A MODELING APPROACH TO SYSTEM EVALUATION IN RESEARCH DATA MANAGEMENT

by

Juliana Mei-Mei Ma

Department of Biostatistics
University of North Carolina at Chapel Hill

Institute of Mimeo Series No. 1822T

December 1986
A MODELING APPROACH TO SYSTEM EVALUATION
IN RESEARCH DATA MANAGEMENT

by

Juliana Mei-Mei Ma

A Dissertation submitted to the faculty of The University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics.

Chapel Hill

1986

Approved by:

[Signature]
Advisor

[Signature]
Reader
JULIANA MEI-MEI MA. A Modeling Approach to System Evaluation in Research Data Management. (Under the direction of Ronald Helms.)

A modeling approach is presented for evaluating error rates and the cost-effectiveness of alternative components of research data management (RDM) systems for small-scale research studies. A flexible model framework is developed for several edit methods; manual proofreading, redundant data entry with or without redundant data collection, and univariate valid value edit tests are considered. Six models are created and the most interesting are investigated in depth. Examples are provided of how the modeling approach is applied and how model predictions compare with actual results. The goal is to propose a workable basis for a cost/benefit analysis of realistic RDM systems.

The focus is on modeling and comparison of systems for studies that involve relatively simple data fields, and are too small to justify development of a sophisticated, predominantly computerized, RDM system. An example of a target study is a telephone survey of 1000 people, who are asked twenty multiple-choice questions, for which all RDM and analysis tasks are completed within six months. In contrast, cost/benefit models typically found in the literature are based on the model proposed by O'Reagan for evaluating large scale studies at the U.S. Census Bureau.

The probability model framework is outlined by presentation of a basic model. Using a record as the measurement unit, as opposed to a data field, estimators for error generation and RDM system cost are defined. Parameters related to error creation, error detection, development cost, and processing cost are identified. Error generation is
investigated using the results of probability derivations. A numerical example of comparing two alternative RDM systems demonstrates the use of the modeling approach.

Five other models, created by extending the basic model, are presented and investigated. One model includes several error detection processes after data entry. Another addresses the possibility of redundant data collection. Explorations of the models provide cost/benefit analysis examples, general conclusions about the relative importance of error generation parameters, and insights into the comparative effectiveness of specific RDM systems. Appropriate ranges for required parameter values are based on reports in the literature and results from studies in which RDM systems were implemented. Several pairs of RDM alternatives are compared to illustrate the utility of the modeling approach. A statistic, called the cost-error ratio, is proposed that provides information for making informed cost/benefit decisions.

The results of experimental studies demonstrate the validity of model predictions. The expected number of records with errors, after complete implementation of a RDM system, is compared with actual results. For one example, the data were divided in half and both RDM alternatives were implemented. The actual error rate and cost difference results support the cost/benefit decision suggested by model estimates.

The modeling approach provides tractable models that provide quantitative evidence for making informed decisions about RDM system alternatives. In addition, we find that redundancy should be seriously considered as a RDM principle. Redundant processing produces better quality data under a wide variety of circumstances and is often cost-effective, assuming errors are corrected when discovered after data processing is considered finished.
ACKNOWLEDGEMENTS

A completed dissertation fulfills an educational requirement for one person, but the creation process depends on many people. I wish to thank my advisor, Prof. Ron Helms, for his helpful advice during our many meetings. He always sent me on my way with a clearer idea of what to do next. I also wish to acknowledge the members of my committee: Dr. Keith Muller, Prof. Dick Shachtman, Dr. Lisa Morrissey LaVange, Dr. Jim Hosking, and Prof. Harvey Jeffries.

The support of the UNC Highway Safety Research Center made this research possible. The rapid adoption of microcomputers in the last few years provided resources and research data that proved invaluable. In addition, I greatly appreciate the tolerance for my unpredictable working schedule during the last two years. I thank Dr. Don Reinfurt for encouraging me to devote time to my own work instead of HSRC. Special thanks go to the HSRC programmers for uncountable reasons.

In this age of the microcomputer no typist was required, instead I thank a company with a vision, and a friend with an editorial eye for making the final version look professional. The development by Apple Computer of the Macintosh and the LaserWriter made the task of creating the words and diagrams enjoyable. I owe my discovery of the difference between rewrite and edit to Carol Leininger.

No one can spend eight years as a graduate student without family support. I cannot ever thank my parents, Prof. T. S. Ma and Dr. G. F. Ma, enough for everything. Their patience and support made the process more worthwhile. To my husband Nick, my loving thanks for staying on the road to let me work without distraction even when he was one very homesick trucker.

November 1986
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>ACKNOWLEDGEMENTS</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>v</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>viii</td>
</tr>
<tr>
<td>CHAPTER 1. INTRODUCTION AND LITERATURE REVIEW</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Statistical Models for RDM Systems</td>
<td>3</td>
</tr>
<tr>
<td>1.3 RDM System Requirements</td>
<td>8</td>
</tr>
<tr>
<td>1.4 Redundancy and Verification</td>
<td>10</td>
</tr>
<tr>
<td>1.5 Error Generation Rates</td>
<td>13</td>
</tr>
<tr>
<td>1.6 Inappropriate RDM Techniques</td>
<td>15</td>
</tr>
<tr>
<td>CHAPTER 2. PRESENTATION OF MODEL 1</td>
<td>17</td>
</tr>
<tr>
<td>2.1 Definition of Terms</td>
<td>17</td>
</tr>
<tr>
<td>2.2 Probability Structure</td>
<td>23</td>
</tr>
<tr>
<td>2.3 Error Generation Derivations</td>
<td>25</td>
</tr>
<tr>
<td>2.4 Cost Estimation</td>
<td>38</td>
</tr>
<tr>
<td>2.5 Numerical Examples</td>
<td>43</td>
</tr>
<tr>
<td>2.6 Discussion</td>
<td>50</td>
</tr>
<tr>
<td>CHAPTER 3. MODEL EXTENSIONS</td>
<td>51</td>
</tr>
<tr>
<td>3.1 Extended Model Background</td>
<td>51</td>
</tr>
<tr>
<td>3.2 Model 2 – Valid Value Edit in Primary Processing</td>
<td>54</td>
</tr>
<tr>
<td>3.3 Model 3 – Valid Value Edit in Primary and Secondary Processing</td>
<td>76</td>
</tr>
<tr>
<td>3.4 Model 4 – Valid Value Edit After Initial Processing</td>
<td>88</td>
</tr>
<tr>
<td>3.5 Model 5 – General Data Entry Model</td>
<td>92</td>
</tr>
<tr>
<td>3.6 Model 6 – Redundant Data Collection</td>
<td>111</td>
</tr>
<tr>
<td>CHAPTER 4. INVESTIGATION OF EXTENDED MODELS</td>
<td>125</td>
</tr>
<tr>
<td>4.1 System Evaluation</td>
<td>125</td>
</tr>
<tr>
<td>4.2 Reducing Model 5</td>
<td>126</td>
</tr>
<tr>
<td>4.3 Parameter Values</td>
<td>130</td>
</tr>
<tr>
<td>4.4 Error Generation</td>
<td>140</td>
</tr>
<tr>
<td>4.5 Cost Estimation</td>
<td>147</td>
</tr>
<tr>
<td>4.6 Subsystem Comparisons</td>
<td>148</td>
</tr>
</tbody>
</table>
## CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER 5. APPLICATIONS</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 RDM Environment</td>
<td>164</td>
</tr>
<tr>
<td>5.2 Model 1 Experiment</td>
<td>165</td>
</tr>
<tr>
<td>5.3 Using Redundancy and Batch VVE</td>
<td>170</td>
</tr>
<tr>
<td>5.4 Intelligent Data Entry, Redundancy, or Manual Proofread</td>
<td>174</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 6. CONCLUSIONS AND SUGGESTIONS</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Conclusions</td>
<td>176</td>
</tr>
<tr>
<td>6.2 Suggestions for Further Research</td>
<td>177</td>
</tr>
<tr>
<td>6.3 Redundancy as a RDM Principle</td>
<td>178</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REFERENCES</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>180</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GLOSSARY</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>184</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>APPENDIX A – MODEL NOTES</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count Parameters</td>
<td>188</td>
</tr>
<tr>
<td>Cost Parameters</td>
<td>189</td>
</tr>
<tr>
<td>Bernoulli Random Variables</td>
<td>190</td>
</tr>
<tr>
<td>Model 5 Count Parameters</td>
<td>191</td>
</tr>
<tr>
<td>Model 5 Theorems</td>
<td>192</td>
</tr>
<tr>
<td>Model 6 Count Parameters</td>
<td>194</td>
</tr>
<tr>
<td>Model 6 Theorems</td>
<td>195</td>
</tr>
<tr>
<td>Model 6 Probability Parameter Values</td>
<td>196</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>APPENDIX B – STUDY SUMMARY</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study Notes</td>
<td>197</td>
</tr>
<tr>
<td>Study Details</td>
<td>200</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Target Study RDM Characteristics</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>Model 1 Overall System Components</td>
<td>20</td>
</tr>
<tr>
<td>2.2</td>
<td>Model 1 Subsystems</td>
<td>21</td>
</tr>
<tr>
<td>2.3</td>
<td>Model 1 Overall System with Random Variables</td>
<td>24</td>
</tr>
<tr>
<td>2.4</td>
<td>Model 1 Probability Parameters and Bernoulli Random Variables</td>
<td>37</td>
</tr>
<tr>
<td>2.5</td>
<td>Model 1 Count Estimation Theorems and Parameter Values</td>
<td>42</td>
</tr>
<tr>
<td>2.6</td>
<td>Model 1 Final Error Rates</td>
<td>45</td>
</tr>
<tr>
<td>2.7</td>
<td>Comparison of Model 1 Final Error Rates</td>
<td>46</td>
</tr>
<tr>
<td>2.8</td>
<td>Model 1 Cost Parameter Values</td>
<td>48</td>
</tr>
<tr>
<td>2.9</td>
<td>Model 1 Subsystem Comparisons</td>
<td>49</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Model 2 Overall System with Random Variables</td>
<td>55</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Model 2 Subsystems</td>
<td>56</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Comparison of Model 1 and Model 2</td>
<td>57</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Model 2 Overall System Cost Components</td>
<td>70</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Model 3 Overall System with Random Variables</td>
<td>74</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Model 3 Subsystems</td>
<td>75</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Model 4 Overall System with Random Variables</td>
<td>86</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Model 4 Subsystems</td>
<td>87</td>
</tr>
</tbody>
</table>
FIGURES

3.5.1 Model 5 Overall System with Random Variables .................. 93
3.5.2 Model 5 Subsystems ........................................... 94
3.5.3 Model 5 Subsystem Probability Parameters ...................... 103
3.5.4 Model 5 Count Parameters ..................................... 104
3.5.5 Model 5 Overall System Cost Components ....................... 106
3.6.1 Model 6 Overall System with Random Variables ................ 109
3.6.2 Model 6 Subsystems ........................................... 110
3.6.3 Model 6 – Primary Processing Outcomes ....................... 113
3.6.4 Model 6 – Table of Error Outcomes .......................... 116
3.6.5 Model 6 Overall System Cost Components ..................... 123

4.1 Comparison of Model 2 and Model 5 .............................. 127
4.2 Comparison of Model 2 and Model 5 Probability Parameters .... 129
4.3 Count Parameter Values .......................................... 141
4.4 Model 5 Final Error Rates ....................................... 145
4.5 Comparison of Model 5 Final Error Rates for Eight Subsystems . 146
4.6 Model 5 – Subsystem 1 and Subsystem 2 ........................ 149
4.7 Model 5 – Subsystem 1 and Subsystem 3 ........................ 154
4.8 Model 5 – Selected Subsystems ................................ 158
4.9 Model 6 Final Error Rates ....................................... 162

5.1 Predicted Final Error Rates for BAC Study ...................... 167
5.2 Count Estimates for Redundancy and Primary VVE ............... 171
CHAPTER 1
INTRODUCTION AND LITERATURE REVIEW

An important aspect of research data management (RDM) is finding cost-effective methods for reducing errors. This research focuses on the use of modeling to investigate RDM methods that are appropriate for relatively small studies. A flexible model framework is developed from concepts introduced in work related to statistical models applicable to large scale operations, such as the U.S. Census. The goal is to present models for comparing undetected error rates and system costs for realistic RDM systems.

1.1 Introduction

This literature review presents work related to modeling RDM systems. Each section is organized chronologically within subject. The first section reviews research presenting statistical models for performing cost/benefit analysis or investigating error generation and detection in RDM. Subsequent sections cover RDM system design and error generation rates. A discussion of the effectiveness of different verification methods is included.

The following definitions apply in the review of the literature.

*Research data management system (RDMS)*: the complete RDM system required to create a study file ready for computer-assisted statistical analysis. This includes data collection, data entry, error detection, and update processes.
Redundancy: the duplication of information either by use of independent data sources or independent parallel processing. Emphasis is on duplication within a data processing system, as opposed to repetition of a question, or inclusion of a similar question, on the same data collection form.

Verification: the process of rekeying or re-coding data for error detection and correction purposes. Dependent verification implies that a verifier works with both the original information and previous processing results. With independent verification, the same process is repeated independently.

Automatic correction, imputation: the computer-aided process of correcting a missing or suspicious data value by automatic replacement to some "legitimate value." A computer program performs the replacement by following a well-defined algorithm.

We are mainly interested in designing models for studies with target characteristics outlined in Figure 1.1. The important RDM features of a target study include: 1) short-term duration (less than two years), 2) lack of a formal RDMS, and 3) the information to be processed consists of relatively simple data fields. An example of an appropriate study is a telephone survey of 1000 people, who are asked twenty questions, for which all RDM and analysis tasks are completed within six months.
### FIGURE 1.1

**Target Study RDM Characteristics**

<table>
<thead>
<tr>
<th>Factor</th>
<th>Target</th>
<th>Inappropriate</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDM system status</td>
<td>No existing formal RDMS</td>
<td>Existing RDMS</td>
</tr>
<tr>
<td>Study duration</td>
<td>Short term, up to 24 months</td>
<td>Long term, &gt; 2 years, on-going</td>
</tr>
<tr>
<td>Data entry</td>
<td>Non-professional or professional</td>
<td>Multiple site entry</td>
</tr>
<tr>
<td>Number of records</td>
<td>≤ 100,000, but mostly ≤ 20,000</td>
<td>&gt; 100,000</td>
</tr>
<tr>
<td>Number of forms</td>
<td>One or separate</td>
<td>Multiple, merged by ID</td>
</tr>
<tr>
<td>Number of fields</td>
<td>&lt; 50</td>
<td>&gt; 100</td>
</tr>
<tr>
<td>Processing flow</td>
<td>&quot;One shot&quot;</td>
<td>Repeated batches, complex inventory</td>
</tr>
<tr>
<td>Field types</td>
<td>Categorical</td>
<td>Long text (Address)</td>
</tr>
<tr>
<td></td>
<td>Simple ID</td>
<td>Numeric, &gt; 3 digits</td>
</tr>
<tr>
<td></td>
<td>Simple numeric (Age of Respondent)</td>
<td></td>
</tr>
</tbody>
</table>

### 1.2 Statistical Models for RDM Systems

There are few statistical models for cost/benefit analysis in RDM. In particular, no models have been designed for the target situation. However, techniques used in developing related models provide model design ideas. The discussion is divided into three parts. The first part covers cost/benefit models for RDM environments that process survey data. The second presents data error models for survey data. Decision models for management information or accounting data are discussed in the last part.

#### 1.2.1 Cost/benefit Models for Survey Data

O'Reagan (1969) outlined a cost/benefit analysis model applicable to very large RDM systems. The model was designed to help decide if a computer edit, which performed automatic correction on U.S. Census data, was cost-effective in comparison
to clerical (manual) methods. The basic concept was

\[ \text{Total cost} = \text{Inspection cost} + \text{Correction cost} + \text{Undetected error cost}. \]

Undetected error cost was defined as the cost of later correction, and assumed much smaller than the cost of automatic correction. Inspection cost incorporates overhead, planning, development, and processing costs.

Error rate and cost parameters were defined in order to calculate total cost estimates. The error rate parameters were:

\[ \alpha = \text{proportion of good items judged bad, Type I error rate}; \]
\[ \beta = \text{proportion of undetected bad items, Type II error rate}; \]
\[ d = \text{proportion of defective items, true error rate}. \]

The cost parameters included the number of items, \( N \), and average costs per item for inspection, correction, and undetected errors: \( I_i, C_i, U_i \). Initially, the model was simplified by ignoring differences between items. Total cost, \( T \), was expressed as

\[ T = I + C + U, \]

in which

\[ I = NI_i \quad \text{inspection cost}, \]
\[ C = [(1 - d)\alpha + d(1 - \beta)]NC_i \quad \text{correction cost}, \]
\[ U = [(1 - d)\alpha + d\beta]NU_i \quad \text{undetected error cost}. \]

Applications of the model demonstrated its usefulness. A numeric example showed automatic correction can be cost-effective; the cost estimates were based on \( N = 12,000,000 \) and parameter values from Census experience. Analytic applications of the model included the derivation of critical values for cost/error decisions, and investigation of the effects on cost of varying the error rate parameters.

O’Reagan concluded that, assuming complex models are not used in practice, a crude cost/benefit analysis was preferable to none at all. Possible extensions suggested for his model included varying \( U_i \) for different items, or allowing some manual review.
Minton (1972) expanded O'Reagan's cost model by defining separate parameters for inspection and correction error rates. Costs of computer and manual systems were compared for extreme situations (inspectors make no mistakes or misclassify 50%). The cost model was applied to a RDM environment at the U.S. Census Bureau, in which process control techniques were used to improve data quality. Average cost per lot was computed based on known costs per item for inspection, correction, non-detection, and later correction.

After investigating the expanded model, Minton concluded that cost analysis should include sensitivity analysis. The effect of parameter value inaccuracies was a problem with this model. However, despite problems associated with determining good parameter values, he felt that models were important tools for deciding between editing system alternatives.

Another cost/benefit model similar to O'Reagan's was developed by Richardson and Cunningham (1972). Their model differed in that fixed overhead cost was separate from inspection cost. A theoretical cost analysis was presented using parameter values based on data processing for the British Census of Production. An automatic correction system was compared with a system in which defectives were rejected and edited manually. The value for the proportion of defectives, \( d = 0.17 \), defined as the proportion "worth correction," was lower than the value of O'Reagan's parameter, \( d = 0.29 \), which was the proportion of "defective" items.

Naus (1975) based his discussion of cost-effectiveness on O'Reagan's model. Cost components and error parameters were presented for a model of an edit test on a single variate. Edit rate parameters were added for both internal (computer) and external (manual) correction processes. This expanded model addressed the possibility that a flagged item (an error detected by an edit test) can remain incorrect, or that a correct item can become incorrect after being flagged because it is updated improperly. The model
was tractable because some interdependent parameters did not need to be estimated individually.

Under specific RDM conditions, Naus derived critical decision levels and compared internal and external correction processes. For a deterministic edit test, the critical level was found by setting the probability that a correct item is judged an error, $\alpha$, to zero in the cost equation. An edit test was cost-effective if the fraction of items flagged was greater than the critical level. For a probabilistic edit test, the model provided a method for determining values of $\alpha$ for which the edit was cost-effective.

Although Naus discussed cost-effectiveness for complete edit systems, he did not present any system models. Two observations about expanding the single variate model were noted: 1) a single variate may have more than one edit test in a complete system, and 2) overall edit costs might be approximated by combining individual edit test costs.

1.2.2 Error Detection Models for Survey Data

The problem of error detection was addressed by Naus, Johnson, and Montalvo (1972) using a probability model. A measure of suspicion was proposed and estimated using elementary probability theory arguments. The idea was to estimate the conditional probability that a variate is in error, given that a multivariate constraint involving the variate is violated. A general model was developed was based on the probability structure presented for modeling a univariate edit test. Sensitivity analysis was discussed for the special case in which independence of recording errors is assumed.

Helms (1981) proposed a stochastic model of data errors to investigate the potential of "total redundancy" in a RDMS. The model allowed for additive error and replacement error (transposition, incorrect update). After every processing stage, a data value was expressed as a function of its previous value and the error possibilities. Separate binominal distributions were proposed for each error type. Although the model proved intractable even under extreme independence assumptions, conclusions about
error generation could be drawn. Assuming that error occurrence probabilities are small, the model provided a method for demonstrating analytically that most data values are processed without errors.

1.2.3 Decision Models for Business Data

Although business or accounting data processing systems differ from RDM systems, the development of an appropriate statistical model involves similar problems. Recent applications of decision modeling to auditing procedures are of particular interest.

Varley (1969) developed a model framework for examining management information input errors. He was interested in the cost-effectiveness of manual processing methods for reducing errors in large on-going management information systems. Detailed classifications were given of data types, error types, fixed cost, and variable cost components of a detection system. Errors were discussed in relation to error sources (collector, recorder), error detectability (known, unknown), and error priorities (important, ignorable). Known errors, defined as detectable errors, had estimable error rates that depended on whether an error was correctable or uncorrectable, while unknown (undetectable) error rates were considered immeasurable. He proposed estimating a true item error rate using the known error detection rate and a detectability coefficient, which indicated the effectiveness of a valid value edit test in detecting an error. The detectability coefficient for a data item was estimated by considering the ratio of the number of valid codes and the number of possible codes, which depended on item length. Two other measures of detection effectiveness, reliability and correction effectiveness, were proposed:

\[
\text{reliability} = \frac{\text{number of item errors detected}}{\text{number of items validated}},
\]

\[
\text{correction effectiveness} = \frac{\text{number of item errors corrected}}{\text{number of item errors detected}}.
\]
Models of control systems for reducing errors in accounting transactions, concentrating on processing one monetary data item, were proposed by Baber (1980) and Knehel (1981). Baber used Bayesian revision in a decision model to compute posterior error distributions, given prior distributions of errors in recorded values. Assuming processing costs and error distributions were definable, optimal cost policies were derived for four common error types: 1) uniform distribution (n possible values), 2) uniform distribution for each digit, 3) transposition, and 4) column shift. Simple probability arguments were sufficient to define error distributions because of data type restrictions. Knehel proposed a general model of error generation that expanded on Baber's ideas. Different accounts, transactions, error types, and processing stages were allowed. However, the model was intractable unless independence of all error probabilities was assumed. Computer simulations were necessary to investigate more realistic situations; transactions were generated and “processed” according to appropriate error generation and correction probabilities.

1.3 RDM System Requirements

This section highlights general RDMS design principles; only the most important features of a RDMS are discussed. We are concerned with ideas applicable to the target situation.

One of the earlier applications of computer-aided editing to survey data was done at the Bureau of Labor Statistics (BLS). Stuart (1966) discussed the advantages of computerized procedures over manual review. He stressed that machine, clerical, and professional review activities must be well integrated. The BLS experience suggested that computer-aided error detection procedures must be clearly defined, and the effectiveness of edit criteria should be reviewed during actual processing.

Naus (1975) provided a thorough treatment of standard editing and quality control procedures. He presented a method for organizing a set of deterministic edit tests by
creating a dictionary of tests. Several techniques for defining probabilistic edit tests were provided. Complex edit methods presented included process control, automatic correction, and multivariate edit tests. Standard edit methods are not always appropriate for our target situation because of the time and expense required for design and implementation.

A concise definition of seven basic areas of computerized research data handling was given by Horwich (1977) in his discussion of interactive data entry for clinical studies. The seven areas identified were: coding, entry, error control, correction, restructure, retrieval, and analysis. Research data were divided into four main types: single digit numeric, numeric (2 – 5 digits), alphanumeric (fixed or variable length), and calendar dates.

Project planning was presented by Helms (1978) in an overview intended for a novice in RDM problems. The major phases of planning a RDM project were outlined, followed by more detailed discussion of initial planning tasks, pilot study activities, and data management system implementation.

Woolson, Tsuang, and Urban (1980) described their RDM experiences with an epidemiological study begun in 1973. Data entry for the study was initially done with punch cards. They found that forms design should include consideration of 1) the ease of use during coding, 2) the amount of space (column width) required for all data, and 3) later analyses. RDMS features that were important included:

1. Keypunching and verification method
2. Data editing system
3. Backup system
4. Master file organization
5. Procedures to create analysis subfiles
6. Statistical reporting system

Christiansen and Hosking (1982) presented alternative strategies for error handling procedures for studies in which a significant time lapse exists between error detection and final resolution. This situation may occur if correction procedures depend
on manual review. The strategies considered as the basis for designing error detection and resolution procedures were:

1. Do nothing
2. Use special missing values [codes]
3. Reject entire record
4. Use clean/dirty files
5. Use status variables

Based on seven design criteria, the first three strategies were deemed unacceptable. They felt both remaining strategies were reasonable, but neither was an ideal choice for all circumstances.

1.4 Redundancy and Verification

The quality of research data is often improved by some form of verification. Although few references explicitly consider redundancy as a general RDM edit method, several serve to illustrate its potential advantages over standard edit methods. In other fields, accounting and demography, methods similar to redundancy are used for improving the quality of information.

1.4.1 Theoretical Utility of Redundancy

Coale and Stephan (1962) detailed an extreme case of errors undetected by standard edit procedures. The 1950 Census contained unreasonable numbers of widowed and divorced white males in the 14 – 18 age bracket. There were also excess numbers of Indians in the 10 – 14 and 20 – 24 age brackets. They traced the problem to shift errors (data found one column over). In theory, an advantage of using a redundancy RDMS (redundant independent data entry) is that shift errors can be detected.

In data processing, redundancy is accepted intuitively as a logical method for reducing errors. Stuart (1966) stated that matched sample editing was "advantageous in isolating punching errors" (p. 379). Varley (1969) suggested that, if redundant data are available, correction procedures have a different priority because recovery of correct
information is easier. Rosen and Kanciruk (1985) believed comparing data entry results from two different people was a good method for detecting errors.

Redundancy, by the use of independent data sources, has been used to study data validity. Bahr (1971) and Katosh (1981) compared interview responses with official records to identify potential errors. Young (1972) presented the use of cross-tabulations to compare information collected from several sources about medical patient symptoms.

The need for redundancy during data collection is illustrated by a study conducted by Schreiber (1975). He studied the reliability of "invariant" variables (age of respondent, education level, father's occupation). The information came from interviews, in the United States and Britain, which were repeated after a two or three year interval. The measures of these "achieved" characteristics were up to thirty percent unreliable.

