMultiBen", index+length(digits)/9-1,
"i", contam.list[i], sep="))
      plotnames <- c(plotnames, paste(mdir, "RBen/RConMultiBen",
index+length(digits)/9-1, "i", contam.list[i], ".n", nlist[nin], ".N", N, ".jpeg", sep="))
    }
    RES.MC <- BenSummary2(fnames, shnames, plotnames)
    # Extract only chi-square statistic ##
    RES.MC.CVM[nin,] <- as.vector(RES.MC[11,])
  }
  RES.MC.CVM <- as.vector(RES.MC.CVM)
  RES.MC.w <- 2*sqrt(RES.MC.CVM*(1-RES.MC.CVM)/N)

  for (nin in 1:length(nlist))

```

```

{
  for (conin in (1:length(contam)))
  {
    cps <- Ben.ps(digits)
    cps[index] <- min(1, cps[index]*contam[conin])
    cps[-index] <- cps[-index]*(1-cps[index])/sum(cps[-index])
    mu <- sqrt(nlist[nin])*(Ben.ps(digits)-cps)
    psi2 <- mu%%diag(1/Ben.ps(digits))%%mu
    Power.MC[conin, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)
  }
}
#jpeg(paste(mdir,"RBen/Benplots/MulCon-dig", index+9, ".jpeg", sep="))
matplot(contam, Power.MC, type='l', lty=nt, col=nt, ylim=c(0,1),
ylab='Power', xlab='Contamination')
#legend(.1, .4, legend=paste("n=", nlist, sep="), lty=nt, col=nt)
title(paste('Digit ', index+9, sep="))
myerrbar(x=rep(contam.list,rep(2,length(contam.list))),
          y=RES.MC.CVM, yplus=RES.MC.CVM+RES.MC.w, yminus=RES.MC.CVM-
RES.MC.w, add=TRUE, lty=1:2,col=1:2 )

#dev.off()
}

#####
# Generalized/Benford #
#####

# Simulated Power Under Generalized Benford

fnames <- shnames <- plotnames <- vector("character", 0)
for (alpha in c(seq(-1, -.1, .1), seq(.1, 1, .1)))
{
  fnames <- c(fnames, paste(mdir, "RBen/RGenBen", alpha, ".n", n, ".N", N, ".txt", sep="))
  shnames <- c(shnames, paste("RGenBen", alpha, sep="))
  plotnames <- c(plotnames, paste(mdir, "RBen/RGenBen", alpha, ".n", n, ".N", N, ".jpeg",
sep="))
}
RES.GB <- BenSummary2(fnames, shnames, plotnames)
### Extract only W2 and A2 for Mixtures ###
RES.GB.CVM <- as.vector(RES.GB[c(5,9),])
### width of power proportions (2*sqrt(margin of errors)) ###
RES.GB.w <- 2*sqrt(RES.GB.CVM*(1-RES.GB.CVM)/N)

# Approximated Power Under Mixture Uniform/Benford

alpha <- c(seq(-1, -.1, .1), seq(.1, 1, .1))
Power.GB <- matrix(0, nrow=length(alpha), ncol=2)

for (nin in 1:length(nlist))
{
  for (ai in (1:length(alpha)))
  {
    ps <- genben.ps(alpha[ai], digits)
    mu <- sqrt(n)*(Ben.ps(digits)-ps)
    BinvAmu <- as.vector(BinvA%%mu)
  }
}

```



```

mu2.W <- as.vector(eigv.W%*%BinvAmu)^2
Power.GB[ai, 1] <- lmfhofnc(Wcrit, CR.eig.W, mu2.W, UPPER=Inf, subdiv=500)

#mu2.U <- as.vector(eigv.U%*%BinvAmu)^2
#Power.GB[ai,nin] <- lmfhofnc(Ucrit, CR.eig.U, mu2.U, UPPER=Inf, subdiv=500)

mu2.A <- as.vector(eigv.A%*%BinvAmu)^2
Power.GB[ai, 2] <- lmfhofnc(Acrit, CR.eig.A, mu2.A, UPPER=Inf, subdiv=500)

#psi2 <- mu%*%diag(1/Ben.ps(digits))%*%mu
#Power.GB[ai, nin] <- 1-pchisq(Xcrit, length(digits)-1, psi2)

dimnames(Power.GB)[[2]] <- c('W2', 'A2')
}
}

matplot(alpha, Power.GB, type='l', lty=1:2, col=1:2, ylim=c(0,1),
        ylab='Power', xlab='Alpha, parameter')

title(paste('Power for Generalized Benford, n=', n, sep=''))
legend(0.35, 0.40, legend=c('W2', 'A2'), lty=1:2, col=1:2)

myerrbar(x=rep(alpha, rep(2,20)),
        y=RES.GB.CVM, yplus=RES.GB.CVM+RES.GB.w, yminus=RES.GB.CVM-RES.GB.w,
        add=TRUE,
        lty=1:2,col=1:2)

```