Clinical studies have found redundancy a useful quality control method. In an overview of the Mayo clinic, O'Fallon et al. (1980) mentioned that study assistants used redundant information from other forms to manually check research data before data entry. Selvey, Olubas, and Roessner (1980) described a RDMS for a clinical study that included independent duplicate data entry. The error detection process was begun by comparing the two data entry files. Non-matches were resolved before further edits were performed.

Redundancy as a RDM principle was discussed by Helms (1981). He proposed using "total redundancy" as an error detection method, in the sense of parallel (duplicate) RDM processing steps. The weaknesses of using partially redundant information, on which other error detection methods are based, were presented. An advantage of using "total redundancy" is that potentially all errors can be detected, including recording and transposition errors that result in "valid" information.

Redundancy is also used for error analysis. Norton (1981) presented a detailed analysis of data processing errors found by triple entry of sampled records. Potential
error patterns were identified by Goodyear (1983) after duplicate entry of sampled data. Sampling reduces the cost of redundant processing.

1.4.2 Dependent Verification Versus Independent Verification

The U.S. Census Bureau has performed several studies on the effectiveness of various verification methods. Dependent and independent verification have been used for data entry and coding. Dependent verification has been found less effective than independent verification for both data entry and coding of Census data. Fasteau, Ingram, and Minton (1964), Minton (1969), and Linebarger (1978) all concluded that independent verification was better for coding. Three-way independent verification, in which majority rules, was the optimal method for maintaining low error rates at a reasonable cost. Minton presented examples of high undetected error rates associated with dependent verification for a variety of data types. For a study in 1963 that used keypunched data cards, 14% of the defective (error) cards remained after dependent verification. His investigation demonstrated that independent verification was effective in detecting errors, even for difficult typing that required rounding and coding. The only drawback seen for large scale production was the additional cost.

Nemanich (1972) performed an experiment with survey data to test the effectiveness of dependent verification. She used four types of survey forms (about 300 forms each), which were composed of coded or pre-coded information from long or short forms. Half of the forms entered were verified using dependent verification. All cards were processed by a computer edit program that performed univariate and multivariate edit tests. Keying errors were found by investigating cards with detected errors. She concluded that, if other edit methods are also used, dependent verification was not necessarily worth the additional cost.
1.4.3 Redundancy in Other Fields

Double-entry bookkeeping is recognized as a basic accounting practice for all but the smallest businesses. The technique is characterized by duplicate recording of one monetary transaction as both a debit and a credit. A single-entry system is considered incomplete accounting. Double-entry has been used since the fourteenth century; Pacioli presented the first theoretical treatment in 1494. Recently, Ijiri (1982) suggested expanding the concept to consider multiple-entry bookkeeping methods.

The dual record system (DRS) is a method used in demography to improve vital rate estimates by using redundant information. Marks, Seltzer, and Krotki (1974) presented a detailed treatment of the method. A DRS is based on gathering information from two independent sources about events (births, deaths) that occurred in the same area during the same time period. The advantage of using a DRS is that the estimated rate adjusts for incomplete coverage and reduces biases. Extending a DRS to a multiple record system has been suggested (Yesilcay, 1975).

1.5 Error Generation Rates

The literature provides error and detection rates, which are required for any model of error generation and correction processes. Both actual values and measurement methods are of interest.

Error rates were reported for the 1940 U.S. Population Census. Information was keypunched from multiple-record forms onto 45-column cards. Deming and Geoffrey (1941) reported error rates as the number of wrong cards per hundred; these rates were measured weekly. During training, error rates were as high as 7/100. A good production keypuncher could have less than one wrong card per hundred cards and work for several weeks with error rates less than 2/100. Poor production keypunchers averaged four errors per hundred cards or had highly variable card error rates between 1/100 and 4/100. Deming, Tepping, and Geoffrey (1942) stated that previous dependent verifi-
cation systems detected errors in 0.5 – 5.0% of the cards verified. The overall card error rate was 1.5% during the sampling period of their study.

Deming et al. (1942) did a detailed study of 25,000 cards with at least one item error. Five samples, each containing 5000 error cards, were selected. On average, an error card had 1.47 errors; 86% had only one error, 9% two, and 5% three or more. Multiple errors on one card were often related since they could be traced to the placement of a data value in an incorrect column. For the 3535 multiple error cards, 24% included transposition errors. More errors occurred in non-standard records (supplementary schedule lines) or codes (non-white race, non-U.S. citizenship).

Error rates have been studied and reported for a wide variety of data types. Klemmer (1962) studied errors made by business keypunchers. Robison and Richardson (1978) included a table of error rates for different error types (same figure, switched, shift, respondent error) in their description of the editing and imputation of the 1977 Truck Inventory and Use Survey. VanGrevenhof (1983) reported error rates for the Mayo Clinic RDM system for monitoring patient flow in a clinical group medical practice.

The magnitude of error rates is very sensitive to the measurement unit used. Nemanich (1972) reported error rates per stroke and per card for her study of keypunch verification, which involved four sets of data. Two sets had one card per case, while the other two had multiple cards per case. The single card data had stroke error rates between 0.011% and 0.016%; the corresponding card error rate was about one percent. Stroke error rates for the more complex data were about 0.075%, for card error rates of six percent. Goldman and Jones (1978) presented item error rates in two ways for their study of data submitted by different clinics on a one-page form. Acceptable error rates were 20 – 30 item errors per hundred forms, or 0.5 errors per 100 entries. Item error rates for a 36-week period were initially around 100 errors per hundred forms, and eventually were consistently below 50 errors per hundred forms.
O'Fallon et al. (1980) provided error rates from a quality control study done for a clinical trial involving multiple clinics. Coding and data entry were performed by study assistants who were thoroughly familiar with the research data. Based on 15,841 items, the overall item error rate were 0.1% for coding, and 0.2% for entry.

Another quality control study that measured error rates was presented by Norton, Buchanan, Rossman, Chakraborty, and Weiss (1981). The data consisted of personal names, place names, and dates. Error rates were computed by week for each data field. During a twenty-week period, the average error rate was 4.0 ± 1.5%, with a range of 1.7 – 7.2% for all fields.

Schwartz (1982) discussed error rates for data transcribed from Internal Revenue Service tax forms for Statistics of Income studies. He stressed the importance of a clear definition of "defect," or "error". Historically, tax form error rates (at least one mistake per form) were about 20% for common forms and as high as 45% for complex forms. Item error rates were generally less than five percent even for complex forms. The 1040 tax form had a 25% form error rate, but only 0.5% of the entries were mistakes. An analysis of errors attributed to transcribers showed that most were omissions (42%) or incorrect entries (37%); transpositions accounted for 8%.

1.6 Inappropriate RDM Techniques

As mentioned earlier, some RDM methods commonly used in complex (large, long-term) research projects are not appropriate to the target studies that are the focus of our research. Developing complex editing or quality control systems is usually beyond the resources of relatively simple (small, short-term) studies. The following papers are illustrations of methods that we are aware of, but do not address in our models. These methods include automatic correction, process control, and the use of complex models for designing edit criteria.
Fellegi (1975), Fellegi and Holt (1976), and Liepins (1980) presented models designed to define and evaluate edit systems that use automatic correction to reduce errors. Biemer (1980) included errors from editing and imputation procedures in a model of response and sampling errors designed to assess the impact of different error types in a sample survey study.

Large projects both require and allow certain RDM techniques that are not appropriate for smaller studies. The process control methods discussed by Minton (1969) are not cost-effective for smaller studies because of the relatively high development costs for designing the sampling review system. Muller, Smith, and Bass (1982) outlined special problems that must be considered for large datasets (more than 100,000 records). Greenberg (1982) proposed using an existing edit system to plan editing specifications for future data processing. Kalton (1982) provided a complete summary of common imputation methods and compared six methods under simple assumptions. The high cost of developing an automatic correction system and potential bias problems make the method inappropriate for the target situation.
CHAPTER 2
PRESENTATION OF MODEL 1

This chapter presents a model for a target study that is done in a RDM environment with limited error detection options. The purpose is to describe the model framework being proposed. This model, called Model 1, is the basis for models of more complex situations. RDM terms used in describing the model are defined for clarity. The probability structure of Model 1 and simplifying assumptions are presented, followed by derivations for estimating error rates and system costs. Numerical examples demonstrate potential applications of the model.

2.1 Definition of Terms

The terms we use in presenting the model framework are based on common RDM terminology, but their exact meaning may be more restrictive than usual. In the following descriptions, the term being defined is highlighted.

2.1.1 RDM System Elements

For modeling purposes, an overall system consists of all potential data processing steps for creating a file ready for analysis using computer programs. An overall system is composed of phases, which involve one or more manual processes or computer programs. A system component is a specific data process that may occur more than once. Subsystems include only a subset of phases in the overall system and correspond to RDM systems that are actually implemented. Model 1, and all later models, are designed to allow at least two useful subsystems to be defined and compared.
The system components of Model 1 are specific versions of common RDM practices. These components are:

Data entry: the manual process of keying information into machine readable form. Performed by professional data entry services or by non-professionals. Professional services usually produce data with lower error rates than non-professionals, but errors still occur.

Dependent verification: the manual process of detecting errors by reprocessing with both original information and initial processing results available to the verifier.

Independent verification: the manual process of detecting errors by reprocessing with initial processing results unavailable to the verifier. This is a standard method for reducing data entry errors.

Redundant data entry: records are entered a second time by the same data entry process used for initial data entry. When redundant data entry is used, the two entry processes are called Primary Entry and Secondary Entry. Two files are created composed of records called primary records and secondary records, respectively. This procedure can be the basis for independent verification.

Proofreading: the manual process of comparing data entry results with original information to detect errors. No data entry is done during proofreading. If an error is detected, then some indication is made that an update is required. The correct value is noted for later updating.

Comparison: the process of comparing corresponding records with a computer program when redundant data entry is done. Corresponding primary and secondary records are called paired records. The assumption is that the comparison program is not complex. Very simple error messages are produced; compared records are printed for manual review if they are not identical. The result of a comparison is called a match if no differences are found. Proofreading is required to detect specific errors in primary records that are non-matches.
Update: the process of updating records in which errors were detected. Using proofreading results, error records are changed directly or indirectly. Changing records directly involves interactive updating while indirect changes are made by creating a separate update file.

Combination (Merge): the computer process of creating a file of all processed records. The file produced is considered the final dataset and combines (merges) the results of all error detection and updating processes.

2.1.2 Error Generation

The definition of an error is very simple for this model. Let records be divided into two types: 1) correct records, and 2) error records. Errors can be introduced in the data entry or update phases. Errors are detected during proofreading, but the "detection of an error" does not mean one actually exists. There can be undetected errors in the final dataset because errors are missed during proofreading or introduced by improper updating. An improper update means that the results of updating a record is an error.

2.1.3 Cost

For the overall system, or a subsystem, total cost is composed of two types of cost: development and processing. Development costs include the cost of designing and implementing computer programs, training cost for manual processes not performed by professional services, and overhead cost where appropriate. Processing costs include both computer and manual processing costs.

2.1.4 Model 1 Overall System

Model 1 has six system components and seven processing steps, as shown in Figure 2.1. The diagram is simplified; not all interim data files are shown. We will
FIGURE 2.1
Model 1 Overall System Components

INPUT

(1) Primary Data Entry
(2) Primary Verification

(1) Secondary Data Entry
(2) Secondary Verification

(3) Comparison
match

(4) Secondary Proofread
error detected

(4) Primary Proofread
no error detected
(use primary record)

(5) Update

Data Group 1

Data Group 3

Data Group 2

(6) Combination

FINAL
describe the processing flow using all phases in the overall system, although in practice only subsystems are implemented.

Processing begins with input data (assumed to be correct original information) that goes through data entry and dependent verification (Components 1 and 2). Both the Primary Entry and Secondary Entry phases include dependent verification. To simplify the model each entry phase is considered as one processing step; one error rate indicates the result of processing by both components. This simplification is appropriate when professional data entry services are used.

After data entry and dependent verification are performed, a comparison program (Component 3) divides the data into two groups before proofreading (Component 4). When paired records match, Primary Proofread is done using only the primary record. Non-matching primary records are proofread by Secondary Proofread based on error messages generated by the comparison program. Secondary records are not processed further once Comparison is performed. Although both data groups are proofread manually, the knowledge that the non-matched primary records have failed one test may make Secondary Proofread more effective than Primary Proofread. A subsystem would usually only include one proofreading phase. When an error is detected, the record is updated (Component 5).

At this point the original data are split into three groups: 1) primary records considered correct by Primary Proofread, 2) primary records considered correct even though the paired secondary record did not match, and 3) updated primary records. The last phase combines the three groups (Component 6) to produce a file ready for computer analysis.

The overall system of Model 1 produces a final dataset, which may contain undetected errors, using redundant data entry and the most basic edit method—manual proofreading. A subsystem that reflects a standard RDM system includes Primary Data Entry, Dependent Verification, Primary Proofread, Update, and Combination. A
subsystem in which redundancy is used includes Primary and Secondary Entry (no
dependent verification), Comparison, Secondary Proofread, Update, and Combination.
Figure 2.2 shows these two subsystems.

2.2 Probability Structure

The goal of designing a RDM probability model is to provide a method of esti-
mating error rates and system costs. The model structure must be flexible enough to
address any reasonable subsystem defined by combining appropriate phases of the
overall system. We want the model to provide a quantitative basis for choosing between
two subsystems. The strategy used to build the probability model was to consider
individual steps while keeping the goal in mind. Definitions of random variables and
any required parameters are designed to make derived error rate estimates depend on
results from previous stages. Each phase is considered in relation to previous phases.
The total cost estimate is also composed of separate estimates for each component.

The basic values to be estimated by the probability model are counts of the number
of records at various stages. Error generation processes can be explored when the
number of records of each type, correct and error, is estimable. The error rate of greatest
interest is the final error rate, which is the number of records with errors remaining in the
final dataset. We also need record counts to estimate processing costs. Therefore, a
record is the basic measurement unit of the probability model. All counts and rates are
based on records, as opposed to data fields or keystrokes. The probability structure is
designed to treat a record as a single entity; no relationships among information on the
same record (if there is more than one field) are considered.

Model 1 addresses random errors introduced in the data entry (and dependent
verification) or update phases. Systematic errors are not considered. Only two states are
possible for a record; a record is either a correct record or an error record. Indepen-
dence is assumed whenever possible and is discussed with the associated derivation.
FIGURE 2.3
Model 1 Overall System with Random Variables

N
INPUT

(1) Primary Data Entry
(2) Primary Verification

no match
(W_i = 1)

(3) Comparison

match
(W_i = 0)

W

(4) Secondary Proofread

no error detected
(Y_i = 0)

error detected
(U_i = 1)

(4) Primary Proofread

no error detected
(U_i = 0)

U

Data Group 1

Data Group 3

U + Y

Data Group 2

W - Y

(6) Combination

Final error rate
\[ \frac{Z}{N} \]

FINAL

Type of Records Counted

N Primary
W Non-matched primary
U Error detected by Primary Proofread
Y Error detected by Secondary Proofread
Z Error in final dataset
Conditional parameters are defined when assuming independence does not provide useful results.

The cost estimation aspects of Model 1 are based on assuming that some information about potential costs exists. In order to have valid cost estimates, reasonable cost values for the various components must be available. The minimum requirement is that a range of values can be indentified for all cost parameters.

2.3 Error Generation Derivations

The foundation of the probability structure of Model 1 consists of random variables defined to estimate the number of records after various processing steps. The natural approach is to define these count random variables as the sum of appropriately defined Bernoulli random variables.

A description of the major random variables is useful at this point. Figure 2.3 shows the overall system with the count random variables. The $N$ records in a input dataset are processed to produce a final dataset containing $N$ records. During processing the $N$ records are split into various groups depending on the outcome of edit checking. The number of records that do not match after Comparison is indicated by $W$ ($W_i = 1$ when paired records do not match). Thus, the number of records processed by Primary Proofread is $N - W$. After Primary Proofread the number of records that are considered correct is $(N - W) - U$; $U$ is defined as the number of records where errors are detected. Similarly for Secondary Proofread, $Y$ is defined as the number of error records and $W - Y$ is the number of records considered correct. The number of records that must be updated is $U + Y$, the total number of records where errors were detected by either proofread step. The number of records in the final dataset is the sum of the number of primary records from the three possible paths:

$$[(N - W) - U] + (W - Y) + (U + Y) = N.$$
The final error rate is $Z / N$, in which $Z$ is defined as the number of records with undetected errors in the final dataset.

Each count random variable has a binomial distribution and an associated probability parameter. The definition and value of these probability parameters determines the situation being modeled. Error creation or detection rates, called count parameters, are defined as necessary. A summary of the count parameters in Model 1 is provided in Figure 2.4.

2.3.1 Notation

The following notation is used.

(i) When $\Sigma$ is used without qualification, it indicates the sum for $i = 1, \ldots, N$.

(ii) The standard notation $B(n, p)$ represents a random variable that has a binomial distribution with parameters $(n, p)$. Thus, a Bernoulli random variable, $X$, is $B(1, p_X)$ where $p_X = P(X = 1)$.

(iii) Let a Bernoulli function, $\beta$, be defined as follows:

$$\beta(\text{argument}) = \begin{cases} 1 \text{ if argument is true,} \\ 0 \text{ otherwise (i.e., argument is false).} \end{cases}$$

Using this notation, $X = \beta(\text{error exists})$ implies that $X \sim B(1, p_X)$ and $p_X = P(\text{error exists})$. 
2.3.2 Data Entry

The Primary and Secondary Entry phases are represented by two random variables, $X_1$ and $X_2$, respectively. Parameters that indicate data entry error rates are necessary in order to get count estimates. One error rate parameter applies to the results of both data entry and, if performed, dependent verification.

Definition 2.1a. Let $X_{1i} \sim \beta(\text{error exists in record } i \text{ after Primary Entry})$. Assuming $X_{1i}$ are i.i.d. $B(1, e_1)$, it follows that $X_1 = \sum X_{1i} \sim B(N, e_1)$. $X_1$ is the number of records with errors present after Primary Entry. The estimated count is $E(X_1) = Ne_1$ with $V(X_1) = Ne_1(1 - e_1)$,

$$e_1 = P(X_{1i} = 1) = P(\text{error after Primary data entry and dependent verification}).$$

Definition 2.1b. Let $X_{2i} \sim \beta(\text{error exists in record } i \text{ after Secondary Entry})$. Assuming $X_{2i}$ are i.i.d. $B(1, e_2)$, then $X_2 = \sum X_{2i} \sim B(N, e_2)$ is the number of errors present after Secondary Entry, and

$$e_2 = P(X_{2i} = 1) = P(\text{error after Secondary entry and dependent verification}).$$

2.3.3 Comparison

The Comparison phase involves both primary and secondary records. The count of interest, defined as $W$, is the number of primary records that require secondary proofreading because they do not match their corresponding secondary records. If a subsystem does not include Secondary Entry then Comparison is omitted, $W = 0$ by definition, and all records pass to Primary Proofread.

For this phase, a conditional parameter is required to make the model more useful. Two possibilities exist when paired records match: either both are correct or both have
the same wrong value(s). A parameter, $m_1$, is defined to consider both possibilities; $m_1$ is the probability that, given an error exists in each record, paired records are identical.

**Theorem 2.1.** Let $W_i = \beta$ (no match for record $i$ at Comparison) and assume $W_i$ are i.i.d., $W_i \sim B(1, w)$. Then $W = \sum W_i \sim B(N, w_1)$ is a count of the number of primary records that do not match their corresponding secondary records; these records are processed by the Secondary Proofread phase. Assuming both Primary and Secondary Entry are performed, and $X_{1i}$ and $X_{2i}$ are independent, then

$$w_1 = P(W_i = 1) = e_1 + e_2 - (1 + m_1)e_1e_2$$

(2.1)

and $m_1 = P(\text{match \& errors exist in both Primary and Secondary})$

$$= P(\text{same bad value} \mid X_{1i} = 1 \& X_{2i} = 1).$$

**Proof:**

$$w_1 = P(W_i = 1)$$

$$= P(\text{different values})$$

$$= 1 - P(\text{same values from Primary and Secondary Entry})$$

$$= 1 - P(\text{both correct or both have the same error})$$

$$= 1 - [P(X_{1i} = 0 \& X_{2i} = 0) + P( X_{1i} = 1 \& X_{2i} = 1 \& \text{same bad value})]$$

$$= 1 - [(1 - e_1)(1 - e_2)$$

$$+ P(\text{same bad value} \mid X_{1i} = 1 \& X_{2i} = 1)P( X_{1i} = 1 \& X_{2i} = 1)]$$

$$= 1 - [(1 - e_1)(1 - e_2) + m_1P( X_{1i} = 1 \& X_{2i} = 1)]$$

$$= 1 - [(1 - e_1)(1 - e_2) + m_1e_1e_2]$$

$$= e_1 + e_2 - (1 + m_1)e_1e_2$$
2.3.4 Primary Proofread

The count of interest for Primary Proofread, $U$, is the number of records in which errors are detected. These are the records that require updating. If Primary Proofread is not performed, then $U = 0$ by definition.

The possible outcomes are shown in the diagram below. Although actual implementation is unlikely, the overall system includes the possibility that Primary Proofread is performed when redundancy is used. If a primary record is not a match ($W_i = 1$) at Comparison, then no error could be detected by Primary Proofread. When the record is proofread, there are several possibilities depending on whether or not an error actually exists.

When considering the possible pathways, it is clear that dependencies are involved. Conditional probabilities allow the model to account for differences in error detection effectiveness under different conditions.
Lemma 2.1. If Primary Entry, Secondary Entry, and Comparison are performed, then

\[
P(X_{1i} = 0 \& W_i = 0) = (1 - e_1)(1 - e_2)
\]

(2.2)

and

\[
P(X_{1i} = 1 \& W_i = 0) = m_i e_1 e_2.
\]

(2.3)

Proof of 2.2:

\[
P(X_{1i} = 0 \& W_i = 0)
\]

\[= P(\text{no error in primary record} \& \text{match})
\]

\[= P(X_{1i} = 0 \& [(\text{both correct}) \text{ or } (\text{same bad value})])
\]

\[= P([X_{1i} = 0 \& (X_{1i} = 0 \& X_{2i} = 0)]
\]

\[\text{or } [X_{1i} = 0 \& (\text{same value} \& X_{1i} = 1 \& X_{2i} = 1)])
\]

\[= P(X_{1i} = 0 \& X_{2i} = 0)) + 0
\]

\[= (1 - e_1)(1 - e_2).
\]

Proof of 2.3:

\[
P(X_{1i} = 1 \& W_i = 0)
\]

\[= P(\text{error in primary record} \& \text{match})
\]

\[= P(X_{1i} = 1 \& [(\text{both correct}) \text{ or } (\text{same bad value})])
\]

\[= P([X_{1i} = 1 \& (X_{1i} = 0 \& X_{2i} = 0)]
\]

\[\text{or } [X_{1i} = 1 \& (\text{same value} \& X_{1i} = 1 \& X_{2i} = 1)])
\]

\[= P(X_{1i} = 1 \& (X_{1i} = 0 \& X_{2i} = 0))
\]

\[+ P(X_{1i} = 1 \& (\text{same value} \& X_{1i} = 1 \& X_{2i} = 1))
\]

\[= 0 + P(\text{same bad value} \mid X_{1i} = 1 \& X_{2i} = 1)P(X_{1i} = 1 \& X_{2i} = 1)
\]

\[= m_i e_1 e_2.
\]
**Theorem 2.2.** Let $U_i = \beta$(error detected for record i at Primary Proofread) and assume $U_i$ are i.i.d., $U_i \sim B(1, u_i)$. Then $U = \sum U_i \sim B(N, u_i)$ is a count of the number of records that must be updated because an error was detected by Primary Proofread.

Let the conditional parameters $r_{111}, r_{101}, r_{110}, r_{100}$, be defined as follows, in which \( pf \) represents proofread:

\[
\begin{align*}
    r_{111} &= P(\text{detected} \mid \text{error exists & pf.}) = P(U_i = 1 \mid X_{1i} = 1 \& W_i = 0), \\
    r_{101} &= P(\text{not detected} \mid \text{error exists & pf.}) = P(U_i = 0 \mid X_{1i} = 1 \& W_i = 0), \\
    r_{110} &= P(\text{detected} \mid \text{no error exists & pf.}) = P(U_i = 1 \mid X_{1i} = 0 \& W_i = 0), \\
    r_{100} &= P(\text{not detected} \mid \text{no error exists & pf.}) = P(U_i = 0 \mid X_{1i} = 0 \& W_i = 0).
\end{align*}
\]

If only Primary Entry is used, then $u_i = u_{11}$, where

\[
u_{11} = r_{110}(1-e_1) + r_{111}e_1. \tag{2.4}
\]

If Primary Proofread is performed when both Primary and Secondary Entry are used, then the primary record is proofread only if the corresponding secondary record matches. Assuming $X_{1i}$ and $X_{2i}$ are independent, then $u = u_{21}$.

\[
u_{21} = r_{110}(1-e_1)(1-e_2) + r_{111}e_1e_2. \tag{2.5}
\]

**Proof of 2.4:**

\[
u_{11} = P(U_i = 1)
\]

\[
= P(\text{detected & proofread})
\]

\[
= P(\text{detected & proofread & (no error exists or error exists)})
\]

\[
= P(U_i = 1 \& \text{proofread} \& X_{1i} = 0) + P(U_i = 1 \& \text{proofread} \& X_{1i} = 1)
\]

\[
= P(U_i = 1 \mid X_{1i} = 0 \& \text{proofread})P(X_{1i} = 0 \& \text{proofread})
\]

\[
+ P(U_i = 1 \mid X_{1i} = 1 \& \text{proofread})P(X_{1i} = 1 \& \text{proofread})
\]

\[
= r_{110}P(X_{1i} = 0) + r_{111}P(X_{1i} = 1) \quad \text{No comparison, all proofread.}
\]

\[
= r_{10}(1-e_1) + r_{11}e_1.
\]
Proof of 2.5:

\[ u_{21} = P(U_i = 1) \]
\[ = P(\text{detected \& proofread}) \]
\[ = P(\text{detected \& proofread \& (no error exists or error exists)}) \]
\[ = P([\text{detected \& proofread \& no error}] \text{ or } [\text{detected \& proofread \& error}]) \]
\[ = P(U_i = 1 \land W_i = 0 \land X_{li} = 0) + P(U_i = 1 \land W_i = 0 \land X_{li} = 1) \]
\[ = P(U_i = 1 \land X_{li} = 0 \land W_i = 0)P(X_{li} = 0 \land W_i = 0) \]
\[ + P(U_i = 1 \land X_{li} = 1 \land W_i = 0)P(X_{li} = 1 \land W_i = 0) \]
\[ = r_{110}P(X_{li} = 0 \land W_i = 0) + r_{111}P(X_{li} = 1 \land W_i = 0) \]
\[ = r_{110}(1 - e_1)(1 - e_2) + r_{111}m_1 e_1 e_2. \]

By Lemma 2.1.

2.3.5 Secondary Proofread

The situation for Secondary Proofread is similar to that of Primary Proofread. The objective is to count the number of records, \( Y \), which have detected errors and require updates. If Secondary Proofread is not performed, then \( Y = 0 \) by definition. Any reasonable subsystem that includes Secondary Entry also includes Comparison and Secondary Proofread.

The diagram of possible outcomes for Secondary Proofread shows the similarity to Primary Proofread. Records are processed by Secondary Proofread only when paired records do not match. The purpose of Secondary Proofread is to determine if a primary record is correct or not, using original input information and the comparison program messages. The secondary record is only a means to detect errors in the primary record.
The consideration of conditional probabilities can be simplified for Secondary Proofread. Since the comparison program provides additional information and fewer records are proofread, we assume that the probability of detecting an error when none exists is zero \( P(Y_i = 1 \mid X_{1i} = 0 \& W_i = 1) = 0 \). With this simplification, only one conditional parameter is required, which allows for the possibility that an existing error is not detected.

*Theorem 2.3.* Let \( Y_i = \beta \) (error detected for record \( i \) at Secondary Proofread) and assume \( Y_i \) are i.i.d., \( Y_i \sim B(1, y_i) \). Then \( Y = \sum Y_i \sim B(N, y_i) \) is a count of the number of records that must be updated because an error is detected by Secondary Proofread.

Let the conditional parameter, \( c_1 \), be defined as follows:

\[
c_1 = P(\text{detected} \mid \text{error exists & proofread}) = P(Y_i = 1 \mid X_{1i} = 1 \& W_i = 1).
\]

When Primary Entry, Secondary Entry, and Comparison are done, assuming \( X_{1i} \) and \( X_{2i} \) are independent, then

\[
y_i = c_1 [e_1 (1 - e_2) + (1 - m_1)e_1 e_2]. \tag{2.6}
\]

By definition, \( 1 - m_1 = P(\text{no match} \mid X_{1i} = 1 \& X_{2i} = 1) \).
Proof:

\[ y_1 = P(Y_i = 1) = P(\text{detected} \& \text{proofread}) \]
\[ = P(\text{detected} \& \text{proofread} \& (\text{no error exists or error exists})) \]
\[ = P(Y_i = 1 \& \text{proofread} \& X_{ii} = 0) + P(Y_i = 1 \& \text{proofread} \& X_{ii} = 1) \]
\[ = P(Y_i = 1 \mid X_{ii} = 0 \& \text{proofread})P(X_{ii} = 0 \& \text{proofread}) \]
\[ + P(Y_i = 1 \mid X_{ii} = 1 \& \text{proofread})P(X_{ii} = 1 \& \text{proofread}) \]
\[ = 0 \times P(X_{ii} = 0 \& W_i = 1) \]
\[ + P(Y_i = 1 \mid X_{ii} = 1 \& W_i = 1)P(X_{ii} = 1 \& W_i = 1) \]
\[ = 0 + c_1 P(W_i = 1 \& X_{ii} = 1) \]
\[ = c_1 \left[ P(W_i = 1 \& X_{ii} = 1 \& (X_{2i} = 0 \text{ or } X_{2i} = 1)) \right] \]
\[ = c_1 \left[ P(W_i = 1 \& X_{ii} = 1 \& X_{2i} = 0) + P(W_i = 1 \& X_{ii} = 1 \& X_{2i} = 1) \right] \]
\[ = c_1 \left[ P(W_i = 1 \mid X_{ii} = 1 \& X_{2i} = 0)P(X_{ii} = 1 \& X_{2i} = 0) \right. \]
\[ + \left. P(W_i = 1 \mid X_{ii} = 1 \& X_{2i} = 1)P(X_{ii} = 1 \& X_{2i} = 1) \right] \]
\[ = c_1 \left[ 1 \times P(X_{ii} = 1 \& X_{2i} = 0) + (1 - m_1)P(X_{ii} = 1 \& X_{2i} = 1) \right] \]
\[ = c_1 \left[ e_1(1 - e_2) + (1 - m_1)e_1e_2 \right] \]

2.3.6 Errors in Final Dataset

The error records that remain in the final dataset have errors that are undetected during proofreading or the result of improper updating. The number of final errors obviously depends on which components are included in a subsystem. There are three mutually exclusive events that lead to the existence of an error record in the final dataset:

(i) an error record passes through Primary Proofread undetected;
(ii) an error record passes through Secondary Proofread undetected;
(iii) an update results in an error record.
Theorem 2.4. Let $Z_i = \beta(\text{error for record } i \text{ in final dataset})$ and assume $Z_i$ are i.i.d., $Z_i \sim B(1, z)$. Then $Z = \sum Z_i \sim B(N, z_1)$ is a count of the number of errors remaining in the final dataset, which are the result of inaccurate error detection or improper updating.

Let the parameter $k$ indicate the probability that an update results in a proper correction, i.e., an updated record is correct:

$k = P(\text{proper correction} \mid \text{update needed}).$

For all subsystems the following holds:

$z_1 = P(Z_i = 1)$

$= P(\text{undetected error in Primary Proofread})$

$+ P(\text{undetected error in Secondary Proofread})$

$+ P(\text{improper update}).$

For a subsystem where Secondary Proofread is not performed, usually because Secondary Entry and Comparison have not been done, $z_1 = z_{11}$, where

$z_{11} = r_{101}e_1 + (1 - k)[r_{110}(1 - e_1) + r_{111}e_1].$ \hspace{1cm} (2.7)

When Secondary Entry, Comparison and Secondary Proofread are performed, and Primary Proofread is not used, then $z = z_{12}$, where

$z_{12} = e_1 - c_1k(e_1 - m_1e_1e_2).$ \hspace{1cm} (2.8)
Proof of 2.7:

\[ z_{11} = P(Z_1 = 1) \]

\[ = P(\text{undetect. Prim. Pf.}) + P(\text{undetect. Sec. Pf.}) + P(\text{improper update}) \]

\[ = P(\text{undetect. Primary Pf.}) + 0 + P(\text{improper update} \mid U_i = 1)P(U_i = 1) \]

\[ = P(\text{not detected} \mid \text{error} \& \text{proofread})P(\text{error} \& \text{proofread}) \]

\[ + P(\text{improper update} \mid U_i = 1)P(U_i = 1) \]

\[ = r_{01}P(X_{1i} = 1 \& W_i = 0) + P(\text{improper update} \mid U_i = 1)P(U_i = 1) \]

\[ = r_{01}e_1 + P(\text{improper update} \mid U_i = 1)P(U_i = 1) \]

\[ = r_{01}e_1 + (1 - k)u_{11} \]

\[ = r_{01}e_1 + (1 - k)[r_{10}(1 - e_1) + r_{11}e_1]. \]

Proof of 2.8:

\[ z_{12} = P(Z_1 = 1) \]

\[ = P(\text{undetect. Prim. Pf.}) + P(\text{undetect. Sec. Pf.}) + P(\text{improper update}) \]

\[ = P(\text{match} \& X_{1i} = 1 \& X_{2i} = 1) + P(\text{undetect. Sec.}) + P(\text{improper update}) \]

\[ = P(\text{match} \mid X_{1i} = 1 \& X_{2i} = 1)P(X_{1i} = 1 \& X_{2i} = 1) \]

\[ + P(\text{undetect. Sec.}) + P(\text{improper update}) \]

\[ = m_1e_1e_2 + P(\text{undetected Sec.}) + P(\text{improper update}) \]

\[ = m_1e_1e_2 + P(\text{not detected} \mid \text{error} \& \text{proofread})P(\text{error} \& \text{proofread}) \]

\[ + P(\text{improper update} \mid Y_i = 1)P(Y_i = 1) \]

\[ = m_1e_1e_2 + (1 - c_i)P(W_i = 1 \& X_{1i} = 1) \]

\[ + P(\text{improper update} \mid Y_i = 1)P(Y_i = 1) \]

\[ = m_1e_1e_2 + (1 - c_i)P(W_i = 1 \& X_{1i} = 1) + (1 - k)(c_1P(W_i = 1 \& X_{1i} = 1)) \]

\[ = m_1e_1e_2 + P(W_i = 1 \& X_{1i} = 1)(1 - kc_1) \]

\[ = m_1e_1e_2 + (e_1(1 - e_2) + (1 - m_1)e_1e_2)(1 - kc_1) \]

See Proof of 2.6.

\[ = e_1 - c_1k_1(e_1 - m_1e_1e_2). \]
FIGURE 2.4
Model 1 Probability Parameters

Data Entry and Dependent Verification

\[ e_1 = \text{Prob(error after Primary Entry (and Verification if done))} = P(X_{1i} = 1) \]
\[ e_2 = \text{Prob(error after Secondary Entry (and Verification if done))} = P(X_{2i} = 1) \]

Comparison

\[ m_1 = \text{Prob(match | errors exist in corresponding Primary and Secondary records)} = P(\text{same bad value} | X_{1i} = 1 & X_{2i} = 1) \]

Primary Proofread

\[ r_{111} = \text{Prob(detected | error exists & proofread)} = P(U_i = 1 | X_{1i} = 1 & W_i = 0) \]
\[ r_{101} = \text{Prob(not detected | error exists & proofread)} = P(U_i = 0 | X_{1i} = 1 & W_i = 0) \]
\[ r_{111} + r_{101} = 1 \]
\[ r_{110} = \text{Prob(detected | no error exists & proofread)} = P(U_i = 1 | X_{1i} = 0 & W_i = 0) \]
\[ r_{100} = \text{Prob(not detected | no error exists & proofread)} = P(U_i = 0 | X_{1i} = 0 & W_i = 0) \]

Secondary Proofread

\[ c_1 = \text{Prob(detected | error exists & proofread)} = P(Y_i = 1 | X_{1i} = 1 & W_i = 1) \]

Correction

\[ k = \text{Prob(proper correction | update needed)} \quad \text{Considered} \approx 1 \text{ and independent of edit method.} \]

Bernoulli Random Variables

\[ X_{1i} = \text{B(error exists in record i after Primary Data Entry)} \sim B(1, e_1) \]
\[ X_{2i} = \text{B(error exists in record i after Secondary Data Entry)} \sim B(1, e_2) \]
\[ W_i = \text{B(no match for record i at Comparison)} \]
\[ U_i = \text{B(error detected by Primary Proofread for record i)} \]
\[ Y_i = \text{B(error detected by Secondary Proofread for record i)} \]
\[ Z_i = \text{B(error exists in Final Dataset for record i)} \]
2.4 Cost Estimation

Estimation of total cost in Model 1 is designed to provide a method for comparing subsystems. Total cost of the overall system must have separate elements for each component to make finding the cost for any reasonable subsystem straightforward. Development and processing costs are considered separately. Development costs are fixed costs that do not vary with the number of records processed. The processing cost for a particular phase depends on the average processing cost per record and the number of records processed.

The six components in the model are considered separately for cost estimation. As shown in Figure 2.3, the components are: 1) data entry, 2) dependent verification, 3) comparison of corresponding records, 4) manual proofread, 5) update, and 6) production of the final dataset. We can simplify cost estimation by assuming that costs for the data entry, dependent verification, and manual proofreading phases are basically the same for both Primary and Secondary processing. Only processing costs double when a component, such as data entry, is used twice in a subsystem because development is a fixed cost.

Any cost estimate requires some knowledge of potential development and processing costs for each system component. Assuming that these cost values are available a target study is reasonable. Previous experience or pilot studies provide sufficient information about cost values in most cases. Relationships between costs for certain components may also provide a basis for cost estimates.

There are three types of cost parameters, which address: 1) development costs, 2) processing costs, and 3) the inclusion of optional components. We define them as follows.
<table>
<thead>
<tr>
<th>System Cost Component</th>
<th>Development Cost</th>
<th>Processing Cost per Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Entry (Primary, Secondary)</td>
<td>$D_{11}$</td>
<td>$P_{11}$</td>
</tr>
<tr>
<td>(2) Dependent Verification (Primary, Secondary)</td>
<td>$D_{12}$</td>
<td>$P_{12}$</td>
</tr>
<tr>
<td>(3) Comparison</td>
<td>$D_{13}$</td>
<td>$P_{13}$</td>
</tr>
<tr>
<td>(4) Proofread (Primary, Secondary)</td>
<td>$D_{14}$</td>
<td>$P_{14}$</td>
</tr>
<tr>
<td>(5) Update</td>
<td>$D_{15}$</td>
<td>$P_{15}$</td>
</tr>
<tr>
<td>(6) Combine to create analysis file</td>
<td>$D_{16}$</td>
<td>$P_{16}$</td>
</tr>
</tbody>
</table>

Let $s = \beta$(Secondary Entry, Comparison, and Secondary Proofread done).

Let $t = \beta$(Primary Proofread done).

The indicator variables, $s$ and $t$, provide an easy method for finding total cost for subsystems. When redundancy is used in subsystem, which means that Secondary Entry, Comparison and Secondary Proofread are done, then $s = 1$. When Primary Proofread is included, then $t = 1$.

Values for the cost parameters depend on the particular RDM situation involved. Development cost values are found by considering overhead, planning, training, or computer programming (including testing) required for a component. Approximate processing cost values (per record) are generally based on the cost of processing a batch of records. The cost of combining data streams is a fixed cost since a merge program must process all the records regardless of how the data are divided by the edit checking steps.
Definition 2.2. The total cost, \( T_1 \), of the overall system is defined as
\[
T_1 = D_1 + P_1. \tag{2.9}
\]
Total development cost is \( D_1 \) and total processing cost is \( P_1 \).
\[
D_1 = D_{11} + D_{12} + sD_{13} + D_{14} + D_{15} + D_{16}, \tag{2.10}
\]
\[
P_1 = NP_{11P} + bNP_{11S} + sNP_{13} + t(N - W)P_{14} + sWP_{14} + (U + Y)P_{15} + P_{16}. \tag{2.11}
\]
\[
P_{11P} = \begin{cases} 
P_{11} + P_{12} & \text{if dependent verification is performed,} \\
P_{11} & \text{if not.} \end{cases}
\]
Similarly for \( P_{12S} \).

Equations 2.10 and 2.11 follow from the definitions of the cost parameters and the count random variables. For Definition 2.2, we assume that some form of edit checking is included in any subsystem of interest. Thus, the costs for comparing, proofreading, updating and merging records (\( D_{13}, D_{14}, D_{15}, D_{16}, P_{13}, P_{14}, P_{15}, P_{16} \)) are non-zero. Entry development cost may be considered zero if professional entry is used. If neither Primary Entry, nor Secondary Entry, include dependent verification, then \( D_{12} = 0 \) and \( P_{12} = 0 \).

Total development cost, \( D_1 \), is simply the sum for all system components. The inclusion of development cost for Comparison depends on the value of \( s \), which essentially indicates whether or not redundancy is used.

The total processing cost, \( P_1 \), involves indicators for optional components, the number of records in each phase, and the processing cost parameters. For each component with variable costs, the processing cost is found by multiplying the number of records processed by the average processing cost for one record. The count random variables appear for the appropriate components: \( W \) is the number of non-matches after
Comparison, \( U \) is the number of records with errors detected by Primary Proofread, and \( Y \) is the number of records with errors detected by Secondary Proofread. By definition, \( W = 0 \) and \( Y = 0 \) if Comparison is not included in a subsystem. If Primary Proofread is not included, then \( U = 0 \). The probability parameters for the count random variables are shown in Figure 2.5.

For the Standard subsystem (see Figure 2.2) the total cost is

\[
T_{1S} = [D_{11} + D_{12} + D_{14} + D_{15} + D_{16}] + [N(P_{11} + P_{12}) + NP_{14} + P_{15}U + P_{16}].
\]

This follows from the definition of total cost since the Standard subsystem includes dependent verification \((P_{11S} = P_{11} + P_{12})\) and Primary Proofread \((t = 1)\), and does not include Comparison and Secondary Proofread \((s = 0, W = 0, Y = 0)\). The expected value for \( T_{1S} \) follows from Theorem 2.2, \( E(U) = Nu_{11} \).

The total cost estimate for the Redundancy subsystem is

\[
T_{1R} = [D_{11} + D_{13} + D_{14} + D_{15} + D_{16}] + [2NP_{11} + NP_{13} + P_{14}W + P_{15}Y + P_{16}].
\]

Redundancy does not have dependent verification \((D_{12} = 0, P_{12} = 0)\), does not include Primary Proofread \((t = 0, U = 0)\), and includes Comparison and Secondary Proofread \((s = 1)\). The expected value for \( T_{1R} \) follows from Theorems 2.1 and 2.3, which imply \( E(W) = Nw_{1} \) and \( E(Y) = Ny_{1} \) when redundancy is used.

The predicted difference in total cost between the Standard and Redundancy subsystems is not a function of all the cost parameters. The difference of the expected total costs,

\[
E(T_{1R}) - E(T_{1S}) = [D_{13} - D_{12}] + N[P_{11} - P_{12} + P_{13} - (1 - w_{1})P_{14} - (u_{11} - y_{1})P_{15}],
\]

has fewer parameters than \( T_{1S} \) or \( T_{1R} \) because the subsystems have some components in common. The only important development costs in comparing the subsystems are for dependent verification and the comparison program.
FIGURE 2.5
Model 1 Count Estimation Theorems

2.1 \( W \sim B(N, w_1) \), number of non-matches at Comparison

\[
(2.1) \quad w_1 = e_1 + e_2 - (1 + m_1)e_1e_2
\]
Redundancy

2.2 \( U \sim B(N, u_1) \), number of errors detected by Primary Proofread

\[
(2.4) \quad u_{11} = r_{110}(1 - e_1) + r_{111}e_1
\]
Standard

\[
(2.5) \quad u_{12} = r_{110}(1 - e_1)(1 - e_2) + r_{111}m_1e_1e_2
\]
Redundancy

2.3 \( Y \sim B(N, y_1) \), number of errors detected by Secondary Proofread

\[
(2.6) \quad y_1 = c_1(e_1(1 - e_2) + (1 - m_1)e_1e_2)
\]
Redundancy

2.4 \( Z \sim B(N, z_1) \), number of errors in final dataset

\[
(2.7) \quad z_{11} = r_{101}e_1 + (1 - k)[r_{110}(1 - e_1) + r_{111}e_1]
\]
Standard

\[
(2.8) \quad z_{12} = e_1 - c_1k[e_1 - m_1e_1e_2]
\]
Redundancy

Count Parameter Values

All rates are per record measures.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>e</td>
<td>0.10, 0.05, 0.01</td>
<td>Data entry error, ( e_1 = e_2 = e )</td>
</tr>
<tr>
<td>( r_{111} )</td>
<td>0.50, 0.90, 0.99</td>
<td>Error detected by Primary Proofread given error exists and proofread</td>
</tr>
<tr>
<td>( r_{110} )</td>
<td>0.05, 0.01</td>
<td>Error detected by Primary Proofread given no error exists and proofread</td>
</tr>
<tr>
<td>( c_1 )</td>
<td>0.90, 0.99</td>
<td>Error detected by Secondary Proofread given error exists and proofread</td>
</tr>
<tr>
<td>( m_1 )</td>
<td>0.25, 0.50</td>
<td>Match at Comparison given errors in both paired records</td>
</tr>
<tr>
<td>k</td>
<td>0.99</td>
<td>Proper correction given update done</td>
</tr>
</tbody>
</table>
2.5 Numerical Examples

The objective of this section is to demonstrate that exploring Model 1 leads to insights about error generation and total cost differences. A variety of situations within the target RDM environment can be modeled by choosing appropriate parameter values. These examples are not meant to be exhaustive; the purpose is to present possible applications in RDM for an analytical model framework. Error generation is considered first, followed by a hypothetical example of comparing alternative subsystems. The required calculations are easily performed using a microcomputer spreadsheet program.

2.5.1 Error Generation

To explore error generation, suppose the RDM situation of interest is an academic environment in which data processing tasks are performed by non-professionals (students). We will compare a Redundancy subsystem with a modified Standard subsystem that does not include dependent verification. The quality of work is assumed to be much lower than that expected for professional data processing.

The model's utility is shown more clearly when conservative, or wide, ranges for the count parameter values are used. Let the data entry error rates for Primary and Secondary Entry be equal \( e_1 = e_2 = e \). Three possibilities are considered for data entry; poor, fair, or good work by non-professionals is modeled by using ten, five or one percent as the data entry error rate. The quality of Primary Proofreading is also considered at three levels. The conditional error detection rate, \( r_{11} \), has values fifty, ninety or ninety-nine percent. Fifty percent describes a situation where proofreaders only detect half of the errors that exist. Such poor proofreading is quite possible with simple numeric data. The probability of detecting a non-existent error, \( r_{10} \), is considered to be small; five and one percent are used. A higher detection rate is expected for Secondary Proofread because additional information is available and fewer records are processed. Ninety or ninety-nine percent is used for \( c_1 \), the conditional detection rate for Secondary
Proofread. Only one situation is considered for updating records. The conditional probability that an update produces a proper correction, $k$, is considered very high; ninety-nine percent is used.

A value for the conditional probability, $m_1$, that a match occurs when paired records are both errors, is more difficult to identify. Unlike other parameters, there is no basis in the literature for proposing a value for $m_1$. However, investigations that consider records with a single data field imply that the value of $m_1$ is small. For the numerical examples, two values are used: 0.25, 0.50. Using large values for $m_1$ produces conservative final error rate estimates for a Redundancy subsystem.

Looking at the final error rate estimates for these parameter values leads to interesting conclusions. Figure 2.6 shows the estimates for both subsystems using all appropriate combinations of the parameter values shown in Figure 2.5. As expected, using either subsystem reduces the final error rate; it is always lower than the entry error rate, which would be the final error rate without any editing. Even for the worst case, half of the errors should be detected. The important parameters for the Standard subsystem appear to be the entry error rate and the detection rate, while only the entry error rate is important for Redundancy.

Comparing the two subsystems is simpler when we focus only on the important parameters. Figure 2.7 shows a plot of the final error rates for the more conservative values of the less important parameters ($r_{110} = 0.05, m_1 = 0.50$). It would appear that using a Standard subsystem can be better than Redundancy, but only if very good Primary Proofreading is available. Even with poor data entry ($e = 0.10$), using Redundancy seems likely to reduce the final error rate to an acceptable value.

The model provides a framework in which to investigate the effect of using different RDM edit methods. Final error rate estimates provide a basis for deciding whether to use a different subsystem or to improve the quality of a particular phase. For instance, we can see that acceptable final error rates may still be attained with good proofreading
FIGURE 2.6
Model 1 Final Error Rates

The conditional probability that an update results in a proper correction is held constant, $k = 0.99$. The number of errors are based on $N = 5000$.

Standard Subsystem

<table>
<thead>
<tr>
<th>Data Entry $e_1 = e_2 = e$</th>
<th>Primary Proofread</th>
<th>Number of errors $E(Z) = Nz_{11}$ s.d.</th>
<th>Error Rate $Z/N$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$r_{111}$</td>
<td>$r_{110}$</td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>0.50</td>
<td>0.05</td>
<td>255</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.01</td>
<td>253</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.05</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.01</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.05</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.01</td>
<td>10</td>
</tr>
<tr>
<td>0.05</td>
<td>0.50</td>
<td>0.05</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.01</td>
<td>127</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.05</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.01</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.05</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.01</td>
<td>5</td>
</tr>
<tr>
<td>0.01</td>
<td>0.50</td>
<td>0.05</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.01</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.05</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.01</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.05</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.01</td>
<td>1</td>
</tr>
</tbody>
</table>

Redundancy Subsystem

<table>
<thead>
<tr>
<th>Data Entry $e_1 = e_2 = e$</th>
<th>Comparison $c_1$</th>
<th>Number of errors $E(Z) = Nz_{12}$ s.d.</th>
<th>Error rate $Z/N$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$c_1$</td>
<td>$m_1$</td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>0.90</td>
<td>0.50</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.25</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.50</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.25</td>
<td>22</td>
</tr>
<tr>
<td>0.05</td>
<td>0.90</td>
<td>0.50</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.25</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.50</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.25</td>
<td>8</td>
</tr>
<tr>
<td>0.01</td>
<td>0.90</td>
<td>0.50</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>0.90</td>
<td>0.25</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.50</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.99</td>
<td>0.25</td>
<td>1</td>
</tr>
</tbody>
</table>
FIGURE 2.7
Comparison of Model 1 Final Error Rates

- $r_{111} = 0.50$ Primary Proofread detection rate
- $c_1$ Secondary Proofread detection rate

Entry error rate $e = 0.10, 0.05, 0.01$  
($r_{110} = 0.05$)  
($m_1 = 0.50$)  
Standard Subsystem  
Redundancy Subsystem
even with poor data entry. Of course, lower error rates are usually associated with increased cost.

2.5.2 System evaluation

We will use hypothetical cost values to demonstrate the model's potential as a decision tool. More discussion and examples of system evaluation are presented in Chapters 4 and 5. The model includes cost estimation because data quality is not usually the only factor in making RDM decisions. Obviously if cost is not a factor, any edit checking method can potentially produce "error-free" data. It is important to remember that the validity of subsystem cost comparisons depends more on relative cost differences between phases than actual dollar values.

Let us consider three situations from the academic environment used for the count estimation examples. Suppose the data to be processed consist of 2000 relatively short, simple records. All processing can be done using existing microcomputers, but professional data entry is a possibility. In all cases, let the Primary Proofread detection rate of non-existent errors be low, $r_{110} = 0.01$ and the proper correction rate be high, $k = 0.99$. In the first situation, student labor and low wages are associated with high data entry error rates ($e = 0.10$) and poor proofreading ($r_{111} = 0.50$, $c_1 = 0.90$). The data entry wage is doubled in the second situation with some improvement in data entry error rates ($e = 0.05$), while proofreading is still poor. The third situation involves professional data entry and better quality proofreading ($e = 0.01$, $r_{111} = 0.90$, $c_1 = 0.99$). Suppose the wages for college students doing data entry and proofreading is $5 or $10 per hour. Let the wage for other components be $20 per hour. Computer costs associated with processing done using existing microcomputers are relatively low and are considered negligible for these examples.

A possible breakdown for development and processing costs is shown in Figure 2.8. The cost values are determined from labor cost values and projections of
the amount of time required for each component. The values are based on experience with small projects employing work-study students.

**FIGURE 2.8**

**Model 1 Cost Parameter Values**

<table>
<thead>
<tr>
<th>Component</th>
<th>Development Cost ($)</th>
<th></th>
<th></th>
<th>Development Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Labor</td>
<td>Computer</td>
<td>Total</td>
<td></td>
</tr>
<tr>
<td>(1) Entry</td>
<td>40 + 5 = 45</td>
<td>0</td>
<td>60</td>
<td>2 hours + 1 hour training</td>
</tr>
<tr>
<td>(2) Dep. Verification</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>(3) Comparison</td>
<td>40</td>
<td>0</td>
<td>40</td>
<td>2 hours</td>
</tr>
<tr>
<td>(4) Proofread</td>
<td>20 + 5 = 25</td>
<td>0</td>
<td>50</td>
<td>1 hour + 1 hour training</td>
</tr>
<tr>
<td>(5) Update</td>
<td>20</td>
<td>0</td>
<td>20</td>
<td>1 hour</td>
</tr>
<tr>
<td>(6) Merge</td>
<td>20</td>
<td>0</td>
<td>20</td>
<td>1 hour</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Processing Cost ($)</th>
<th>Processing Capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Entry</td>
<td>0.05 / record</td>
<td>100 per hour @$5 per hour</td>
</tr>
<tr>
<td></td>
<td>0.10 / record</td>
<td>100 per hour @$10 per hour</td>
</tr>
<tr>
<td>(2) Dep. Verification</td>
<td>Same cost as data entry when performed</td>
<td></td>
</tr>
<tr>
<td>(3) Comparison</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>(4) Proofread</td>
<td>0.04 / record</td>
<td>125 records per hour @$5 per hour</td>
</tr>
<tr>
<td></td>
<td>0.08 / record</td>
<td>125 records per hour @$10 per hour</td>
</tr>
<tr>
<td>(5) Update</td>
<td>0.40 / record</td>
<td>50 per hour @$20 per hour</td>
</tr>
<tr>
<td>(6) Merge</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

The results of comparing the two subsystems, Redundancy and Standard, are presented in Figure 2.9. Redundancy has fewer undetected errors for all three situations. The cost differences are not large for these cost parameter values, but they do demonstrate possible decision problems. In the first situation, Redundancy is much
better with regard to the final error rate, but the total cost is higher. The choice of the “best” subsystem would depend on the priorities of the project. In the second situation Redundancy is also more effective and more expensive. The Standard subsystem achieves a final error rate less than one percent in the third situation, which includes dependent verification, but Redundancy is now both more effective in reducing errors and less expensive.

**FIGURE 2.9**

**Model 1 Subsystem Comparisons**

<table>
<thead>
<tr>
<th></th>
<th>Situation 1</th>
<th>Situation 2</th>
<th>Situation 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data entry error rate</td>
<td>10%</td>
<td>5%</td>
<td>1%</td>
</tr>
<tr>
<td>Cost of entry per record</td>
<td>$0.05</td>
<td>$0.10</td>
<td>$0.05</td>
</tr>
<tr>
<td>Dep. verif. in Standard</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Proofreading</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Development cost</td>
<td>115</td>
<td>115</td>
<td>70</td>
</tr>
<tr>
<td>Processing cost</td>
<td>237</td>
<td>398</td>
<td>385</td>
</tr>
<tr>
<td>Total cost ($)</td>
<td>352</td>
<td>513</td>
<td>455</td>
</tr>
<tr>
<td>Undetected errors</td>
<td>101</td>
<td>51</td>
<td>10</td>
</tr>
<tr>
<td>Final error rate</td>
<td>5.1%</td>
<td>2.5%</td>
<td>0.5%</td>
</tr>
</tbody>
</table>

Model estimates provide additional information for RDM system evaluation, but the choice of the “best” subsystem may still depend on subjective priorities. One advantage of using a model is that the effects of a difference in quality, with or without an associated change in cost, can be considered objectively. The most expensive subsystem does not necessarily produce the best quality data.
2.6 Discussion

The objective of developing Model 1 was to create a framework for creating models of commonly used RDM systems. We find that the probability structure does provide insights into error generation as well as cost differences between two subsystems. Model estimates support the idea that redundancy is an edit method well worth considering. The next step is to present a complex, but tractable, model that addresses a greater variety of RDM possibilities. Once a sufficiently interesting model is developed, it can be explored and validated by comparing model estimates with actual implementation results.
CHAPTER 3
MODEL EXTENSIONS

The basic model, Model 1, is readily extended in a variety of ways. Five extended models are presented that apply to the target situation. Models 2 – 5 add edit checking phases at various stages: after data entry but before comparison of corresponding records, and after comparison. Model 6 considers redundant data collection.

3.1 Extended Model Background

The extended models add RDM components to Model 1, but are based on the same general assumptions. The glossary provides a summary of all terms used in discussing the models.

3.1.1 Definition of terms

The following definitions, in addition to those in Chapter 2, apply to the extended models.

Valid value: a value for a field that satisfies previously established guidelines. For a categorical field, the valid values are the valid responses for the field, together with special values that indicate unknown or missing information. For a continuous field, such as Age of Respondent, valid values are defined by a range of values and special “unknown” values.

Valid value edit: a simple edit check that compares the value of a field with the valid values for that field.
Switch error: an error that occurs when the value of a field is valid, but not correct. This type of error cannot be detected by a valid value edit.

Finding corrections: the process of using original information to investigate records with detected errors. The objective is to find the correct value(s) if an error actually exists, or to determine that no update is needed since the record is actually correct. Records are updated after corrections are found.

Valid value edit process, (VVE): the complete process that begins with a computer edit, which does valid value edits for all applicable fields, and ends with the update of incorrect records. Each field is checked separately; no multivariate checks are performed. Corrections are found based on error messages generated for records that have one or more detected errors.

Intelligent data entry: computer-assisted data entry with simple valid value editing. The valid values, or just the valid characters, for a field are specified before entry begins so that an invalid entry will not be accepted. A record with data entry error(s) detected this way is corrected quickly since the original input form is still present. Valid value editing may not apply to all fields of a record. Since only simple editing capability is assumed, no multivariate checks are included.

Primary Processing: this includes all steps used to produce primary records that are compared to corresponding secondary records. In Model 2, Primary Processing includes data entry, dependent verification and VVE.

Secondary Processing: this includes all steps used to produce secondary records. In Model 3, Secondary Processing includes data entry, dependent verification and VVE.

Initial records: edited primary records after Comparison and Manual Proofread corrections are made.

Initial Processing: this includes all steps used to produce initial records. In Model 4, VVE follows Initial Processing.
Data collection: the manual process of recording information that will be analyzed using computer programs. The information may be recorded by a respondent directly or by a data collector. Less complex situations are considered by assuming that all information for one record is collected in one session. Model 6 includes redundant data collection phases, called Primary Collection and Secondary Collection.

Compound error: a compound error is an error produced by a) a data collection error, and b) a data entry error in the same record. The distinction between an error and a compound error is necessary because of the very small possibility that a data collection error could be inadvertently corrected by an entry error. Model 6 includes the possibility of compound errors.

3.1.2 General Assumptions

The restrictions on the research data management situations being addressed apply to all the extended models. The objective is to describe more realistic situations by adding components, while keeping the models tractable. For each model, the overall system is designed so that interesting subsystems can be compared. As defined earlier, a subsystem of a particular model corresponds to an implemented RDM system, which is a subset of phases in the overall system.

Since the extended models build on the basic model, similar simplifications are used as necessary. The measurement unit is the record. A record is counted in two possible ways, as correct or as an error. For Models 2 – 5, which add VVE phases, the original input data are assumed correct before data entry. The sources for errors are the entry, edit, and update phases. The possibility of errors in data collection, in addition to other phases, is included in Model 6.

The probability structure of the extended models build on the model framework presented in Chapter 2. Only random errors are addressed; systematic errors are not considered. The same notation for count random variables is used in different models
when the same definitions apply, e.g., $W = \sum W_i$ is the number of non-matches in the Comparison phase of all models. For the same random variable, the difference between models is in the derivation of the binomial probability parameter.

3.2 Model 2 – Valid Value Edit in Primary Processing

Model 2 extends the basic model by adding a valid value edit process after Primary Entry. The overall system for Model 2 is shown in Figure 3.2.1. VVE is added to the Model 1 components (data entry, dependent verification, proofreading, comparison, update, combination). The phases in Model 2 are: Entry, VVE, Comparison, Manual Proofread, and Creation of final data. As before, Primary and Secondary Entry include the possibility of dependent verification. All of the edit checking phases include the process of finding corrections and updating primary records. Unlike Model 1, the update steps for Comparison and Manual Proofread are separate. Secondary Proofread in Model 1 becomes the find correction step of the Comparison phase.

There are five subsystems of interest in Model 2. In the simplest subsystem, the final dataset is created after only Primary Entry and VVE. Two subsystems combine VVE in Primary Processing with either Redundancy or Manual Proofread. Redundancy includes Secondary Entry and Comparison as in Model 1. The other two subsystems consist of only Redundancy or Manual Proofread, i.e., no VVE follows Entry. Since VVE is not included, these last two subsystems are the same as the Standard and Redundancy subsystems in Model 1. No subsystem that includes both Redundancy and Manual Proofread is considered. The three subsystems that include VVE are shown in Figure 3.2.2.

A comparison of the random variables in Model 2 with those in Model 1 is shown in Figure 3.2.3. Since the only additional phase is VVE, the count random variables for the other phases (entry—$X_1$ and $X_2$, comparison—$W$ and $Y$, proofread—$U$, undetected—$Z$) use the same notation.
FIGURE 3.2.1
Model 2 Overall System with Random Variables

\[ N \]
\[ \text{INPUT} \]

Primary Data Entry
Primary Verification

\[ X_1 \]

Valid value edit
error detected
no error detected

V_1

F_1

Update

N - V_1

primary

X_3

Secondary Data Entry
Secondary Verification

\[ X_2 \]

Find correction

V_1 - F_1

no correction

\[ \text{Comparison} \]

no match
match

W

N - W

Manual Proofread
detected
no error detected

U

(U - (N - W) - U)

Find correction
correction
no correction

Y

W - Y

Update

Create analysis file

\[ \text{FINAL} \]

\[ \frac{Z}{N} \]

X_1 \text{ Primary entry errors}
X_2 \text{ Secondary entry errors}
X_3 \text{ Primary Processing errors}
V_1 \text{ VVE detected errors}
F_1 \text{ VVE corrections}
W \text{ Comparison non-matches}
U \text{ Manual Proofread detected errors}
Y \text{ Comparison corrections}
Z \text{ Undetected errors}
FIGURE 3.2.2
Model 2 Subsystems
FIGURE 3.2.3
Comparison of Model 1 and Model 2

Model 1

\( x_1 \) Primary entry errors
\( x_2 \) Secondary entry errors
\( W \) Comparison non-matches
\( U \) Primary Proofread detected errors
\( Y \) Comparison corrections
\( Z \) Undetected errors

Model 2

\( x_1 \) Primary entry errors
\( x_2 \) Secondary entry errors
\( x_3 \) Primary Processing errors
\( V \) VVE detected errors
\( F \) VVE corrections
\( W \) Comparison non-matches
\( U \) Manual Proofread detected errors
\( Y \) Comparison corrections
\( Z \) Undetected errors
3.2.1 Error Generation Derivations

The error generation process in Model 2 is a direct expansion of Model 1. When appropriate, Model 2 theorems are based on corresponding Model 1 theorems. Correspondences occur because primary records in Model 2, produced by Primary Processing that includes VVE, are analogous to primary records produced by Primary Entry in Model 1.

3.2.1.1 Data Entry

As in Model 1, the results of the Primary and Secondary Entry phases are represented by two random variables, $X_1$ and $X_2$, respectively. Error rate parameters, $e_1$ and $e_2$, are associated with the entry phases, which may or may not include dependent verification. Following Definitions 2.1a and 2.1b, $X_1 = \sum X_{1i}$ and $X_2 = \sum X_{2i}$ are the number of records with entry errors; $X_1 \sim B(N, e_1)$ and $X_2 \sim B(N, e_2)$. Since Secondary Entry is performed without knowledge of Primary Entry results, all $X_{1i}$ and $X_{2i}$ are assumed independent.

3.2.1.2 Valid Value Edit

In the VVE phase, the first count of interest is the number of records, $V_1$, in which at least one field does not pass its valid value edit. These are the records that require additional processing before the next phase.

Let the conditional parameters $d_{11}, d_{01}, d_{10}, d_{00}$ be defined as

- $d_{11} = P(\text{VVE error detected | error exists}) = P(V_{1i} = 1 | X_{1i} = 1)$
- $d_{01} = P(\text{VVE error not detected | error exists}) = P(V_{1i} = 0 | X_{1i} = 1)$
- $d_{10} = P(\text{VVE error detected | no error exists}) = P(V_{1i} = 1 | X_{1i} = 0) = 0$
- $d_{00} = P(\text{VVE error not detected | no error exists}) = P(V_{1i} = 0 | X_{1i} = 0) = 1$
The conditional probability, $d_{10}$, that an error is detected given that no error exists can be assumed to be zero. This describes the situation in which all valid value edits are based on appropriate limits since the original input data are assumed correct. Under these conditions, if a record is not an error then all its correct values cannot fail their valid value edits. However, a record that passes all valid value edits can still include switch errors. The parameter $d_{01}$ is the conditional probability that switch errors occurs during data entry.

**Theorem 3.2.1.** Let $V_i = \sum V_{i1} \sim B(N, v_{21})$, assuming $V_i \sim B(1, v_{21})$ are i.i.d. and $V_{i1} = \beta$(error detected in primary record $i$ by Valid Value Edit). Using the parameter definitions above, assuming $d_{10} = 0$ and that VVE is included in a subsystem, then

$$v_{21} = d_{11} e_1.$$  \hspace{1cm} (3.2.1)

If VVE is not done, then $d_{11} = 0$ and $V = 0$ by definition.

**Proof:**

$v_{21} = P(V_{i1} = 1)$

$= P(\text{invalid value detected})$

$= P(\text{detected} \& \ (\text{no error exists or error exists}))$

$= P(\text{detected} \& \ \text{no error exists}) + P(\text{detected} \& \ \text{error exists})$

$= 0 + P(\text{detected} \& \ \text{error exists})$

$= P(\text{detected} \mid \text{error exists})P(\text{error exists})$

$= P(V_{i1} = 1 \mid X_{i1} = 1)P(X_{i1} = 1)$

$= d_{11} e_1$
3.2.1.3 Valid Value Edit Process Update

The second count of interest for the VVE phase is the number of records, $F_1$, that must be updated because a correction is found. Assuming correct input data and appropriate valid value edits, then the probability that a record requires updating is the same as the probability that an error is detected in a record. In this situation, $F_1 = V_1$ since an error must exist if an error is detected by VVE.

**Theorem 3.2.2** Let $F_1 = \sum F_{1i} \sim B(N, f_{21})$, assuming $F_{1i} \sim B(1, f_{21})$ are i.i.d. and $F_{1i} = \beta($error found in record i after error detected by Valid Value Edit of primary records$)$. Assuming $d_{10} = 0$, correct input data and appropriate edits, then $f_{21} = v_{21}$.

**Proof:**

Assuming that original input data are correct and appropriate edits are used implies that

$g_{21} = P(\text{error found | VVE error detected & error exists})$

$= P(F_{1i} = 1 | V_{1i} = 1 & X_{1i} = 1)$

$= 1$

and $P(\text{error exists | VVE error detected}) = P(X_{1i} = 1 | V_{1i} = 1) = 1$.

Thus,

$f_{21} = P(F_{1i} = 1)$

$= P(\text{found & (detected or undetected)})$

$= P(\text{found & detected}) + 0$

$= P(\text{found & detected & (error or no error)})$

$= P(F_{1i} = 1 | V_{1i} = 1 & X_{1i} = 1) P(X_{1i} = 1 | V_{1i} = 1) P(V_{1i} = 1)$

$= 1 \times 1 \times P(V_{1i} = 1)$

$= v_{21}$. 
3.2.1.4 Errors after Primary Processing

To simplify the remaining derivations, let us define $X_3$ as the number of error records after Primary Processing. In Model 2 Primary Processing includes data entry, dependent verification, and valid value editing. Data entry must be performed in any subsystem; the latter two steps are optional.

**Theorem 3.2.3.** Let $X_3 = \sum X_{3i} \sim B(N, e_{23})$, assuming $X_{3i} \sim B(1, e_{23})$ are i.i.d., $X_{3i} = \beta$(error exists in primary record $i$ after Primary Processing). Assuming $d_{10} = 0$ and corrections are always found when an error is detected by VVE, then

$$e_{23} = (1 - kd_{11})e_1,$$

where $k = \text{Prob}(\text{proper correction found} | \text{detected & correction found})$.

If VVE is not done, then all $X_{3i} = X_{1i}$ and $X_3 = X_1$, by definition $d_{11} = 0$ and so $e_{23} = e_1$.

**Proof:**

$$e_{23} = P(X_{3i} = 1)$$

$$= P(\text{error remaining after VVE})$$

$$= P(\text{error undetected by VVE or improper update})$$

$$= P(V_{1i} = 0 & X_{1i} = 1)$$

$$+ P(\text{imp update & detected & found})P(\text{detected & found})$$

$$= P(V_{1i} = 0 | X_{1i} = 1)P(X_{1i} = 1) + (1 - k)P(V_{1i} = 1)$$

$$= d_{01}e_1 + (1 - k)d_{11}e_1$$

$$= (1 - kd_{11})e_1$$

since $d_{01} = 1 - d_{11}$. 
Lemma 3.2.1. Assuming all $X_{1i}$ and $X_{2i}$ are independent, and $V_{1i}$ and $X_{2i}$ are independent when VVE is performed, then $X_{3i}$ and $X_{2i}$ are independent.

Proof:
If no VVE is performed, then clearly all $X_{3i} = X_{1i}$ and so is independent of $X_{2i}$.

If VVE is performed, the assumption that all $V_i$ and $X_{2i}$ are independent is reasonable since Primary and Secondary Processing are separate. Thus, all $X_{3i}$ and $X_{2i}$ are independent since $X_{3i}$ is a function of $X_{1i}$ and $V_i$, which are both independent of $X_{2i}$.

Primary records produced by Primary Processing in Model 2 are analogous to primary records produced by Primary Entry in Model 1. Following Theorem 3.2.3 and Lemma 3.2.1, $X_{3i}$ is analogous to $X_{1i}$ in Model 1. The theorems for the remaining phases of Model 2 are based directly on the corresponding theorems for Model 1.

3.2.1.5 Comparison
The comparison phase involves records from both Primary Processing and Secondary Entry. The count of interest, $W$, is how many corresponding records do not match and so require further checking to find corrections. If Secondary Entry is not in a subsystem, $W = 0$ by definition.

For this step, a conditional probability parameter analogous to $m_1$ in Model 1 is needed: $m_2$ is defined as the probability that, given an error exists in each corresponding record, the paired records are identical ($m_2 = P(W_i = 0 | X_{3i} = 0 & X_{4i} = 0)$).
Theorem 3.2.4. Let \( W = \sum W_i - B(N, w_2) \), assuming \( W_i \sim B(1, w_2) \) are i.i.d. and \( W_i = \beta \) (no match for record \( i \) at Comparison). Assuming both Primary and Secondary Entry are performed, and all \( X_{1i} \) and \( X_{2i} \) are independent, then

\[
\begin{align*}
    w_2 &= e_{23} + e_2 - (1 + m_2)e_{23}e_2 \\
    &= (1 - kd_{11})e_1 + e_2 - (1 + m_2)(1 - kd_{11})e_1e_2
\end{align*}
\]

where \( m_2 = P(\text{match 1 errors exist in both Primary and Secondary}) \).

If no VVE, \( d_{11} = 0 \) by definition so

\[
    w_2 = e_1 + e_2 - (1 + m_2)e_1e_2.
\]

Proof:

Using Lemma 3.2.1 (\( X_{3i} \) and \( X_{1i} \) independent) and substituting \( X_{3i} \) for \( X_{1i} \) and \( e_{23} \) for \( e_1 \) in the proof of Theorem 2.1 (Comparison in Model 1) yields

\[
\begin{align*}
    w_2 &= P(W_i = 1) \\
    &= e_{23} + e_2 - (1 + m_2)e_{23}e_2 \\
    &= (1 - kd_{11})e_1 + e_2 - (1 + m_2)(1 - kd_{11})e_1e_2 \quad \text{by Equation 3.2.2.}
\end{align*}
\]

3.2.1.6 Comparison Correction

As in Model 1, let the random variable \( Y \) be defined as the number of records where errors are found after Comparison. The process of finding corrections is done for primary records when corresponding secondary records do not match. The probability of detecting an error when none exists is assumed to be zero \( P(Y_i = 1 \mid X_{3i} = 0 \& W_i = 1) = 0 \).

Let the conditional parameter, \( c_2 \), be defined as the probability of finding a correction after Comparison \( (c_2 = P(Y_i = 1 \mid X_{3i} = 1 \& W_i = 1)) \). This is analogous to the parameter \( c \) in Model 1.
Theorem 3.2.5. Let $Y = \sum Y_i \sim B(N, y_2)$, assuming $Y_i \sim B(1, y_2)$ are i.i.d. and $Y_i = \beta$(error found for record $i$ after Comparison).

When Comparison is done after Primary Processing and Secondary Entry are performed, assuming $X_{1i}$ and $X_{2i}$ are independent,

$$y_2 = c_2 \left[ e_{23}(1 - e_2) + (1 - m_2)e_{23}e_2 \right]$$

$$= c_2 \left[ (1 - kd_{11})e_1(1 - e_2) + (1 - m_2)(1 - kd_{11})e_1e_2 \right].$$

By definition,

$$1 - m_2 = P(\text{no match } | X_{3i} = 1 \& X_{2i} = 1) = P(W_i = 1 | X_{3i} = 1 \& X_{2i} = 1).$$

If no VVE is used, then

$$y_2 = c_2 \left[ e_1(1 - e_2) + (1 - m_2)e_1e_2 \right].$$

Proof:

Using Lemma 3.2.1 ($X_{3i}$ and $X_{1i}$ independent) and substituting $X_{3i}, e_{23}, m_2$ and $c_2$ where appropriate in the proof of Theorem 2.3 (Secondary Proofread in Model 1) yields

$$y_2 = P(Y_i = 1)$$

$$= c_2[e_{23}(1 - e_2) + (1 - m_2)e_{23}e_2]$$

$$= c_2[(1 - kd_{11})e_1(1 - e_2) + (1 - m_2)(1 - kd_{11})e_1e_2] \quad \text{by Equation 3.2.2.}$$
3.2.1.7 Manual Proofread

For Manual Proofread, the count of interest, $U$, is the number of records where errors are detected. In this edit process, corrections are found as errors are detected. If Manual Proofread is not performed, then $U = 0$ by definition.

Let the conditional error detection probabilities for Manual Proofread be represented by the parameters $r_{211}, r_{201}, r_{210}, r_{200}$:

- $r_{211} = P(\text{detected | error exists & proofread}) = P(U_i = 1 | X_{3i} = 1 & W_i = 0)$
- $r_{201} = P(\text{not detected | error exists & proofread}) = P(U_i = 0 | X_{3i} = 1 & W_i = 0)$
- $r_{210} = P(\text{detected | no error exists & proofread}) = P(U_i = 1 | X_{3i} = 0 & W_i = 0)$
- $r_{200} = P(\text{not detected | no error exists & proofread}) = P(U_i = 0 | X_{3i} = 0 & W_i = 0)$

**Theorem 3.2.6.** Let $U = \sum U_i \sim B(N, u_2)$, assuming $U_i \sim B(1, u_2)$ are i.i.d. and $U_i = \beta(\text{error detected for record i at Manual Proofread})$. Depending on the subsystem, $u_2 = u_{21}$ or $u_2 = u_{22}$ as follows.

If only Primary processing is used, then $u_2 = u_{21}$, where

$$u_{21} = r_{210}(1 - e_{23}) + r_{211}e_{23}.$$  \hspace{1cm} (3.2.5)

If no VVE is used, then

$$u_{21} = r_{210}(1 - e_{1}) + r_{211}e_{1}.$$

If Manual Proofread is performed when both Primary processing and Secondary Entry are used, then a primary record is proofread only if the corresponding secondary record matches. Assuming all $X_{1i}$ and $X_{2i}$ are independent, then $u_2 = u_{22}$, where

$$u_{22} = r_{210}(1 - e_{23})(1 - e_{2}) + r_{211}m_2 e_{23}e_{2}.$$ \hspace{1cm} (3.2.6)

If no VVE is used, then

$$u_{22} = r_{210}(1 - e_{1})(1 - e_{2}) + r_{211}m_2 e_{1}e_{2}.$$
Proof:

Using Lemma 3.2.1 \((X_{3i} \text{ and } X_{1i} \text{ independent})\) and substituting \(X_{3i}, e_{23}, \text{ and } m_2\) in the proof of Theorem 2.2 (Primary Proofread in Model 1) yields the results.

3.2.1.8 Errors in Final Dataset

The errors remaining in the final dataset are those undetected by all editing processes or the result of improper updating. The number of undetected errors obviously depends on which components are included in a subsystem.

There are four mutually exclusive events in the overall system of Model 2 that lead to undetected errors in the final dataset. Let PP stand for Primary Processing. The possibilities are:

(i) an error is undetected after PP and Manual Proofread;
(ii) an error is undetected after PP and Comparison correction;
(iii) an update due to Manual Proofread introduces an error;
(iv) an update due to Comparison introduces an error.

The possibility that an update introduces an error is allowed by defining the parameter \(k\) as the probability that an update results in a proper correction. Note that \(k\) does not depend on which edit process is used.
Theorem 3.2.7. Let \( Z = \sum Z_i \sim B(N, z_2) \), assume \( Z_i \sim B(1, z_2) \) are i.i.d. and \( Z_i = \beta \) (error for record i in final data). For all subsystems the following holds:

\[
\begin{align*}
Z_2 & = P(Z_1 = 1) \\
& = P(\text{undetected error after PP and Manual Proofread}) \\
& \quad + P(\text{undetected error after PP and Comparison correction}) \\
& \quad + P(\text{improper update after Manual Proofread}) \\
& \quad + P(\text{improper update after Comparison correction}) \\
& = P(\text{undetect. MP}) + P(\text{undetect. CC}) + P(\text{imp MP}) + P(\text{imp CC})
\end{align*}
\]

Let \( k = \text{Prob(proper correction I correction found)} \).

For the different subsystems of interest, \( z_2 = z_{21} \) or \( z_2 = z_{22} \) or \( z_2 = z_{23} \), depending on the components included.

For a simple subsystem with VVE only, no Secondary Entry and no Manual Proofread, \( z_2 = z_{21} \), where

\[
Z_{21} = e_{23} = (1 - kd_{11})e_1.
\] (3.2.7)

For a subsystem in which Redundancy (and Secondary Entry) is not used, so that only Manual Proofread is possible after Primary Processing, \( Z_2 = z_{22} \), where

\[
Z_{22} = r_{201} e_{23} + (1 - k)[r_{210}(1 - e_{23}) + r_{211} e_{23}].
\] (3.2.8)

If no VVE is used, \( d_{11} = 0 \), then

\[
Z_{22} = r_{201} e_1 + (1 - k)[r_{210}(1 - e_1) + r_{211} e_1].
\]

When Manual Proofread is not used, Secondary Entry is done and the Redundancy phase is included after Primary Processing, then \( Z_2 = z_{23} \), where

\[
Z_{23} = e_{23} - c_2 k (e_{23} - m_2 e_{23} e_2).
\] (3.2.9)

If no VVE is used, then

\[
Z_{23} = e_1 - c_2 k (e_1 - m_2 e_1 e_2).
\]
Proof of 3.2.7

The simplest subsystem has only VVE, no Secondary Entry, no Comparison, and no Manual Proofread.

\[ z_{21} = P(Z_1 = 1) \]

\[ = P(\text{undetect. MP}) + P(\text{undetect. CC}) + P(\text{imp MP}) + P(\text{imp CC}) \]

\[ = P(\text{undetect. MP}) + 0 + 0 + 0 \]

\[ = P(\text{error after PP} \& \text{und. MP}) \]

\[ = P(\text{und. MP} | \text{error after PP}) P(\text{error after PP}) \]

\[ = 1 \times P(\text{error after PP}) \]

\[ = P(X_{3i} = 1) \]

\[ = (1 - kd_{1i})e_1 \]

by Theorem 3.2.3.

Proof of 3.2.8:

When only Manual Proofread is possible after Primary Processing, no Secondary Entry or Comparison is included, then the subsystem is analogous to the Standard subsystem in Model 1. Thus, substituting the appropriate parameters in Theorem 2.4 for the Standard subsystem yields the result.

Proof of 3.2.9:

When only Comparison is done after Primary Processing (and Secondary Entry), no Manual Proofread is included, then the subsystem is analogous to the Redundancy subsystem in Model 1. Thus, using Lemma 3.2.1 and substituting the appropriate parameters in Theorem 2.4 for Redundancy yields the result.
3.2.2 Cost Estimation

Estimating subsystem costs in Model 2 expands on the ideas used in Model 1. The objective is to allow useful subsystem comparisons assuming cost estimates for the different phases are available. Development costs are fixed regardless of the number of records to be processed, while processing costs are estimated as the average costs for one record. Total cost estimates are defined using cost parameters and the count random variables. The inclusion or exclusion of a particular phase in a subsystem is accomplished by defining appropriate indicator variables.

For Model 2, we define the following cost parameters.

<table>
<thead>
<tr>
<th>System Cost Component</th>
<th>Development Cost</th>
<th>Processing Cost per Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)    Entry (Primary, Secondary)</td>
<td>$D_{21}$</td>
<td>$P_{21}$</td>
</tr>
<tr>
<td>(2)    Dep. Verification (Primary, Secondary)</td>
<td>$D_{22}$</td>
<td>$P_{22}$</td>
</tr>
<tr>
<td>(3)    Find correction (VVE, Comparison)</td>
<td>$D_{23}$</td>
<td>$P_{23}$</td>
</tr>
<tr>
<td>(4)    Update (VVE, Comparison, Manual Pf.)</td>
<td>$D_{24}$</td>
<td>$P_{24}$</td>
</tr>
<tr>
<td>(5)    Value Value Edit</td>
<td>$D_{25}$</td>
<td>$P_{25}$</td>
</tr>
<tr>
<td>(6)    Comparison</td>
<td>$D_{26}$</td>
<td>$P_{26}$</td>
</tr>
<tr>
<td>(7)    Manual Proofread</td>
<td>$D_{27}$</td>
<td>$P_{27}$</td>
</tr>
<tr>
<td>(8)    Create analysis file</td>
<td>$D_{28}$</td>
<td>$P_{28}$</td>
</tr>
</tbody>
</table>

Let $b = \beta$(Valid Value Edit process done).

Let $s = \beta$(Redundancy used: Secondary Entry and Comparison done).

Let $t = \beta$(Manual Proofread process done).

Let $a = \text{adjustment factor for intelligent data entry; range zero to one.}$
FIGURE 3.2.4
Model 2 Overall System Cost Components

INPUT

(5) Valid value edit
- no error detected
- error detected

V

(3) Find correction
(4) Update

N - V

(1) Primary Data Entry
(2) Primary Verification

N - V

(6) Comparison
- match
- no match

N - W

(7) Manual Proofread
- error detected
- no error detected

N - W - U

(4) Update

(3) Find correction
(4) Update

W - Y

(1) Secondary Data Entry
(2) Secondary Verification

primary

(8) Create analysis file

FINAL

Cost Components
(1) Data entry
(2) Dependent verification
(3) Finding corrections
(4) Update
(5) Valid value edit program
(6) Comparison
(7) Manual Proofread
(8) Create analysis file

Record counts
N Total number of records
V VVE detected errors
W Comparison non-matches
U Manual Pf. detected errors
Y Comparison corrections
Figure 3.2.4 shows how the eight cost components describe the overall system. Costs for each component type are assumed to be the same regardless of which phase is involved. For instance, the processing cost of data entry does not depend on whether Primary or Secondary Entry is being considered. When redundancy is used, development for entry occurs only once. The indicator variables \((b, s, t)\) make it easy to get a subsystem cost from the definition of total cost for the overall system.

The adjustment factor, \(a\), is needed to allow for intelligent data entry. Processing costs are clearly lower when corrections are found and records updated during data entry. Batch processing after entry is completed requires locating original data and may require separate update processing. When batch processing is used, then \(a = 1\). With intelligent data entry, the adjustment factor is between zero and one.

The components included in total development cost for all subsystems, assuming that at least one edit phase is used, are: entry, finding corrections, updating, and creating a final dataset. By multiplying indicator parameters by their associated cost parameters the development cost for a particular edit phase is added only when appropriate.

For each phase, the total processing cost depends on the average cost and the number of records processed. As with development cost, the indicator parameters provide an easy method for including the appropriate edit phase processing costs. For entry, the processing cost is simply the total number of records, \(N\), multiplied by the average cost of entering a record. A value for average cost can be estimated by considering how many records can be entered in a given time period assuming the cost for that time period is known. For VVE, the adjustment factor is applied to the correction and update processes. Comparison and Manual Proofread involve straightforward calculations using the appropriate cost parameters and count random variables. The creation of a final dataset is usually more of a batch process with a fixed cost, but can be defined using an average cost.
The cost of data entry depends on whether or not dependent verification is included in a subsystem.

Let \( D_{2E} = \begin{cases} D_{21} + D_{22} & \text{if dependent verification is performed}, \\ D_{21} & \text{if not}. \end{cases} \)

Let \( P_{21P} = \begin{cases} P_{21} + P_{22} & \text{if dependent verification is performed}, \\ P_{21} & \text{if not}. \end{cases} \)

Let \( P_{21U} \) be defined similarly to \( P_{21P} \).

The development cost of finding corrections and updating records depends on whether or not this process is part of intelligent data entry. There are essentially no separate development costs if intelligent data entry is used since training for handling errors would be part of data entry development; in this case \( D_{2U} = 0 \). When separate batch processes are used for finding corrections, by first locating the original input data, and updating records, then

\[
D_{2U} = D_{23} + D_{24}.
\]

**Definition 3.2.** Total cost is a combination of development and processing costs for all phases included in a subsystem. Figure 3.2.4 shows how the count random variables relate to the cost parameters.

Let \( T_2 = D_2 + P_2 \) where

\[
D_2 = D_{2E} + D_{2U} + bD_{25} + sD_{26} + tD_{27} + D_{28},
\]

\[
P_2 = NP_{21P} + sNP_{21U} + b(NP_{25} + aV_1P_{23} + aF_1P_{24})
\]

\[
+ s(NP_{26} + WP_{23} + YP_{24}) + t((N - W)P_{27} + UP_{24}) + NP_{28}.
\]
For the five subsystems of interest, the subsystem costs derived from the definition of total cost are as follows.

In the first subsystem, only Primary Entry and VVE are used so \( b = 1, \ s = 0, \) and \( t = 0: \)

\[
T_{21} = D_{2E} + D_{2U} + D_{25} + D_{28} + NP_{21P} + (NP_{25} + aV_1P_{23} + aF_1P_{24}) + NP_{28}.
\]

When VVE (\( b = 1 \)) is combined with Redundancy (\( s = 1, \ t = 0 \)) or Manual Proofread (\( s = 0, \ t = 1 \)), then the following apply. When Comparison is not done, \( W = 0 \) by definition.

\[
T_{22} = D_{2E} + D_{2U} + D_{25} + D_{26} + D_{28} + NP_{21P} + NP_{21S}
\]

\[
+ (NP_{25} + aV_1P_{23} + aF_1P_{24}) + (NP_{26} + WP_{23} + YP_{24}) + NP_{28}.
\]

\[
T_{23} = D_{2E} + D_{2U} + D_{25} + D_{27} + D_{28}
\]

\[
+ NP_{21P} + (NP_{25} + aV_1P_{23} + aF_1P_{24}) + (NP_{27} + UP_{24}) + NP_{28}.
\]

The last two subsystems do not include VVE (\( b = 0 \)). Only one edit phase, Comparison for Redundancy (\( s = 1, \ t = 0 \)) or Manual Proofread (\( s = 0, \ t = 1 \)), is used.

\[
T_{24} = D_{2E} + D_{2U} + D_{26} + D_{28} + NP_{21P} + NP_{21S} + (NP_{26} + WP_{23} + YP_{24}) + NP_{28}.
\]

\[
T_{25} = D_{2E} + D_{2U} + D_{27} + D_{28} + NP_{21P} + (NP_{27} + UP_{24}) + NP_{28}.
\]
FIGURE 3.3.1
Model 3 Overall System and Random Variables

N
INPUT

Primary Data Entry
Primary Verification

X₁

Valid Value Edit
no error detected
N - V₁

error detected
Find correction
V₁
Update
F₁

Secondary Data Entry
Secondary Verification

X₂

Valid Value Edit
no error detected
N - V₂

error detected
Find correction
V₂
Update
F₂

primary
X₃

Comparison
no match
W
match
N - W

Manual Proofread
no error detected
(N - W) - U

error detected
U
Update

Find correction
no correction
Y
Update
W - Y

Create analysis file

FINAL

N

Final error rate
Z
N

X₁ Primary entry errors
X₂ Secondary entry errors
X₃ Primary Processing errors
X₄ Secondary Processing errors
V₁, V₂ VVE detected errors
F₁, F₂ VVE corrections
W Comparison non-matches
U Manual Proofread detected errors
Y Comparison corrections
Z Undetected errors
FIGURE 3.3.2
Model 3 Subsystems
3.3 Model 3 – VVE in both Primary and Secondary Processing

A valid value edit process after Secondary Entry is added to Model 2 to create Model 3. As shown in Figure 3.3.1, the phases following Primary and Secondary Processing are the same (Comparison, Manual Proofread). The Model 3 subsystems are very similar those of Model 2. The advantage is that comparing RDM situations that use intelligent data entry is more appropriate with VVE allowed after both entry phases. When VVE is not included, Model 3 describes the same situation as Model 2. Thus, only subsystems with VVE, Figure 3.3.2, are of interest.

3.3.1 Model 3 Error Generation Derivations

The error generation process in Model 3 is a direct expansion of Model 2.

3.3.1.1 Data Entry

As in Model 1, the results of the Primary and Secondary Entry phases are represented by two random variables, $X_1 \sim B(N, e_1)$ and $X_2 \sim B(N, e_2)$, that are the number of records with entry errors. All $X_{1i}$ and $X_{2i}$ are assumed independent.

3.3.1.2 Primary Valid Value Edit

For a VVE phase, the first count of interest is the number of records, $V_1$, in which at least one field does not pass its valid value edit. These are the primary records that require additional processing before the next phase.

Let the conditional detection probability parameters $d_{3111}$, $d_{3110}$, be defined similarly to Model 2. The conditional probability of detecting an existing error is $d_{3111}$. As in Model 2, the conditional probability of detecting an error when none exists, $d_{3110}$, is assumed to be zero.
Theorem 3.3.1. Let \( V_1 = \sum V_{1i} \sim B(N, v_{31}) \), assuming \( V_{1i} \sim B(1, v_{31}) \) are i.i.d. and \( V_{1i} = \beta(\text{error detected in record } i \text{ by Valid Value Edit}) \). Let \( d_{3111} = P(V_{1i} = 1 \mid X_{1i} = 1) \). Assuming \( d_{3110} = 0 \) and that VVE is included in a subsystem, then

\[
v_{31} = d_{3111} e_1. \tag{3.3.1}
\]

If Primary VVE is not done, then \( d_{3111} = 0 \) and \( V_1 = 0 \) by definition.

Proof:

Following Theorem 3.2.1 (VVE in Model 2),

\[
v_{31} = P(V_{1i} = 1)
\]

\[
= P(V_{1i} = 1 \mid X_{1i} = 1) P(X_{1i} = 1)
\]

\[= d_{3111} e_1.\]

3.3.1.3 Primary Valid Value Edit Process Update

The second count of interest for VVE is the number of records, \( F_1 \), that must be updated because a correction is found. As in Model 2, appropriate edits are assumed so that \( F_1 = V_1 \), since an error must exist if an error is detected by a valid value edit.

Theorem 3.3.2. Let \( F_1 = \sum F_{1i} \sim B(N, f_{31}) \), assuming \( F_{1i} \sim B(1, f_{31}) \) are i.i.d. and \( F_{1i} = \beta(\text{error found in record } i \text{ after error detected by Valid Value Edit}) \). Assuming \( d_{3110} = 0 \) (correct input data and appropriate edits) then \( f_{31} = v_{31} \).

Proof:

Substituting \( F_{1i} \) and \( V_{1i} \) in the proof of Theorem 3.2.2 yields the result.
3.3.1.4 Secondary Valid Value Edit Process

The Secondary VVE phase is very similar to the Primary VVE phase. The counts of interest are: 1) the number of records, \( V_2 \), in which at least one field does not pass its valid value edit, and 2) the number, \( F_2 \), that need updating.

*Theorem 3.3.3.* Let \( V_2 = \sum V_{2i} \sim B(N, v_{32}) \), assuming \( V_{2i} \sim B(1, v_{32}) \) are i.i.d. and \( V_{2i} = \beta(\text{error detected in record } i \text{ by Valid Value Edit}) \). Let \( d_{3211} = P(V_{2i} = 1 \mid X_{1i} = 1) \) and assume that \( d_{3210} = P(V_{2i} = 1 \mid X_{1i} = 0) = 0 \). When VVE is included in a subsystem, then

\[
v_{32} = d_{3211} e_2.
\]  

(3.3.2)

If Secondary VVE is not done, then \( d_{3211} = 0 \) and \( V_2 = 0 \) by definition.

*Proof:*

Using Theorem 3.3.1, substituting appropriately.

*Theorem 3.3.4.* Let \( F_2 = \sum F_{2i} \sim B(N, f_{32}) \), assuming \( F_{2i} \sim B(1, f_{32}) \) are i.i.d. and \( F_{2i} = \beta(\text{error found in record } i \text{ after error detected by Valid Value Edit}) \). Assuming \( d_{3210} = 0 \), correct input data and appropriate edits, then \( f_{32} = v_{32} \).

*Proof:*

Following Theorem 3.3.2 and substituting \( F_{1i} \) and \( V_{1i} \) yields the result.
3.3.1.5 Errors after Primary and Secondary Processing

Define $X_3$ and $X_4$ as the number of error records after Primary Processing and Secondary Processing, respectively.

**Theorem 3.3.5.** Let $X_3 = \sum X_{3i} \sim B(N, e_{33})$, assuming $X_{3i} \sim B(1, e_{33})$ are i.i.d., $X_{3i} = \beta($error exists in record $i$ after Primary Processing$)$. Assuming $d_{3110} = 0$ and corrections are always found when an error is detected by Primary VVE, then

$$e_{33} = (1 - kd_{3111})e_1$$  \hspace{1cm} (3.3.3)

where $k = \text{Prob(proper correction | detected & correction found)}$.

If Primary VVE is not done, then all $X_{3i} = X_{1i}$ and $X_3 = X_1$, and

$$e_{33} = e_1.$$

**Theorem 3.3.6.** Let $X_4 = \sum X_{4i} \sim B(N, e_{34})$, assuming $X_{4i} \sim B(1, e_{34})$ are i.i.d., $X_{4i} = \beta($error exists in record $i$ after Secondary Processing$)$. Assuming $d_{3210} = 0$ and corrections are always found when an error is detected by Secondary VVE, then

$$e_{34} = (1 - kd_{3211})e_2$$  \hspace{1cm} (3.3.4)

where $k = \text{Prob(proper correction | detected & correction found)}$.

If Secondary VVE is not done, then all $X_{4i} = X_{2i}$ and $X_4 = X_2$, then

$$e_{34} = e_{32}.$$

If Secondary Entry is not included, then $X_4 = 0$ by definition.

**Proof of Theorems 3.3.5 and 3.3.6:**

Follow Theorem 3.2.3 (Model 2 Primary Processing) substituting appropriately.
Lemma 3.3.1. Assuming all $X_{1i}$ and $X_{2i}$ are independent, then all $X_{3i}$ and $X_{4i}$ are independent. If VVE is performed as part of Primary or Secondary Processing, assume that all $X_{1i}$, $X_{2i}$, $V_{1i}$, and $V_{2i}$ are mutually independent.

Proof:

If no VVE is performed, then clearly all $X_{3i} = X_{1i}$ and $X_{4i} = X_{2i}$. Thus, $X_{3i}$ and $X_{4i}$ are independent since $X_{1i}$ and $X_{2i}$ are independent.

If VVE is performed, the assumption of mutual independence is reasonable since Primary and Secondary processing are separate. Thus, all $X_{3i}$ and $X_{4i}$ are independent since $X_{3i}$ is a function of $X_{1i}$ and $V_{1i}$, $X_{4i}$ is a function of $X_{2i}$ and $V_{2i}$, and those random variables are independent.

Primary records produced by Primary Processing in Model 3 are analogous to primary records produced by Primary Entry in Model 1. The analogy is similar for secondary records. Using Theorems 3.3.5, 3.3.6 and Lemma 3.3.1, $X_{3i}$ is analogous to $X_{1i}$ and $X_{4i}$ is analogous to $X_{2i}$ in Model 1. Thus, the theorems for the remaining phases of Model 3 are based directly on the corresponding theorems for Model 1.
3.3.3.6 Comparison

**Theorem 3.3.7.** Let \( W = \sum W_i \sim B(N, w_3) \), assuming \( W_i \sim B(1, w_3) \) are i.i.d. and \( W_i = \beta \) (no match for record \( i \) at Comparison). Assuming both Primary and Secondary Processing are performed, and all \( X_{1i} \) and \( X_{2i} \) are independent, then

\[
w_3 = e_{33} + e_{34} - (1 + m_3)e_{33}e_{34}
\]

where \( m_3 = P( W_i = 0 \mid X_{3i} = 1 \& X_{4i} = 1) = P(\text{match both errors}) \). If Primary VVE is not in a subsystem, then \( e_{33} = e_1 \). Similarly, \( e_{34} = e_2 \) if Secondary Entry is done without Secondary VVE.

**Proof:**

Using Lemma 3.3.1 (\( X_{3i} \) and \( X_{4i} \) independent) and substituting appropriately in the proof of Theorem 2.1 (Comparison in Model 1) yields the result.
3.3.1.7 Comparison Correction

As in Model 1, the random variable $Y$ is defined as the number of records where errors are found after Comparison. The process of finding corrections is done only for primary records when corresponding secondary records do not match. The probability of detecting an error when none exists is assumed to be zero

$$P(Y_i = 1 \mid X_{3i} = 0 \& W_i = 1) = 0.$$

Let the conditional parameter, $c_3$, be defined as the probability of finding a correction after Comparison ($c_3 = P(Y_i = 1 \mid X_{3i} = 1 \& W_i = 1)$).

*Theorem 3.3.8.* Let $Y = \sum Y_i \sim B(N, \gamma_3)$, assuming $Y_i \sim B(1, \gamma_3)$ are i.i.d. and $Y_i \sim B$ (error found for record $i$ after Comparison).

When both Primary Processing and Secondary Processing are performed, which implies Comparison is done, then, assuming all $X_{1i}$ and $X_{2i}$ are independent,

$$\gamma_3 = c_3[\epsilon_{33}(1 - \epsilon_{34}) + (1 - m_3)\epsilon_{33}\epsilon_{34}].$$

(3.3.6)

By definition, $1 - m_3 = P(W_i = 1 \mid X_{3i} = 1 \& X_{4i} = 1)$. If Primary VVE is not in a subsystem, then $\epsilon_{33} = \epsilon_1$. Similarly $\epsilon_{34} = \epsilon_2$ if Secondary VVE is not included.

*Proof:*

Use Lemma 3.3.1 ($X_{3i}$ and $X_{4i}$ independent) and substituting where appropriate in the proof of Theorem 2.3 (Secondary Proofread in Model 1).
3.3.1.8 Manual Proofread

For Manual Proofread, the count of interest, $U$, is the number of records where errors are detected. In this edit process, corrections are found as errors are detected. If Manual Proofread is not performed, then $U = 0$ by definition.

Let the conditional error detection probabilities for Manual Proofread be represented by the parameters $r_{311}, r_{301}, r_{310}, r_{300}$.

**Theorem 3.3.9.** Let $U = \sum U_i \sim B(N, u_3)$, assuming $U_i \sim B(1, u_3)$ are i.i.d. and $U_i = \beta$(error detected for record $i$ at Manual Proofread). Depending on the subsystem, $u_3 = u_{31}$ or $u_3 = u_{32}$ as follows.

If only Primary Processing is used, then $u_3 = u_{31}$, where

$$u_{31} = r_{310}(1 - e_{33}) + r_{311}e_{33}. \tag{3.3.7}$$

If Manual Proofread is performed when both Primary Processing and Secondary Processing are used, then a primary record is proofread only if the corresponding secondary record matches. Assuming all $X_{1i}$ and $X_{2i}$ are independent, then $u_3 = u_{32}$, where

$$u_{32} = r_{310}(1 - e_{33})(1 - e_{34}) + r_{311}m_3e_{33}e_{34}. \tag{3.3.8}$$

**Proof:**

Using Lemma 3.3.1 ($X_{3i}$ and $X_{4i}$ independent) and substituting in the proof of Theorem 3.2.6 (Manual Proofread in Model 2) yields the result.
3.3.1.9 Errors in Final Dataset

The errors that remain in the final dataset are those undetected by all editing processes or the result of improper updating. The number of undetected errors obviously depends on which components are included in a subsystem.

*Theorem 3.3.10.* Let \( Z = \sum Z_i \sim B(N, z) \), assume \( Z_i \sim B(1, z) \) are i.i.d. and \( Z_i = \beta(\text{error for record } i \text{ in final data}) \). For all subsystems the following holds:

\[
z = P(Z_i = 1) = P(\text{undetected error after Primary Processing and Manual Proofread}) \\
+ P(\text{undetected error after Primary Proc. and Comparison correction}) \\
+ P(\text{improper update after Manual Proofread}) \\
+ P(\text{improper update after Comparison correction}).
\]

Let \( k = \text{Prob}(\text{proper correction} | \text{correction found}) \).

For the different subsystems of interest, \( z = z_{31} \) or \( z = z_{32} \) or \( z = z_{33} \), depending on the components included.

For a simple subsystem with Primary VVE only, no Secondary Processing and no Manual Proofread, \( z = z_{31} \), where

\[
z_{31} = e_{33} = (1 - kd_{3111})e_1. 
\]

(3.3.9)

For a subsystem that does not include Secondary Processing and Comparison, which implies only Manual Proofread is possible after Primary Processing, \( z = z_{32} \), where

\[
z_{32} = r_{301}e_{33} + (1 - k)[r_{310}(1 - e_{33}) + r_{311}e_{33}].
\]

(3.3.10)

When Manual Proofread is not used, but Secondary Processing is done and Comparison is included after Primary Processing, then \( z = z_{33} \), where

\[
z_{33} = e_{33} - c_3 k(e_{33} - m_3 e_{33}e_{34}).
\]

(3.3.11)
Proof:

Substitute appropriately in Theorem 3.2.7 (undetected errors in Model 2).

3.3.2 Cost Estimation

Estimating subsystem costs for Model 3 uses the same framework as Model 2.

Details of subsystem costs are not provided here since they can be derived directly from definitions presented later for Model 5.
FIGURE 3.4.1
Model 4 Overall System and Random Variables

N
INPUT

Primary Data Entry
Primary Verification

X₁

X₂
Secondary Data Entry
Secondary Verification

primary
X₁

X₂ secondary

N

Comparison
no match
W

match
N-W

Manual Proofread
error detected
U

no error detected
(N-W)-U

Update

Find correction
no correction
Y

W-Y

Combination
X₅ initial

Valid value edit
error detected
V₃

no error detected
N-V₃

correction
F₃

Find correction
Update

V₃-F₃

Create analysis file
FINAL

Final error rate
Z
N

X₁ Primary entry errors
X₂ Secondary entry errors
W Comparison non-matches
U Manual Pf. detected errors
Y Comparison corrections
X₅ Initial processing errors
V₃ VVE detected errors
F₃ VVE corrections
Z Undetected errors
FIGURE 3.4.2
Model 4 Subsystems
3.4 Model 4 – VVE after Initial Processing

Model 4 introduces a valid value edit process that comes after corrections are made in other edit processes. In order to concentrate on the effect of doing valid value editing at a later stage, primary and secondary records are produced using only data entry, with possible dependent verification. As shown in Figure 3.4.1, the other edit processes are Comparison and Manual Proofread. The processing that precedes valid value editing in Model 4, called Initial Processing, is the same as Model 1. Three subsystems, shown in Figure 3.4.2, allow consideration of whether or not doing valid value editing after other edit checking is worthwhile.

3.4.1 Model 4 Error Generation

The theorems for Model 4 are derived from Models 1 – 3. No proofs are given since they are based directly on earlier theorems.

3.4.1.1 Initial Processing

The phases included in Initial Processing are Primary and Secondary Entry, Manual Proofread and Comparison. Initial Processing includes the same phases as Model 1. Let \( X_1 \) and \( X_2 \) be data entry error counts as defined in earlier models.

**Theorem 3.4.1.** Let \( W = \sum W_i \sim B(N, w_4) \), assuming \( W_i \sim B(1, w_4) \) are i.i.d. and \( W_i = \beta \) (no match for record \( i \) at Comparison). Assuming both Primary and Secondary Entry are performed, and all \( X_{1i} \) and \( X_{2i} \) are independent, then

\[
\begin{align*}
    w_4 &= w_1 = e_1 + e_2 - (1 + m_1)e_1e_2 \\
    m_1 &= P(W_i = 0 \mid X_{1i} = 1 \& X_{2i} = 1) = P(\text{match} \mid \text{both errors}).
\end{align*}
\]
Theorem 3.4.2. Let \( Y = \Sigma Y_i \sim B(N, y_4) \), assuming \( Y_i \sim B(1, y_4) \) are i.i.d. and \( Y_i = \beta(\text{error found for record } i \text{ after } \text{Comparison}) \).

When both Primary and Secondary Entry are performed, which implies Comparison is done, then, assuming all \( X_{1i} \) and \( X_{2i} \) are independent,

\[
y_4 = y_1 = c_1(e_1(1 - e_2) + (1 - m_1)e_1e_2).
\]

By definition,

\[
c_1 = P(Y_i = 1 \mid X_{1i} = 1 \& W_i = 1)
\]

and \( 1 - m_1 = P(W_i = 1 \mid X_{1i} = 1 \& X_{2i} = 1) \).

Theorem 3.4.3. Let \( U = \Sigma U_i \sim B(N, u_4) \), assuming \( U_i \sim B(1, u_4) \) are i.i.d. and \( U_i = \beta(\text{error detected for record } i \text{ at Manual Proofread}) \). As defined in Model 1, \( r_{110} \) and \( r_{111} \) are conditional detection probabilities. Depending on the subsystem, \( u_4 = u_{41} \) or \( u_4 = u_{42} \) as follows.

If only Primary processing is used, then \( u_4 = u_{41} \), where

\[
u_{41} = u_{11} = r_{110}(1 - e_1) + r_{111}e_1.
\]

If Manual Proofread is performed when both Primary and Secondary Entry are used, then a primary record is proofread only if the corresponding secondary record matches. Assuming all \( X_{1i} \) and \( X_{2i} \) are independent, then \( u_4 = u_{42} \), where

\[
u_{42} = u_{12} = r_{110}(1 - e_1)(1 - e_2) + r_{111}m_1e_1e_2.
\]
Theorem 3.4.4. Let \( X_5 = \sum X_{5i} \sim B(N, e_{45}) \), assuming \( X_{5i} \sim B(1, e_{45}) \) are i.i.d., \( X_{5i} = \beta(\text{error exists in primary record i after Initial Processing}) \). Depending on the subsystem, \( e_{45} = e_{451}, e_{45} = e_{452}, \) or \( e_{45} = e_{453} \) as follows.

If VVE is the only edit process used, then Initial Processing only includes Primary Entry, then

\[
e_{451} = e_1. \tag{3.4.1}
\]

When only Manual Proofread is used, then Initial Processing is the same as the Standard subsystem of Model 1, then

\[
e_{452} = z_{11} = r_{101}e_1 + (1 - k)[r_{110}(1 - e_1) + r_{111}e_1]. \tag{3.4.2}
\]

When Secondary Entry is done and Manual Proofread is not, then Initial Processing is the same as the Redundancy subsystem of Model 1, then

\[
e_{453} = z_{12} = e_1 - c_1k(e_1 - m_1e_1e_2). \tag{3.4.3}
\]

3.4.1.2 Valid Value Edit Process

The valid value edit process in Model 4 is similar to VVE in Models 2 and 3. The difference is that the incoming error probability is the result of Initial Processing instead of just an entry phase.

Theorem 3.4.5. Let \( V_3 = \sum V_{3i} \sim B(N, v_{43}) \), assuming \( V_{3i} \sim B(1, v_{43}) \) are i.i.d. and \( V_{3i} = \beta(\text{error detected in record i by Valid Value Edit}) \).

Let \( d_{4311} = P(V_{3i} = 1 | X_{5i} = 1) \).

When VVE is used, then \( v_{43} = d_{4311}e_{45} \). \tag{3.4.4}

Theorem 3.4.6. Let \( F_3 = \sum F_{3i} \sim B(N, f_{43}) \), assuming \( F_{3i} \sim B(1, f_{43}) \) are i.i.d. and \( F_{3i} = \beta(\text{error found in record i after error detected by Valid Value Edit}) \).

Assuming correct input data and appropriate edits, then \( f_{43} = v_{43} \).
3.4.1.3 Errors in Final Dataset

The errors in the final dataset are those that are not detected or not corrected properly by VVE. The number of errors that exist before VVE depends on which phases are included in Initial Processing.

**Theorem 3.4.7.** Let \( Z = \sum Z_i \sim B(N, z_4) \), assuming \( Z_i \sim B(1, z_4) \) are i.i.d., \( Z_i = \beta \) (error for record \( i \) in final data).

Let \( k = \text{Prob(proper correction|correction found)} \).

For all subsystems,

\[
z_4 = (1 - kd_{4311})e_{45}. \tag{3.4.5}
\]

If VVE is the only edit process used, then \( z_4 = z_{41} \):

\[
z_{41} = (1 - kd_{4311})e_1. \tag{3.4.6}
\]

When only Manual Proofread is used, then \( z_4 = z_{42} \):

\[
z_{42} = (1 - kd_{4311})e_{452}. \tag{3.4.7}
\]

When Secondary Entry is done and Manual Proofread is not included, then \( z_4 = z_{43} \):

\[
z_{43} = (1 - kd_{4311})e_{453}. \tag{3.4.8}
\]

3.4.2 Cost Estimation

As with Model 3, the details of cost estimation using Model 4 are not presented separately. Section 3.5.2 includes cost definitions that are easily modified to apply to the subsystems in Model 4.
3.5 Model 5 – Editing After Data Entry

Model 5 combines all the phases used in Models 1 – 4 into one system. Figure 3.5.1 shows the overall system except for the details of VVE phases. Each VVE phase includes finding corrections and updating records as in the earlier models.

The interesting subsystems in Model 5 are all the subsystems defined for Models 1 – 4. Only eight subsystems are considered for RDMS evaluation purposes. Other subsystems are possible, but would not be implemented in practice. In particular, Redundancy and Manual Proofread are not considered together regardless of the VVE phases included, and no subsystem with all three VVE phases is considered. The components of the interesting subsystems are shown in Figure 3.5.2. Subsystems that use redundancy (2,5,6,8) must include both Secondary Entry and Comparison. Since the goal of modeling is to evaluate different RDMS possibilities objectively, some seemingly less practical subsystems, such as Subsystem 4 (Primary VVE, Manual Proofread), are still included.

Model 5 allows a greater variety of subsystems to be compared because more phases are included. A simpler model is easier to use if that model describes all the RDM options available in a particular situation. A more complex model is appropriate to investigate situations with more flexibility. For instance, if VVE could be done by using intelligent data entry or a separate error detection program, then comparing Subsystems 6 and 8 would be useful. A more informed decision could then be made between using redundant intelligent data entry or just detecting valid value errors after correcting errors found by comparing paired records from redundant simple data entry.

3.5.1 Model 5 Error Generation

The theorems for Model 5 are derived from Models 1 – 4. No proofs are given since they are based directly on earlier theorems.
FIGURE 3.5.1
Model 5 Overall System and Random Variables

N
INPUT

Primary Data Entry
Primary Verification

X₁

secondary

VALID VALUE EDIT PROCESS

V₁

X₄

VALID VALUE EDIT PROCESS

X₂

Secondary Data Entry
Secondary Verification

N
Comparison

no match W

match N-W

Manual Proofread

error detected U

no error detected (N-W)-U

Update

Find correction correction Y

no correction

W-Y

X₁ Primary entry errors
X₂ Secondary entry errors
V₁, V₂ VVE detected errors
F₁, F₂ VVE corrections
X₃ Primary Proc. errors
X₄ Secondary Proc. errors
W Comparison non-matches
U Manual Pf. detected errors
Y Comparison corrections

X₅ Initial Processing errors
V₃ Final VVE detected errors
Z Undetected errors

Combination

initial X₅

Create analysis file

FINAL

N

Final error rate

Z
N
## FIGURE 3.5.2
Model 5 Subsystem Components for Eight Subsystems

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
<th>Phase 4</th>
<th>Phase 5</th>
<th>Phase 6</th>
<th>Phase 7</th>
<th>Phase 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Entry</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Primary VVE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secondary Entry</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Secondary VVE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Comparison</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Final VVE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Other Models</th>
<th>Phases *</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 – 4</td>
<td>No VVE, Manual Proofread only</td>
</tr>
<tr>
<td>2</td>
<td>1 – 4</td>
<td>No VVE, Redundancy only</td>
</tr>
<tr>
<td>3</td>
<td>2, 3</td>
<td>Primary VVE only</td>
</tr>
<tr>
<td>4</td>
<td>2, 3</td>
<td>Primary VVE, Manual Proofread</td>
</tr>
<tr>
<td>5</td>
<td>2, 3</td>
<td>Primary VVE, Redundancy</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>Primary VVE, Secondary VVE, Redundancy</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>Manual Proofread, Final VVE</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>Redundancy, Final VVE</td>
</tr>
</tbody>
</table>

* Redundancy implies Secondary Entry and Comparison phases
3.5.1.1 Data Entry

As in previous models, the results of the Primary and Secondary Entry phases are represented by two random variables, $X_1 \sim B(N, e_1)$ and $X_2 \sim B(N, e_2)$, that are the number of records with entry errors. All $X_{1i}$ and $X_{2i}$ are assumed independent.

3.5.1.2 Initial Valid Value Edit Processes

The Primary and Secondary VVE phases in Model 5 are the same as in Model 3. The counts of interest are: a) the number of records, $V_1$ or $V_2$, in which at least one field does not pass its valid value edit, and b) the number, $F_1$ or $F_2$, that need updating. The conditional detection probabilities are described by the parameters $d_{110}$ and $d_{111}$ ($d_{111} = P(V_{1i} = 1 \mid X_{1i} = 1)$) for Primary VVE and $d_{210}$ and $d_{211}$ for Secondary VVE.

**Theorem 3.5.1.** Let $V_1 = \sum V_{1i} \sim B(N, v_1)$, assuming $V_{1i} \sim B(1, v_1)$ are i.i.d. and $V_{1i} = \beta$(error detected in record $i$ by Valid Value Edit). Let $d_{111} = P(V_{1i} = 1 \mid X_{1i} = 1)$ and assume $d_{110} = 0$. When VVE is included in a subsystem, then

$$v_1 = d_{111} e_1.$$  

(3.5.1)

If Primary VVE is not done, then $d_{111} = 0$ and $V_1 = 0$ by definition.

**Theorem 3.5.2.** Let $F_1 = \sum F_{1i} \sim B(N, f_1)$, assuming $F_{1i} \sim B(1, f_1)$ are i.i.d. and $F_{1i} = \beta$(error found in record $i$ after error detected by Valid Value Edit). Assuming $d_{110} = 0$, correct input data and appropriate edits, then $f_1 = v_1$. 

Theorem 3.5.3. Let $V_2 = \sum V_{2i} \sim B(N, v_2)$, assuming $V_{2i} \sim B(1, v_2)$ are i.i.d. and $V_{1i} = \beta$ (error detected in record $i$ by Valid Value Edit). Let $d_{211} = P(V_{1i} = 1 | X_{1i} = 1)$ and assume $d_{210} = 0$. VVE is included in a subsystem, then

$$v_2 = d_{211}e_2.$$ (3.5.2)

If Secondary VVE is not done, then $d_{211} = 0$ and $V_2 = 0$ by definition.

Theorem 3.5.4. Let $F_2 = \sum F_{2i} \sim B(N, f_2)$, assuming $F_{2i} \sim B(1, f_2)$ are i.i.d. and $F_{2i} = \beta$ (error found in record $i$ after error detected by Valid Value Edit).

Assuming $d_{210} = 0$, correct input data and appropriate edits, then $f_2 = v_2$. 
3.5.1.3 Errors after Primary and Secondary Processing

As in Model 3, \( X_3 \) and \( X_4 \) are defined in Model 5 as the number of error records after Primary Processing and Secondary Processing, respectively.

**Theorem 3.5.5.** Let \( X_3 = \sum X_{3i} \sim B(N, e_3) \), assuming \( X_{3i} \sim B(1, e_3) \) are i.i.d., \( X_{3i} = \beta(\text{error exists in record } i \text{ after Primary Processing}) \). Assuming \( d_{110} = 0 \) and corrections are always found when an error is detected by Primary VVE, then

\[
e_3 = (1 - kd_{111}) e_1
\]  

(3.5.3)

where \( k = \text{Prob(proper correction } | \text{ detected & correction found).} \)

If Primary VVE is not done, then all \( X_{3i} = X_{1i} \) and \( X_3 = X_1 \), and \( e_3 = e_1 \).

**Theorem 3.5.6.** Let \( X_4 = \sum X_{4i} \sim B(N, e_4) \), assuming \( X_{4i} \sim B(1, e_4) \) are i.i.d., \( X_{4i} = \beta(\text{error exists in record } i \text{ after Secondary Processing}) \). Assuming \( d_{210} = 0 \) and corrections are always found when an error is detected by Secondary VVE, then

\[
e_4 = (1 - kd_{211}) e_2
\]  

(3.5.4)

where \( k = \text{Prob(proper correction } | \text{ detected & correction found).} \)

If Secondary Entry is not done, then all \( X_4 = 0 \) be definition.

If Secondary VVE is not done, then all \( X_{4i} = X_{2i} \) and \( X_4 = X_2 \), and \( e_4 = e_2 \).

**Lemma 3.5.1.** Assuming all \( X_{1i} \) and \( X_{2i} \) are independent, then all \( X_{3i} \) and \( X_{4i} \) are independent. If VVE is performed as part of Primary or Secondary Processing, assume that all \( X_{1i}, X_{2i}, V_{1i} \) and \( V_{2i} \) are mutually independent.
3.5.1.4 Initial Processing

After Primary and Secondary Processing the remaining phases included in Initial Processing are Comparison and Manual Proofread. There are several random variables that describe these phases: a) \( W \) is the number of non-matches, b) \( Y \) is the number of corrections found for non-matches, c) \( U \) is number of errors detected by Manual Proofread, and d) \( X_5 \) is the number of errors after all phases of Initial Processing.

Theorem 3.5.7. Let \( W = \sum W_i \sim B(N, w) \), assuming \( W_i \sim B(1, w) \) are i.i.d. and \( W_i = \beta \) (no match for record \( i \) at Comparison). Assuming both Primary and Secondary Processing are performed, and all \( X_{1i}, X_{2i}, V_{1i}, \) and \( V_{2i} \) are mutually independent, then

\[
\begin{align*}
  &w = e_3 + e_4 - (1 + m)e_3e_4 \\
\end{align*}
\]  (3.5.5)

where \( m = P(W_i = 0 | X_{3i} = 1 \& X_{4i} = 1) = P(\text{match} | \text{both errors}) \).

Theorem 3.5.8. Let \( Y = \sum Y_i \sim B(N, y) \), assuming \( Y_i \sim B(1, y) \) are i.i.d. and \( Y_i = \beta \) (error found for record \( i \) after Comparison).

When both Primary and Secondary Processing are performed (implies Comparison is done), and assuming all \( X_{1i}, X_{2i}, V_{1i}, \) and \( V_{2i} \) are mutually independent, then

\[
\begin{align*}
  &y = c[e_3(1 - e_4) + (1 - m)e_3e_4]. \\
\end{align*}
\]  (3.5.6)

By definition,

\[
\begin{align*}
  &c = P(Y_i = 1 | X_{1i} = 1 \& W_i = 1) \\
\end{align*}
\]

and \( 1 - m = P(W_i = 1 | X_{3i} = 1 \& X_{4i} = 1) \).
Theorem 3.5.9. Let $U = \sum U_i \sim B(N, u)$, assuming $U_i \sim B(1, u)$ are i.i.d. and $U_i = \beta$(error detected for record i at Manual Proofread). Let $r_{10}$ and $r_{11}$ be conditional detection probabilities ($r_{10} = P(U_i = 1 | X_{3i} = 1 \& W_i = 0)$). Depending on the subsystem, $u = u_1$ or $u = u_2$ as follows.

If only Primary processing is used, then $u = u_1$, where

$$u_1 = r_{10}(1 - e_3) + r_{11}e_3. \tag{3.5.7}$$

If Manual Proofread is performed when both Primary and Secondary Entry are used, then a primary record is proofread only if the corresponding secondary record matches. Assuming all $X_{1i}, X_{2i}, V_{1i}$, and $V_{2i}$ are mutually independent, then $u = u_2$, where

$$u_2 = r_{10}(1 - e_2)(1 - e_4) + r_{11}me_3e_4. \tag{3.5.8}$$

Theorem 3.5.10. Let $X_5 = \sum X_{5i} \sim B(N, e_5)$, assuming $X_{5i} \sim B(1, e_5)$ are i.i.d., $X_{5i} = \beta$(error exists in primary record i after Initial Processing). Depending on the subsystem, $e_5 = e_{51}, e_5 = e_{52}$, or $e_5 = e_{53}$ as follows.

If Initial Processing only includes Primary Processing, i.e., Manual Proofread and Secondary Processing are not done, then

$$e_{51} = e_3. \tag{3.5.9}$$

When Initial Processing includes Primary Processing followed by Manual Proofread (no Secondary Processing), then

$$e_{52} = r_{01}e_3 + (1 - k)(r_{10}(1 - e_2) + r_{11}e_3). \tag{3.5.10}$$

When Initial Processing includes Primary and Secondary Processing followed by Comparison only, no Manual Proofread is used, then

$$e_{53} = e_3 - ck(e_3 - me_3e_4). \tag{3.5.11}$$
3.5.1.5 Final Valid Value Edit Process

The final valid value edit process in Model 5 is similar to VVE in Model 4. The difference is that the incoming error probability is the result of Initial Processing, which includes Primary and Secondary VVE.

Theorem 3.5.11. Let \( V_3 = \sum V_{3i} \sim B(N, \nu_3) \), assuming \( V_{3i} \sim B(1, \nu_3) \) are i.i.d. and \( V_{3i} = \beta(\)error detected in record i by Valid Value Edit\( )\).

Let \( d_{311} = P(V_{3i} = 1 | X_{3i} = 1) \).

When Final VVE is used, then
\[
\nu_3 = d_{311}e_3. \tag{3.5.12}
\]

Depending on the subsystem, \( \nu_3 = \nu_{31}, \nu_3 = \nu_{32}, \) or \( \nu_3 = \nu_{33} \) as follows. If Final VVE is not included, then \( d_{311} = 0 \) and \( \nu_3 = 0 \) by definition.

If Initial Processing only includes Primary Processing, i.e., Manual Proofread and Secondary Processing are not done, then
\[
\nu_{31} = d_{311}e_3. \tag{3.5.13}
\]

When Initial Processing includes Primary Processing followed by Manual Proofread (no Secondary Processing), then
\[
\nu_{32} = d_{311}[r_{01}e_3 + (1 - k)(r_{10}(1 - e_3) + r_{11}e_3)]. \tag{3.5.14}
\]

When Initial Processing includes Primary and Secondary Processing followed by Comparison and Manual Proofread, then
\[
\nu_{33} = d_{311}[e_3 - ck(e_3 - me_4e_4)]. \tag{3.5.15}
\]

Theorem 3.5.12. Let \( F_3 = \sum F_{3i} \sim B(N, f_3) \), assuming \( F_{3i} \sim B(1, f_3) \) are i.i.d. and \( F_{3i} = \beta(\)error found in record i after error detected by Valid Value Edit\( )\).

Assuming correct input data and appropriate edits, then \( f_3 = \nu_3 \).
3.5.1.6 Errors in Final Dataset

For Model 5, the errors in the final dataset depend on which phases are included in Initial Processing and whether or not Final VVE is used. By using appropriate results from earlier phases, the binomial parameters for estimating final error rates for the eight interesting subsystems are easily stated from the following theorem.

**Theorem 3.5.13.** Let \( Z = \sum Z_i \sim B(N, z) \), assuming \( Z_i \sim B(1, z) \) are i.i.d., \( Z_i = \beta \) (error for record i in final data).

Let \( k = \text{Prob}(\text{proper correction} \mid \text{correction found}) \).

For all subsystems,

\[
Z = (1 - kd_{311})e_5.
\]

Depending on the subsystem, \( z = z_1, z = z_2, \) or \( z = z_3 \) as follows. If Final VVE is not included then \( d_{311} = 0 \). From Equations 3.5.3 and 3.5.4:

\[
e_3 = (1 - kd_{111})e_1,
\]

\[
e_4 = (1 - kd_{211})e_2.
\]

If Initial Processing only includes Primary Processing, i.e., Manual Proofread and Secondary Processing are not done, then

\[
z_1 = (1 - kd_{311})e_3.
\]

When Initial Processing includes Primary Processing followed by Manual Proofread (no Secondary Processing), then

\[
z_2 = (1 - kd_{311})[r_{01}e_3 + (1 - k)(r_{10}(1 - e_3) + r_{11}e_2)].
\]

When Initial Processing includes Primary and Secondary Processing followed by Comparison (no Manual Proofread), then

\[
z_3 = (1 - kd_{311})[e_3 - ck(e_3 - me_3 e_4)].
\]
3.5.1.7 Subsystem Summary

A summary of the binomial parameter results for the eight interesting subsystems of Model 5 is shown in Figure 3.5.3. The formulae are the results of using the appropriate theorem for each phase included in a subsystem. When a phase is not included, then the count random variable is zero by definition. The count parameters required for count estimation are summarized in Figure 3.5.4. Appendix A contains a summary of the theorems for Model 5. Evaluation of two alternative subsystems is based on the difference in the number of undetected errors, $Z$, and the difference in total cost, which depends on the record counts at the various processing phases. Chapters 4 and 5 provide discussion and examples of subsystem comparisons.
**FIGURE 3.5.3**

**Model 5 Subsystem Probability Parameters**

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_s$ $V_1$</td>
<td>0</td>
<td>0</td>
<td>$d_{111}e_1$</td>
<td>$d_{111}e_1$</td>
<td>$d_{111}e_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$V_s$ $V_2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$X_s$ $e_1$</td>
<td>$e_1$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$e_1$</td>
<td>$e_1$</td>
<td>0</td>
</tr>
<tr>
<td>$X_s$ $e_2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$W$ $w$</td>
<td>0</td>
<td>$e_1 + e_2 - (1+m)e_1e_2$</td>
<td>0</td>
<td>0</td>
<td>$e_1 + e_2 - (1+m)e_1e_2$</td>
<td>0</td>
<td>$e_1 + e_2 - (1+m)e_1e_2$</td>
<td>0</td>
</tr>
<tr>
<td>$Y$ $y$</td>
<td>0</td>
<td>$e_1[(1-e_1) + (l-m)e_1]$</td>
<td>0</td>
<td>0</td>
<td>$e_1[(1-e_1) + (l-m)e_1]$</td>
<td>0</td>
<td>$e_1[(1-e_1) + (l-m)e_1]$</td>
<td>0</td>
</tr>
<tr>
<td>$U$ $u$</td>
<td>$r_n(1-e_1) + s_1e_1$</td>
<td>0</td>
<td>0</td>
<td>$r_n(1-e_1) + s_1e_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$X_s$ $e_1$</td>
<td>$z$</td>
<td>$z$</td>
<td>$z$</td>
<td>$z$</td>
<td>$z$</td>
<td>$z$</td>
<td>$z$</td>
<td>0</td>
</tr>
<tr>
<td>$V_s$ $v_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$Z$ $z$</td>
<td>$e_1 - c_1(e_1 - m_n e_2)$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$e_1 - c_1(e_1 - m_n e_2)$</td>
<td>$e_1 - c_1(e_1 - m_n e_2)$</td>
<td>$e_1 - c_1(e_1 - m_n e_2)$</td>
<td>$(1-kd_{111})e_1$</td>
<td>$(1-kd_{111})e_1$</td>
<td>0</td>
</tr>
</tbody>
</table>

**Subsystems Phases**

1. No VVE, Manual Proofread only
2. No VVE, Redundancy only
3. Primary VVE only
4. Primary VVE, Manual Proofread
5. Primary VVE, Redundancy
6. Primary VVE, Secondary VVE, Redundancy
8. Redundancy, Final VVE

*Redundancy implies Secondary Entry and Comparison

**Count Random Variables**

- $V_s, V_1$: Errors detected by Primary or Secondary VVE
- $X_s, X_1$: Errors after Primary or Secondary Processing
- $W$: Non-matches at Comparison
- $Y$: Corrections found at Comparison
- $U$: Errors detected by Manual Proofread
- $X$: Errors after Initial Processing
- $V_s$: Errors detected by Final VVE
- $Z$: Final errors

103
FIGURE 3.5.4
Model 5 Count Parameters

Data Entry and Dependent Verification

\[ e_1 = \text{Prob(error after Primary Entry (and Verification if done))} = P(X_{11} = 1) \]
\[ e_2 = \text{Prob(error after Secondary Entry (and Verification if done))} = P(X_{21} = 1) \]

Primary Valid Value Edit

\[ d_{111} = \text{Prob(detected | error exists)} \]
\[ d_{101} = \text{Prob(not detected | error exists)} \]
\[ d_{111} = P(V_{11} = 1 | X_{11} = 1) \]
\[ d_{101} = P(V_{11} = 0 | X_{11} = 1) \]

Secondary Valid Value Edit

\[ d_{211} = \text{Prob(detected | error exists)} \]
\[ d_{201} = \text{Prob(not detected | error exists)} \]
\[ d_{211} = P(V_{21} = 1 | X_{21} = 1) \]
\[ d_{201} = P(V_{21} = 0 | X_{21} = 1) \]

Comparison

\[ m = \text{Prob(match | errors exist in both Primary and Secondary)} \]
\[ = P(\text{same bad value(s)} | X_{3i} = 1 \& X_{4i} = 1) \]

\[ c = \text{Prob(correction found | error exists & checked)} = P(Y_i = 1 | X_{3i} = 1 \& W_i = 1) \]

Manual Proofread

\[ r_{11} = \text{P(detected | error exists & proofread)} \]
\[ r_{01} = \text{P(not detected | error exists & proofread)} \]
\[ r_{10} = \text{P(detected | no error exists & proofread)} \]
\[ r_{00} = \text{P(not detected | no error exists & proofread)} \]
\[ r_{11} = P(U_i = 1 | X_{3i} = 1 \& W_i = 0) \]
\[ r_{01} = P(U_i = 0 | X_{3i} = 1 \& W_i = 0) \]
\[ r_{10} = P(U_i = 1 | X_{3i} = 0 \& W_i = 0) \]
\[ r_{00} = P(U_i = 0 | X_{3i} = 0 \& W_i = 0) \]

Final Valid Value Edit

\[ d_{311} = \text{Prob(detected | error exists)} \]
\[ d_{301} = \text{Prob(not detected | error exists)} \]
\[ d_{311} = P(V_{3i} = 1 | X_{3i} = 1) \]
\[ d_{301} = P(V_{3i} = 0 | X_{3i} = 1) \]

Correction

\[ k = \text{Prob(proper correction | error detected & correction found)} \]

Considered = 1 and independent of where the error was detected.
3.5.2 Cost Estimation

Cost estimation for Model 5 is similar to Model 2. The basic cost components are the same, although more indicator variables are required for the additional edit phases.

For Model 5, the cost parameters are defined as follows.

<table>
<thead>
<tr>
<th>System Cost Component</th>
<th>Development Cost</th>
<th>Processing Cost per Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Entry (Primary, Secondary)</td>
<td>$D_{51}$</td>
<td>$P_{51}$</td>
</tr>
<tr>
<td>(2) Dep. Verification (Primary, Secondary)</td>
<td>$D_{52}$</td>
<td>$P_{52}$</td>
</tr>
<tr>
<td>(3) Find correction (VVE, Comparison)</td>
<td>$D_{53}$</td>
<td>$P_{53}$</td>
</tr>
<tr>
<td>(4) Update (VVE, Comparison, Manual Pf.)</td>
<td>$D_{54}$</td>
<td>$P_{54}$</td>
</tr>
<tr>
<td>(5) Value Value Edit</td>
<td>$D_{55}$</td>
<td>$P_{55}$</td>
</tr>
<tr>
<td>(6) Comparison</td>
<td>$D_{56}$</td>
<td>$P_{56}$</td>
</tr>
<tr>
<td>(7) Manual Proofread</td>
<td>$D_{57}$</td>
<td>$P_{57}$</td>
</tr>
<tr>
<td>(8) Create complete file (combine, analysis)</td>
<td>$D_{58}$</td>
<td>$P_{58}$</td>
</tr>
</tbody>
</table>

Let $b_1 = \beta$(Primary Valid Value Edit process done),

$\quad b_2 = \beta$(Secondary Valid Value Edit process done),

$\quad b_3 = \beta$(Final Valid Value Edit process done).

Let $b = \beta$(any Valid Value Edit process done).

Let $s = \beta$(Redundancy used: Secondary Entry and Comparison done).

Let $t = \beta$(Manual Proofread process done).

Let $a_1 = \text{adjustment factor for intelligent data entry as Primary Entry},$

$\quad a_2 = \text{adjustment factor for intelligent data entry as Secondary Entry},$

$\quad a_3 = \text{adjustment factor for computer-assisted Final VVE}.$
FIGURE 3.5.5
Model 5 Overall System Cost Components

INPUT

(1) Primary Data Entry
(2) Primary Verification

N

Valid Value Edit Process
(3, 4, 5)

X_1

X_2

(1) Secondary Data Entry
(2) Secondary Verification

Valid Value Edit Process
(3, 4, 5)

V_1

V_2

primary
X_3

X_4 secondary

N

(6) Comparison

W

no match

N-W match

(7) Manual Proofread

U

error detected

no error detected

(N-W)-U

(3) Find correction

Y

correction

(4) Update

(4) Update

(8) Combination

initial
X_5

Valid Value Edit Process
(3, 4, 5)

V_3

(8) Create analysis file

FINAL

N

Cost Components
(1) Data entry
(2) Dependent verification
(3) Finding corrections
(4) Update
(5) Valid value edit program
(6) Comparison
(7) Manual Proofread
(8) Create complete file
Cost components for the overall system are in Figure 3.5.5. As before, costs for each component type are assumed to be the same regardless of which phase is involved.

Data entry costs depends on whether or not dependent verification is included in a subsystem.

\[
\begin{align*}
D_{5E} & = \begin{cases} 
D_{51} + D_{52} & \text{if dependent verification is performed,} \\
D_{51} & \text{if not.}
\end{cases} \\
\end{align*}
\]

\[
\begin{align*}
P_{51P} & = \begin{cases} 
P_{51} + P_{52} & \text{if dependent verification is performed,} \\
P_{51} & \text{if not.}
\end{cases} \\
\end{align*}
\]

Let \( P_{51S} \) be defined similarly to \( P_{51P} \).

In Model 5, let \( D_{5U} \) be the development cost for correcting and updating records. There is no cost, \( D_{5U} = 0 \), if all three VVE phases exclude batch processing. Otherwise, \( D_{5U} = D_{53} + D_{54} \).

\( \text{Definition} \ 3.5 \). Total cost is the sum of development and processing costs for all phases in a subsystem according to the structure shown in Figure 3.5.5.

Let \( T_5 = D_5 + P_5 \) where

\[
\begin{align*}
D_5 & = D_{5E} + D_{5U} + bD_{55} + sD_{56} + tD_{57} + D_{58}, \\
P_5 & = NP_{51P} + sNP_{51S} \\
& + b_1(NP_{55} + a_1V_1P_{53} + a_1F_1P_{54}) \\
& + sb_2(NP_{55} + a_2V_2P_{53} + a_2F_2P_{54}) \\
& + s(NP_{56} + WP_{53} + YP_{54}) \\
& + t((N - W)P_{57} + UP_{54}) \\
& + b_3(NP_{58} + NP_{55} + a_3V_3P_{53} + a_3F_3P_{54}) \\
& + NP_{58}. 
\end{align*}
\]

(3.5.20)
For the subsystems in Figure 3.5.2, the total costs are:

\[ T_{51} = D_{SE} + D_{SU} + D_{57} + D_{58} + NP_{51P} + (NP_{57} + UP_{54}) + NP_{58}, \]

\[ T_{52} = D_{SE} + D_{SU} + D_{56} + D_{58} + NP_{51P} + NP_{51S} + (NP_{56} + WP_{53} + YP_{54}) + NP_{58}, \]

\[ T_{53} = D_{SE} + D_{SU} + D_{55} + D_{58} + NP_{51P} + (NP_{55} + a_1V_{1}P_{53} + a_1F_{1}P_{54}) + NP_{58}, \]

\[ T_{54} = D_{SE} + D_{SU} + D_{55} + D_{57} + D_{58} + NP_{51P} + (NP_{55} + a_1V_{1}P_{53} + a_1F_{1}P_{54}) + (NP_{57} + UP_{54}) + NP_{58}, \]

\[ T_{55} = D_{SE} + D_{SU} + D_{55} + D_{56} + D_{58} + NP_{51P} + NP_{51S} + (NP_{55} + a_1V_{1}P_{53} + a_1F_{1}P_{54}) + (NP_{56} + WP_{53} + YP_{54}) + NP_{58}, \]

\[ T_{56} = D_{SE} + D_{SU} + D_{55} + D_{56} + D_{58} + NP_{51P} + NP_{51S} + (NP_{55} + a_1V_{1}P_{53} + a_1F_{1}P_{54}) + (NP_{56} + a_2V_{2}P_{53} + a_2F_{2}P_{54}) + (NP_{56} + WP_{53} + YP_{54}) + NP_{58}, \]

\[ T_{57} = D_{SE} + D_{SU} + D_{55} + D_{57} + D_{58} + NP_{51P} + (NP_{57} + UP_{54}) + NP_{58} + (NP_{55} + a_3V_{3}P_{53} + a_3F_{3}P_{54}) + NP_{58}, \]

\[ T_{58} = D_{SE} + D_{SU} + D_{55} + D_{56} + D_{58} + NP_{51P} + NP_{51S} + (NP_{56} + WP_{53} + YP_{54}) + NP_{58} + (NP_{55} + a_3V_{3}P_{53} + a_3F_{3}P_{54}) + NP_{58}. \]
FIGURE 3.6.1
Model 6 Overall System and Random Variables

N
INPUT

Primary Data Collection

T1

Only One

T2

Secondary Data Collection

Primary Data Entry
Primary Verification

X1

X2

Secondary Data Entry
Secondary Verification

primary X3

secondary X4

Comparison

no match W

match

N-W

Find correction

no correction

correction Y

Update

Find non-matches W-Y

Create analysis file

N

Final error rate

FINAL

Z

N
3.6 Model 6 – Redundancy in Data Collection

This model addresses the use of redundancy in data collection. The number of subsystems is reduced by always including Comparison as the only edit phase. No valid value editing or Manual Proofread is considered at this stage. The overall system, Figure 3.6.1, includes the restriction that Secondary Entry only uses one data collection source. The two subsystems of interest, shown in Figure 3.6.2, include 1) using complete redundancy in which both Primary and Secondary Processing have collection and entry phases, and 2) using partial redundancy in which Primary Collection results are used for both Primary and Secondary Entry.

3.6.1 Model 6 Error Generation Derivations

The derivations for Model 6 require defining more conditional parameters. Addressing the existence of errors after both data collection and entry is more complex since secondary records may be based on identical collection results, including errors, as primary records. Independence of primary and secondary records cannot always be assumed for the Comparison phase.

3.6.1.1 Data Collection

The number of records with errors made during data collection are represented by two random variables, $T_1$ and $T_2$. Let

$$T_1 = \sum T_{1i} \sim B(N, t_1) \text{ and } T_2 = \sum T_{2i} \sim B(N, t_2).$$

All $T_{1i}$ and $T_{2i}$ are assumed to be independent. Thus $t_1$ is the probability that a record counts as an error after Primary Collection.
3.6.1.2 Data Entry

The number of errors from data entry are represented by $X_1$ and $X_2$ as before. These are counts of errors as a result of data entry, not the number of errors that exist after data entry. Thus $X_1 = \sum X_{1i} \sim B(N, e_1)$ and $X_2 = \sum X_{2i} \sim B(N, e_2)$; all $X_{1i}$ and $X_{2i}$ are assumed independent. As in the other models, data entry may or may not include dependent verification; data entry error rates are higher without dependent verification.

3.6.1.3 Errors before Comparison

The number of records that count as errors before Comparison depends on the cumulative effect of both collection and entry. As shown in Figure 3.6.1 the error counts for primary and secondary records are represented by the random variables $X_3$ and $X_4$.

The possible outcomes for primary and secondary records can be defined using the random variables from data collection and entry. Figure 3.6.3 shows the outcomes for Primary Processing. The situation is similar for Secondary Processing. A record is correct only if no error is made in either collection or entry. If an error is made in one phase, but not the other, then the record counts as an error. When errors are made in both phases, it is not obvious how to count the result. Thus, let the parameters $h_{11}$, $h_{12}$ and $h_{22}$ represent the conditional probabilities of a compound error result.
FIGURE 3.6.3
Primary Processing Outcomes

<table>
<thead>
<tr>
<th>Primary Collection</th>
<th>Primary Entry</th>
<th>Correct $X_{1i} = 0$</th>
<th>Error $X_{1i} = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correct $T_{1i} = 0$</td>
<td>Correct</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error $T_{1i} = 1$</td>
<td>Error</td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>

$h_{11} = P(\text{error after Primary Collection and Entry} \mid T_{1i} = 1 \& X_{1i} = 1)$

$h_{22} = P(\text{error after Secondary Collection and Entry} \mid T_{2i} = 1 \& X_{2i} = 1)$

$h_{12} = P(\text{error after Primary Collection and Secondary Entry} \mid T_{1i} = 1 \& X_{2i} = 1)$

This model is designed mainly to consider situations in which data collection and entry are separate tasks. When considering random errors only, the existence of an error made during collection should not affect the chance of making an entry error since the person doing entry would not know that the information is incorrect. Thus, independence between data collection and data entry phases is assumed.

Theorem 3.6.1. Let $X_3 = \sum X_{3i} \sim B(N, e_{63})$, assuming $X_{3i} \sim B(1, e_{63})$ are i.i.d. and $X_{3i} = \beta(\text{error after Primary Processing})$. Assuming Primary Processing includes separate data collection and entry phases, i.e., that $T_{1i}$ and $X_{1i}$ are independent, then

$$e_{63} = t_i (1 - e_1) + (1 - t_i) e_1 + h_{11} t_1 e_1$$

$$= t_i + e_1 - (2 - h_{11}) t_1 e_1. \quad (3.6.1)$$
Proof:

\[ e_{63} = P(X_{3i} = 1) \]

\[ = P(\text{error exists after Primary Collection and Entry}) \]

\[ = P(\text{error from the four mutually exclusive possible outcomes}) \]

\[ = 0 + P(T_{1i} = 1 \& X_{1i} = 0) + P(T_{1i} = 0 \& X_{1i} = 1) \]

\[ + P(\text{error} \mid T_{1i} = 1 \& X_{1i} = 1) \]

\[ = P(T_{1i} = 1)P(X_{1i} = 0) + P(T_{1i} = 0)P(X_{1i} = 1) \]

\[ + P(\text{error} \mid T_{1i} = 1 \& X_{1i} = 1)P(T_{1i} = 1 \& X_{1i} = 1) \]

\[ = t_{1}(1 - e_{1}) + (1 - t_{1})e_{1} + h_{11}t_{1}e_{1} \]

\[ = t_{1} + e_{1} - (2 - h_{11})t_{1}e_{1}. \]

**Theorem 3.6.2.** Let \( X_{4i} = \sum X_{4i} \sim B(N, e_{64}) \), assuming \( X_{4i} \sim B(1, e_{64}) \) are i.i.d. and \( X_{4i} = \beta(\text{error after Secondary Processing}) \).

Assuming Secondary Processing includes separate data collection and entry phases, i.e., \( T_{2i} \) and \( X_{2i} \) are independent, then \( e_{64} = e_{641} \) where

\[ e_{641} = t_{2}(1 - e_{2}) + (1 - t_{2})e_{2} + h_{22}t_{2}e_{2}. \]  

(3.6.2)

If Secondary Processing does not include Secondary Collection, Primary Collection provides the input for Secondary Entry, i.e., \( T_{1i} \) and \( X_{2i} \) are independent, then \( e_{64} = e_{642} \) where

\[ e_{642} = t_{1}(1 - e_{2}) + (1 - t_{1})e_{2} + h_{12}t_{1}e_{2}. \]  

(3.6.3)

Proof:

In the first case, Secondary Processing is analogous to Primary Processing. Thus, substituting \( T_{2i} \) and \( X_{2i} \) in the proof of Theorem 3.6.1 gives the result.

In the second case, substitute \( X_{2i} \) for \( X_{1i} \) in the proof of Theorem 3.6.1. Since Primary Collection is used, \( T_{1i} \) remains appropriate.
3.6.1.4 Comparison

The description of error generation in the Comparison phase is more complex in Model 6 than in the previous models. When complete redundancy (separate Primary and Secondary Processing) is used, Comparison is similar to other models since independence between primary and secondary records is applicable. The subsystem in which Secondary Processing includes Primary Collection is more complicated.

The outcome possibilities when only Primary Collection is used are shown in Figure 3.6.4. We assume that all $T_{ii}, X_{1i}$ and $X_{2i}$ are mutually independent. Events 1.1 - 1.4 are the outcomes for primary and secondary records, $X_{3i}$ and $X_{4i}$, when no error is made in data collection, $T_{ii} = 0$. For these combinations of $T_{ii}, X_{1i}$ and $X_{2i}$, the probabilities do not depend on any conditional probability parameters since the values of $X_{3i}$ and $X_{4i}$ are easily defined. When a data collection error is made, $T_{ii} = 1$, there are more possible outcomes, Events 2.1 - 2.9, because of compound errors. For instance, when $T_{ii} = 1, X_{1i} = 1$ and $X_{2i} = 0$, there are two possibilities (Events 2.2 and 2.3) for the values of $X_{3i}$ and $X_{4i}$. Although the secondary record must be an error, $X_{4i} = 1$, the primary record may or may not be an error. The probabilities for these events use the conditional parameters defined earlier and $h_{111}$, which is the conditional probability that, for one record, the primary and secondary records are both compound errors given that errors were made in all preceding steps.
### FIGURE 3.6.4

**Table of Compound Error Outcomes**

<table>
<thead>
<tr>
<th>$T_{1i}$</th>
<th>$X_{1i}$</th>
<th>$X_{2i}$</th>
<th>$x_{3i}$</th>
<th>$x_{4i}$</th>
<th>Event #</th>
<th>$P(X_{3i} = x_{3i} &amp; X_{4i} = x_{4i})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.1</td>
<td>$(1 - t_1)(1 - e_1)(1 - e_2)$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1.2</td>
<td>$(1 - t_1)e_1(1 - e_2)$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1.3</td>
<td>$(1 - t_1)(1 - e_1)e_2$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1.4</td>
<td>$(1 - t_1)e_1e_2$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2.1</td>
<td>$t_1(1 - e_1)(1 - e_2)$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2.2</td>
<td>$t_1(1 - h_{11})e_1(1 - e_2)$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2.3</td>
<td>$t_1h_{11}e_1(1 - e_2)$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2.4</td>
<td>$t_1(1 - e_1)(1 - h_{12})e_2$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2.5</td>
<td>$t_1(1 - e_1)h_{12}e_2$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2.6</td>
<td>$h_{100}t_1e_1e_2$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2.7</td>
<td>$(1 - h_{11} - h_{100})t_1e_1e_2$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2.8</td>
<td>$(h_{11} - h_{111})t_1e_1e_2$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2.9</td>
<td>$h_{111}t_1e_1e_2$</td>
</tr>
</tbody>
</table>
Lemma 3.6.1. When Primary and Secondary Processing include separate data collection and entry phases, assuming all $T_{1i}, T_{2i}, X_{1i}$ and $X_{2i}$ are mutually independent, then all $X_{3i}$ and $X_{4i}$ are independent.

Lemma 3.6.2. For the subsystem that uses Primary Collection results as the input for both Primary Entry and Secondary Entry, then

$$e_{611} = P(X_{3i} = 1 \& X_{4i} = 1)$$

$$= (1 - t_1)e_1e_2 + t_1[(1 - e_1)(1 - e_2) + h_{11}e_1(1 - e_2) + h_{12}(1 - e_1)e_2 + h_{111}e_1e_2].$$

By definition

$$e_{610} = P(X_{3i} = 1 \& X_{4i} = 0)$$

$$= e_{63} - e_{611}. \quad (3.6.5)$$

Proof of Lemma 3.6.2:

$$e_{611} = P(X_{3i} = 1 \& X_{4i} = 1)$$

$$= P(\text{Event 1.4}) + P(\text{Event 2.1}) + P(\text{Event 2.3}) + P(\text{Event 2.5})$$

$$+ P(\text{Event 2.9}) \quad \text{from Figure 3.6.4},$$

$$= (1 - t_1)e_1e_2 + t_1(1 - e_1)(1 - e_2) + h_{11}t_1e_1(1 - e_2) + h_{12}t_1(1 - e_1)e_2$$

$$+ h_{111}t_1e_1e_2$$

$$= (1 - t_1)e_1e_2$$

$$+ t_1[(1 - e_1)(1 - e_2) + h_{11}e_1(1 - e_2) + h_{12}(1 - e_1)e_2 + h_{111}e_1e_2].$$
Theorem 3.6.3. Let \( W = \sum W_i - B(N, w_6) \), assuming \( W_i - B(1, w_6) \) are i.i.d. and \( W_i = \beta \) (no match for record i at Comparison).

Let \( m_6 = P(W_i = 0 \mid X_{3i} = 1 \& X_{4i} = 1) \).

Assuming Primary Processing and Secondary Processing both include separate data collection and entry phases, \( T_{1i}, X_{1i}, T_{2i}, \) and \( X_{2i} \) are independent, then

\[
w_6 = w_{61},
\]

\[
w_{61} = e_{63} + e_{64} - (1 + m_6)e_{63}e_{64}.
\]

(3.6.6)

When only Primary Collection is included, then \( w_6 = w_{62} \).

\[
w_{62} = 1 - [(1 - t_1)(1 - e_1)(1 - e_2) + m_6 e_{61}].
\]

(3.6.7)

Proof:

In the first case, substituting appropriately in the proof of Theorem 2.1 and using Lemma 3.6.1 yields the result.

When Secondary Entry uses the results of Primary Collection the proof of Theorem 2.1 cannot be used because \( X_{3i} \) and \( X_{4i} \) are clearly not independent.

However, we assume that all \( T_{1i}, X_{1i} \) and \( X_{2i} \) are mutually independent.

\[
w_{62} = P(W_i = 1)
\]

\[
= P(\text{different values})
\]

\[
= 1 - P(\text{both correct or both same error})
\]

\[
= 1 - [P(X_{3i} = 0 \& X_{4i} = 0) + P(X_{3i} = 1 \& X_{4i} = 1 \& \text{match})]
\]

\[
= 1 - [P(T_{1i} = 0 \& X_{1i} = 0 \& X_{2i} = 0) + P(X_{3i} = 1 \& X_{4i} = 1 \& \text{match})]
\]

\[
= 1 - [P(T_{1i} = 0)P(X_{1i} = 0)P(X_{2i} = 0) + P(X_{3i} = 1 \& X_{4i} = 1 \& \text{match})]
\]

\[
= 1 - [(1 - t_1)(1 - e_1)(1 - e_2) + P(X_{3i} = 1 \& X_{4i} = 1 \& \text{match})]
\]

\[
= 1 - [(1 - t_1)(1 - e_1)(1 - e_2) + m_6 e_{61}]
\]

by Lemma 3.6.2.
Theorem 3.6.4. Let \( Y = \sum Y_i - B(N, y_6) \), assuming \( Y_i - B(1, y_6) \) are i.i.d. and \( Y_i = \beta \) (error found for record \( i \) after Comparison).

Let \( c_6 = P(Y_i = 1 | X_{3i} = 1 \& W_i = 1) \).

By an earlier definition, \( 1 - m_6 = P(W_i = 1 | X_{3i} = 1 \& X_{4i} = 1) \).

Assuming Primary Processing and Secondary Processing include separate data collection and entry phases, then \( y_6 = y_{61} \),

\[
y_{61} = c_6(e_{63} - m_6e_{63}e_{641}).
\]

(3.6.8)

When Primary Collection is common to Primary and Secondary Processing then \( y_6 = y_{62} \),

\[
y_{62} = c_6(e_{63} - m_6e_{611}).
\]

(3.6.9)

Proof:

Following the proof of Theorem 2.3:

\[
y_6 = P(Y_i = 1)
\]

\[= P(Y_i = 1 | X_{3i} = 1 \& W_i = 1)P(X_{3i} = 1 \& W_i = 1)
\]

\[= c_6P(W_i = 1 \& X_{3i} = 1)
\]

\[= c_6P(W_i = 1 \& X_{3i} = 1 \& (X_{4i} = 0 \text{ or } X_{4i} = 1))
\]

\[= c_6[P(W_i = 1 \& X_{3i} = 1 \& X_{4i} = 0) + P(W_i = 1 \& X_{3i} = 1 \& X_{4i} = 1)]
\]

\[= c_6[P(W_i = 1 \& X_{3i} = 1 \& X_{4i} = 0)P(X_{3i} = 1 \& X_{4i} = 0)
\]

\[+ P(W_i = 1 \& X_{3i} = 1 \& X_{4i} = 1)P(X_{3i} = 1 \& X_{4i} = 1)]
\]

\[= c_6[1 \times P(X_{3i} = 1 \& X_{4i} = 0) + (1 - m_6)P(X_{3i} = 1 \& X_{4i} = 1)].
\]

In the first case, \( X_{3i} \) and \( X_{4i} \) are independent by Lemma 3.6.1, thus

\[
y_{61} = c_6[P(X_{3i} = 1)P(X_{4i} = 0) + (1 - m_6)P(X_{3i} = 1)P(X_{4i} = 1)]
\]

\[= c_6(e_{63} - m_6e_{63}e_{641}).
\]

In the second case, Lemma 3.6.2 applies, so

\[
y_{62} = c_6[(e_{63} - e_{611}) + (1 - m_6)e_{611}]
\]

using Equations 3.6.4, 3.6.5.

\[= c_6(e_{63} - m_6e_{611}).
\]
3.2.1.8 Errors in Final Dataset

The errors that remain in the final dataset are those undetected by Comparison or the result of improper updating. The number of undetected errors depends on whether or not Secondary Collection is used.

There are three mutually exclusive events that lead to undetected errors. The parameter \( k \) is the probability that an update results in a proper correction. The possibilities are:

(i) an error is undetected by Comparison;
(ii) an error is undetected by Comparison Correction;
(iii) an update due to Comparison introduces an error.

*Theorem 3.6.5.* Let \( Z = \sum Z_i \sim B(N, z_\theta) \), assume \( Z_i \sim B(1, z_\theta) \) are i.i.d. and \( Z_i = \beta \text{(error for record } i \text{ in final data)} \).

Let \( k = \text{Prob(proper correction | correction found)} \).

Assuming Primary Processing and Secondary Processing include separate data collection and entry phases, then \( z_6 = z_{61} \),

\[
z_{61} = e_{63} - c_\theta k(e_{63} - m_6 e_{63} e_{641}). \tag{3.6.10}
\]

When Primary Collection is common to Primary and Secondary Processing then \( z_6 = z_{62} \),

\[
z_{62} = e_{63} - c_\theta k(e_{63} - m_6 e_{611}). \tag{3.6.11}
\]
Proof:

\[ z_6 = P(Z_1 = 1) \]

\[ = P(\text{undetected error by Comparison}) \]

\[ + P(\text{undetected error by Comparison correction}) \]

\[ + P(\text{improper update after Comparison correction}) \]

\[ = P(\text{match & error}) + P(\text{not found & checked}) + P(\text{imp update}) \]

\[ = P(\text{match } X_{3i} = 1 \& X_{4i} = 1)P(X_{3i} = 1 \& X_{4i} = 1) \]

\[ + (1 - c_6)p(W_i = 1 \& X_{3i} = 1) + (1 - k)P(Y_i = 1) \]

\[ = m_6 P(X_{3i} = 1 \& X_{4i} = 1) + (1 - c_6)P(W_i = 1 \& X_{3i} = 1) \]

\[ + (1 - k)c_6 P(W_i = 1 \& X_{3i} = 1) \]

\[ = m_6 P(X_{3i} = 1 \& X_{4i} = 1) + (1 - kc_6) P(W_i = 1 \& X_{3i} = 1) \]

In the first case, \( X_{3i} \) and \( X_{4i} \) are independent by Lemma 3.6.1, so

\[ z_{61} = m_6 e_{63} e_{641} + (1 - kc_6)(e_{63} - m_6 e_{63} e_{641}) \quad \text{See proof of Theorem 3.6.4.} \]

\[ = e_{63} - c_6 k(e_{63} - m_6 e_{63} e_{641}) \]

In the second case, using Lemma 3.6.2, then

\[ z_{62} = m_6 e_{611} + (1 - kc_6)(e_{63} - m_6 e_{611}) \quad \text{See proof of Theorem 3.6.4.} \]

\[ = e_{63} - c_6 k(e_{63} - m_6 e_{611}) \]
3.6.2 Cost Estimation

The cost components for Model 6 are based on Model 5 with the addition of data collection. The same numbering is used when a component appears in earlier models.

<table>
<thead>
<tr>
<th>System Cost Component</th>
<th>Development Cost</th>
<th>Processing Cost per Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0) Data collection (Primary, Secondary)</td>
<td>$D_{60}$</td>
<td>$P_{60}$</td>
</tr>
<tr>
<td>(1) Entry (Primary, Secondary)</td>
<td>$D_{61}$</td>
<td>$P_{61}$</td>
</tr>
<tr>
<td>(2) Dep. Verification (Primary, Secondary)</td>
<td>$D_{62}$</td>
<td>$P_{62}$</td>
</tr>
<tr>
<td>(3) Find correction (VVE, Comparison)</td>
<td>$D_{63}$</td>
<td>$P_{63}$</td>
</tr>
<tr>
<td>(4) Update (VVE, Comparison, Manual Pf.)</td>
<td>$D_{64}$</td>
<td>$P_{64}$</td>
</tr>
<tr>
<td>(6) Comparison</td>
<td>$D_{66}$</td>
<td>$P_{66}$</td>
</tr>
<tr>
<td>(8) Create complete file (combine, analysis)</td>
<td>$D_{68}$</td>
<td>$P_{68}$</td>
</tr>
</tbody>
</table>

Let $s_6 = \beta$(Secondary Collection, and Secondary Entry, done).

The overall system with cost components is shown in Figure 3.6.5. Costs for each component type are assumed to be the same regardless of which phase is involved.

Data entry costs depend on whether or not dependent verification is included in a subsystem.

Let $D_{6E} = \begin{cases} 
D_{61} + D_{62} & \text{if dependent verification is performed}, \\
D_{61} & \text{if not.}
\end{cases}$

Let $P_{61P} = \begin{cases} 
P_{61} + P_{62} & \text{if dependent verification is performed}, \\
P_{61} & \text{if not.}
\end{cases}$

Let $P_{61S}$ be defined similarly to $P_{61P}$.
FIGURE 3.6.5
Model 6 Overall System Cost Components

Cost Components
(0) Data collection
(1) Data entry
(2) Dependent verification
(3) Finding corrections
(4) Update
(6) Comparison
(8) Create complete file

(8) Create analysis file
FINAL
Definition 3.6. Total cost is the sum of development and processing costs for all phases in a subsystem according to the structure shown in Figure 3.6.5.

\[ T_6 = D_6 + P_6 \text{ where} \]
\[ D_6 = D_{60} + D_{6E} + D_{63} + D_{64} + D_{66} + D_{68}, \quad (3.6.12) \]
\[ P_6 = NP_{60} + s_6 NP_{60} + NP_{61P} + NP_{61S} + NP_{66} + WP_{63} + YP_{64} + NP_{68}. \quad (3.6.13) \]

The difference in total cost between the subsystems shown in Figure 3.6.2 depends only on processing costs. Development cost are the same since each cost component appears at least once in both subsystems.

With redundant data collection, the estimated processing cost is
\[ E(P_6) = 2NP_{60} + NP_{61P} + NP_{61S} + NP_{66} + E(W)P_{63} + E(Y)P_{64} + NP_{68} \]
\[ = 2NP_{60} + NP_{61P} + NP_{61S} + NP_{66} + (Nw_{61})P_{63} + (Ny_{61})P_{64} + NP_{68} \]
using Equations 3.6.6 and 3.6.8.

With only Primary Collection, then
\[ E(P_6) = NP_{60} + NP_{61P} + NP_{61S} + NP_{66} + E(W)P_{63} + E(Y)P_{64} + NP_{68} \]
\[ = NP_{60} + NP_{61P} + NP_{61S} + NP_{66} + (Nw_{62})P_{63} + (Ny_{62})P_{64} + NP_{68} \]
using Equations 3.6.7 and 3.6.9.
CHAPTER 4
INVESTIGATION OF EXTENDED MODELS

This chapter presents insights into RDM provided by this modeling approach. We will concentrate on Model 5, the most general of the data entry models, with some discussion of Model 6, the data collection model. The other data entry models, Models 1 – 4, are not discussed directly since they address situations that are also described by Model 5.

4.1 System Evaluation

The models are useful tools for deciding between alternative RDM systems. A discussion is presented of parameter values, for both count and cost estimation, that are required when using a model for system evaluation. The error generation process and relationships between count parameters are explored using Model 5 once parameter value ranges are established. Problems posed by cost estimation are discussed.

Methods for using the models for system evaluation are demonstrated by several specific comparisons of subsystem pairs. In certain cases, conclusions can be drawn the circumstances in which one subsystem is better than another. Important relationships between differences in final error rates and total subsystem cost are explored in cost comparison examples for well defined RDM environments. The objective is to provide examples of decision making based on model predictions.
4.2 Reducing Model 5

Model 5 reduces to simpler models for particular situations since it is a general model of RDM systems that begin with data entry. As an example, we will show how Model 5 describes the situation addressed by Model 2, which has Primary VVE, Manual Proofread, and Comparison as editing phases. The models are equivalent when comparable parameters in the count and cost estimation formulae have the same values. Appendix A contains information about parameter correspondences for the other models.

The difference between the two models is that Model 5 includes more processing steps. As shown in Figure 4.1, the models have the same structure when Secondary and Final VVE are not used in Model 5. Both models begin with data entry and include the same editing phase possibilities. In practice, Model 2 is easier to use for this particular situation since fewer parameters are involved. However, investigating Model 5 provides information about a greater variety of situations and results can be applied to the simpler models.

4.2.1 Count Estimation

The correspondences between the count parameters are easily shown. In this situation, some parameters are zero in Model 5 since VVE is only included in Primary Processing \((d_{211} = 0, d_{311} = 0\), and so \(V_2 = 0, V_3 = 0\)). The data entry error rates, \(e_1\) and \(e_2\), are the same for both models since no processing before data entry is included in either model. The other parameters correspond as follows:

<table>
<thead>
<tr>
<th>Count parameter (rate)</th>
<th>Model 2</th>
<th>Model 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary VVE detection</td>
<td>(d_{11})</td>
<td>(d_{111})</td>
</tr>
<tr>
<td>Match at Comparison</td>
<td>(m_2)</td>
<td>(m)</td>
</tr>
<tr>
<td>Comparison correction</td>
<td>(e_2)</td>
<td>(c)</td>
</tr>
<tr>
<td>Manual Proofread detection</td>
<td>(r_{211}, r_{210}, r_{201})</td>
<td>(r_{11}, r_{10}, r_{01})</td>
</tr>
</tbody>
</table>
To consider count estimates for the two models, note that several subsystems are possible even with this simple situation. There are three subsystems of interest when Primary VVE is included: 1) Primary VVE only, 2) Primary VVE with Manual Proofread, and 3) Primary VVE with Redundancy. The definitions of the count random variables are the same for the phases included in both models. \( V_1 \) is the count of error records detected by Primary VVE, \( W \) is the count for Comparison, \( U \) is the count for Manual Proofread, and \( Z \) is the number of errors in the final dataset.

The models are equivalent when the count estimates are the same. Figure 4.2 shows that the probability parameters would be the same when the count parameter values are the same \( (d_{11} = d_{111}, m_2 = m, c_2 = c) \). For example, \( v_{21} = v_1 = v \) holds when \( d_{11} = d_{111} \) since \( v_{21} = d_{11} e_1 \) and \( v_1 = d_{111} e_1 \). Thus, the count estimate, \( E(V_1) = Nv \), is the same using either model.

### 4.2.2 Cost Estimation

For the same situation, if count estimates are the same then cost estimates also will be the same using either model. The cost parameters have the same values since the phases are identical, i.e., the development and processing costs are the same for all phases \( (D_{2j} = D_{5j} \text{ and } P_{2j} = P_{5j} \text{ for all } j) \). Model 5 cost is simplified by eliminating Secondary and Final VVE \( (b_2 = 0 \text{ and } b_3 = 0) \). The total cost formulae in Sections 3.3.2 and 3.5.2, for Model 2 and Model 5 respectively, are equivalent under these conditions.
**FIGURE 4.2**
Comparison of Model 2 and Model 5 Probability Parameters

<table>
<thead>
<tr>
<th></th>
<th>Model 2</th>
<th>Model 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1) Primary VVE only</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE</td>
<td>$v_{21} = d_{11}e_1$</td>
<td>$v_1 = d_{111}e_1$</td>
</tr>
<tr>
<td>Final error</td>
<td>$z_2 = e_{23} = (1 - kd_{11})e_1$</td>
<td>$z = e_3 = (1 - kd_{111})e_1$</td>
</tr>
<tr>
<td><strong>2) Primary VVE and Manual Proofread</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE</td>
<td>$v_{21} = d_{11}e_1$</td>
<td>$v_1 = d_{111}e_1$</td>
</tr>
<tr>
<td>Primary Processing</td>
<td>$e_{23} = (1 - kd_{11})e_1$</td>
<td>$e_3 = (1 - kd_{111})e_1$</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>$u_2 = r_{210}(1 - e_{23}) + r_{211}e_{23}$</td>
<td>$u = r_{10}(1 - e_3) + r_{11}e_3$</td>
</tr>
<tr>
<td>Final error</td>
<td>$z_2 = r_{201}e_{23} + (1 - k)u_2$</td>
<td>$z = r_{01}e_3 + (1 - k)u$</td>
</tr>
<tr>
<td><strong>3) Primary VVE and Comparison</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE</td>
<td>$v_{21} = d_{11}e_1$</td>
<td>$v_1 = d_{111}e_1$</td>
</tr>
<tr>
<td>Primary Processing</td>
<td>$e_{23} = (1 - kd_{11})e_1$</td>
<td>$e_3 = (1 - kd_{111})e_1$</td>
</tr>
<tr>
<td>Secondary Processing</td>
<td>$e_2$</td>
<td>$e_4 = e_2$</td>
</tr>
<tr>
<td>Comparison</td>
<td>$w_2 = e_{23} + e_2$</td>
<td>$w = e_3 + e_2$</td>
</tr>
<tr>
<td></td>
<td>$- (1 + m_2)e_{23}e_2$</td>
<td>$- (1 + m)e_3e_2$</td>
</tr>
<tr>
<td>Comparison correction</td>
<td>$y_2 = c_2[e_{23}(1 - e_2)$</td>
<td>$y = c[e_3(1 - e_2)$</td>
</tr>
<tr>
<td></td>
<td>$+ (1 - m_2)e_{23}e_2]$</td>
<td>$+ (1 - m)e_3e_2]$</td>
</tr>
<tr>
<td>Final error</td>
<td>$z_2 = e_{23} - c_2k(e_{23} - m_2e_{23}e_2)$</td>
<td>$z = e_3 - ck(e_3 - me_3e_2)$</td>
</tr>
</tbody>
</table>
4.3 Parameter Values

Since estimates from the models depend on many parameters, we will discuss values for them before presenting applications of this modeling approach. There are two groups of model parameters, those that are used in count estimation, which affect processing costs, and those that only appear in cost estimation. Some classes of parameters (data entry error rates, VVE detection rates) may not require different values for each processing step. The objective is to establish a range of values for each parameter that applies to the target situations of interest.

The values presented are drawn from the literature and several studies conducted in 1985 and 1986. Most of the studies were done at the UNC Highway Safety Research Center (HSRC), a small research organization that does in-house data entry for some studies. Data entry is done using a simple microcomputer database program by people who are not hired specifically for data entry. Three studies were handled as experiments to obtain additional parameter values; these are called the BAC, Nevada, and Bike studies. More information about HSRC and details of the experimental studies are in Chapter 5. A separate study, called Glaucoma, provided information about error rates for more difficult data and VVE detection during data entry. Appendix B contains a summary of all studies that provided information about parameter values.

4.3.1 Count Parameters

The count estimation parameters in the models are error rates or, in edit checking phases, conditional rates. Error rates are required for data collection and entry phases. Conditional detection rates appear in the VVE, Manual Proofread, and Comparison phases. A conditional proper correction rate is involved for all edit phases. The Comparison phase has another conditional parameter in order to consider the possibility of matched errors.
4.3.1.1 Parameter Simplifications

Whenever redundant processing steps are involved, using the same value for similar parameters is a reasonable simplification. In Model 6, the data collection error rates, \( t_1 \) and \( t_2 \), for Primary and Secondary Data Collection can be considered equal, \( t_1 = t_2 = t \). For all the models, the error rates for Primary and Secondary Entry can be considered equal, \( e_1 = e_2 = e \), provided that both methods either include or exclude dependent verification. VVE checking can be considered equally effective in detecting errors regardless of which processing step is involved. This would imply \( d_{311} = d_{3211} \) in Model 3, and \( d_{111} = d_{211} = d_{311} \) in Model 5.

The assumption was made in all models that the update process was essentially the same for all edit phases. Thus, the conditional probability of a proper correction given that an update is done, \( k \), has the same value regardless of which phase is involved.

4.3.1.2 Error Rates

Error rates for data collection are difficult to obtain. Most studies of data processing errors deal with those that are detected after data collection is complete. However, the idea that collection error rates are not the same as, and are probably greater than, detected error rates is supported by studies that use redundant data collection to consider data quality. There may be multiple sources of error related to data collection that can produce undetectable errors.

Data entry error rates vary depending on the input data, but a reasonable range can be identified for the target situation. The results from the experimental studies are presented together with pertinent reports from the literature.

Although data entry methods have changed, record error rates are relatively stable. In 1941 and 1942, Deming reported card (45-column) error rates of one-half to five percent in U.S. Census processing. This is in a similar range as the six percent card (80-column) error rate reported by Nemanich in 1972. The input data in all these studies
were on simple forms with no skip fields. For more complicated records, error rates can be quite high, as evidenced by the twenty-five percent record error rate for 1040 income tax forms reported by Schwartz (1982).

Professional data entry services are expected to have low error rates. A stroke error rate of 1/5000, which is generally accepted, could mean a card (80-column) error rate of 1.6%. With survey data, error rates tend to be higher unless a form is optimized for data entry instead of data collection. Dependent verification is often used by professional services to reduce error rates. A study done at HSRC, called NC Belt, had a detected record error rate of one-half percent after professional data entry without dependent verification or intelligent data entry. There were about 20,000 records with an average of ten short (mostly single character) fields in which switch errors were likely to occur. The actual data entry error rate could have been as high as one percent.

Data entry error rates for studies in which non-professionals enter data using standard database software on microcomputers are higher than for professional services, but are not necessarily very much higher. The error rates for in-house entry at HSRC ranged from two percent to fifteen percent for a variety of input data; most entry error rates were between five and ten percent. When very simple data was involved, the BAC study had only three numeric fields, the record error rate was about two percent in three separate trials (about 600 records each). In two small studies (about 300 records), in which one line of a form provided input for one record, the error rates were five and six percent. Another study used a well designed form that included some long character fields (name, address) and five other short fields; the forms were divided into four groups of 150 to 350 forms. The error rates for Primary and Secondary Entry ranged from five to fifteen percent. The Nevada study had about 1000 survey forms with an average of fifteen categorical answers on each form. Primary and Secondary Entry error rates were eight and nine percent. The Glaucoma study (350 records) had very messy
forms and some coding decisions had to be made during data entry. The error rates in three trials were between fifteen and twenty percent.

The appropriate range for data entry error rates clearly depends on whether professional services or non-professionals are involved. For professional services, a range of 0.1 – 5.0% will cover most situations that satisfy the prototype criteria. With non-professionals using microcomputers, 1 – 10% is a good range for relatively neat and straightforward input data, while 5 – 20% would be reasonable for more complex data.

4.3.1.3 Conditional Rates

Values for the conditional parameters are only available from the research done for this paper. A conditional detection rate is defined as the probability that an error is detected by a certain edit check given that an error exists. All the detection rates vary with different types of input data, but similar studies do have similar rates. The conditional parameter in the update step of all edit phases is the conditional probability that a record is corrected properly given that an update is done.

Two studies provide VVE detection rates. In the Glaucoma study, 163 records had errors of which 97 were detected by VVE checks, for a detection rate of sixty percent. For Primary and Secondary VVE of the Nevada study the detection rates were thirty and thirty-seven percent, based on about 80 error records. The Glaucoma data included numeric fields for which range checks were possible, which made detecting switch errors more likely than with the Nevada survey data.

Detection rates for Manual Proofread most likely decrease as the number of fields and/or records increase. The input data for the Bike study were on a computer listing so proofreading was very simple. The detection rate was about ninety percent (15 errors existed in 250 records). In the BAC study, undergraduate students working in pairs proofread 600 records (3 fields). They only detected four of eleven errors for a detection rate of thirty-six percent.
The empirical evidence supports the assumption that the probability of detecting an error (finding a correction), when corresponding records do not match at Comparison, is near one. In the Nevada study, checking primary records resulted in a detection rate of ninety-nine percent for 76 error records. Checking secondary records by a different method yielded a detection rate of ninety-eight percent (82 errors existed).

The conditional probability for proper corrections is also near one. In the BAC and Bike studies, the proper correction rates were about ninety percent based on 15 updates in each study. In the Nevada study, the correction rates were ninety-seven percent for both the primary and secondary records, which had 75 and 80 updates, respectively.

4.3.1.4 Matched Errors at Comparison

The Comparison phase includes a conditional parameter related to the possibility of matched errors. The parameter, m, is the conditional probability that a match occurs when paired error records exist, i.e., the records match when both corresponding primary and secondary records are errors. Since we are using record oriented models, this means that both records would have at least one error in at least one field, but could have errors in more than one field. In order for a matched error to occur, the error(s) must be in the same field(s) and have exactly the same incorrect value(s).

The value of m is probably small in most situations, but could be near one in certain situations. Suppose a record consisted of a single field with two possible values. If VVE were done during data entry, then any error that existed after Primary Processing must be a switch error. If paired error records occurred they must match; thus, m = 1 in this case. In contrast, consider a situation in which there were 50 fields in a record and an equal probability of an error occurring in any field. This could apply to a survey in which all answers are on a three-point scale. If the paired error records each had an error in only one field, each error must be in the same field before a matched error can occur.
The probability that the error would occur in the same field is clearly small for this situation and the value of $m$ is even smaller since a matched error also requires the same incorrect value.

In any case, the unconditional probability of a matched error is always quite small, certainly less than one percent. Suppose $m = 1$ and the primary and secondary entry error rates are ten percent, $e_1 = e_2 = 0.10$. Then

$$P(\text{match} \& \text{primary error} \& \text{secondary error})$$

$$= P(\text{match} \mid X_{1i} = 1 \& X_{2i} = 1)P(X_{1i} = 1 \& X_{2i} = 1)$$

$$= m \times e_1 \times e_2 = 0.01.$$

Experimental study results support the idea that the value of $m$ is small. In the Nevada study, there were eight paired error records. Of these, only one was a matched error for $m = 1 / 8 = 0.13$. If Primary and Secondary Processing had both included VVE then only three pairs of error records would have occurred. The matched error would still have been undetected so $m = 1 / 3 = 0.33$ for that situation.

4.3.2 Cost Parameters

The cost parameter values presented in this section demonstrate that, for similar situations, reasonable estimates for development and processing costs can be identified for planning purposes. Even if exact estimates are unavailable, we can make assumptions about relative differences between parameters. Values for all cost parameters are only required if a total cost estimate for a subsystem is of interest. Only certain cost parameters are necessary for subsystem comparisons.

Actual cost values are a combination of labor costs and computer costs. The labor rates used for the experimental studies are $5 per hour for undergraduate “work-study” students and $20 per hour for programmers or research assistants. These labor rates apply roughly to the environment at HSRC. Computer costs usually depend on whether a mainframe computer or a microcomputer is used. In the target situation, using a
mainframe computer commonly entails some cost for computer time while costs associated with the use of existing microcomputers are considered as part of overhead cost. Microcomputer costs would occur if equipment is purchased or rented to implement a subsystem. Therefore, costs for phases that require computers always include labor cost and may include computer cost.

The cost values presented below are from studies done in 1985 or 1986. Whenever microcomputers were used, only labor costs were associated with their use. Overhead costs for existing microcomputers were relatively small compared to labor costs. While specific costs are constantly changing, these examples provide a basis for investigating the models.

4.3.2.1 Development Costs

Development costs are fixed costs that include the cost of developing, testing, and implementing a processing step. Manual processes, such as Manual Proofread, only involve labor costs for time spent in planning and training. When mainframe computer charges exists, then development cost for a phase such as Comparison includes both labor and computer cost.

A general range for data collection development cost cannot be reasonably defined. The amount of time required to develop input forms, data collection protocols, and training varies too much even for the target situation. Obviously, developing a protocol for a mail-back survey is much less costly than developing a survey for use by trained telephone interviewers. However, given a specific study, estimating development cost is usually possible for, especially when previous experience exists for the RDM environment.

At HSRC, setting up a simple data entry process using a microcomputer database program is not a difficult task. The total development cost was $40 for the Glaucoma
study. Since the Bike study had simpler data values, the total development cost was only $10, which included providing VVE during entry.

Dependent verification is done only as part of professional data entry in a target study. Thus, in our models there is essentially no development cost.

Not much development is required for manual proofreading or updating records interactively. Using a computer program to update from a separate update file is a different situation. The Glaucoma, BAC and Bike studies had development costs of $5 – 10 for time spent planning and/or training.

Development costs are relatively higher for phases that require computer programs. Skilled labor, someone with programming ability, is required and computer costs may exist. In the Glaucoma study, mainframe programs were used. The development costs for the VVE and comparison programs were about $50 each, including labor and computer costs for testing. In the BAC study, a very simple comparison program that only compared complete records was developed using a mainframe at a cost of $20. The cost of creating a complete working file was about $10 in these two studies. New software for microcomputers is making it possible to lower development costs by eliminating mainframe computer charges and decreasing the time required to create and test programs. Using appropriate microcomputer software, the development cost at HSRC for Comparison would range from $10 to $40, batch VVE would cost $20 – 60, and creating a working file would cost $5 – 20.

4.3.2.2 Processing Costs

Total processing costs are based on an average cost for processing one record and the number of records processed. As with development, the relative importance of processing cost for a phase depends on whether manual processing, mainframe computer processing, or microcomputer processing are involved.
Data collection processing costs increase with the amount of manual labor involved. The data collection step involves relatively small handling costs for a study based on a survey form that is completed by a respondent. When input data are created by coding from existing hardcopy information, the processing cost depends on the time required to code one record. The most expensive data collection method uses interviewers or observers to collect information. In most cases, processing costs can be estimated reasonably once a data collection method is chosen.

Although data entry processing costs depend on the complexity of the input data, similar studies provide useful values for planning purposes. The BAC, Bike, and Nevada data had very similar entry processing costs, about $0.05 per record. For more complex data, processing costs are naturally higher. The Glaucoma data had average processing costs of $0.10 - 0.15. Another HSRC study with long character fields (name, address) cost $0.20 - 0.30 per record for data entry.

The NC Belt study, mentioned earlier, provides examples of professional entry service costs. An input form contained twenty records (lines) with an average of ten short fields per record. At one service, without intelligent data entry, the charge was $0.05 for one record. Dependent verification was one-third the cost of entry, $0.02 per record. Another service used intelligent data entry and charged $0.02 for entry or dependent verification of one record.

The cost of finding corrections depends a great deal on the effort required to access the original input data. The average processing cost was $0.10 and $0.11 for the Glaucoma and Bike studies in which original input data was readily available. In the Nevada study the cost was $0.10 - 0.20 when survey forms requiring investigation (about 80 of 950) were gathered together before looking for corrections. The cost was much higher, about $1 each, when a survey form was retrieved, checked, and replaced before another record was considered.
Update processing costs were much lower at HSRC when microcomputers were used. Processing costs for the experimental studies in which records were updated interactively on a mainframe computer ranged from $0.20 to $0.90. For the Nevada study, corrections found by VVE were updated on a mainframe at a cost of $0.26 each, while the average cost was $0.03 – 0.06 for updates done on a microcomputer for error records found as a result of Comparison.

The cost of manual proofreading varies inversely with the detection rate. In the BAC study (3 fields), the processing cost for proofreading by students was $0.02 per record, compared to $0.05 for entry. The Bike study had slightly more complex data (31 fields) and was proofread by a research assistant. Proofreading was more carefully done in this case; processing cost was at least $0.05 per record, which was about the same as the cost of entry.

Computer processing is generally much less costly than manual processing. The average processing cost for a mainframe batch VVE program was about $0.005 in the Glaucoma and Nevada studies. The Comparison program used was more expensive. The range was from $0.016 to $0.040 per record for eight program runs from several studies; most of the runs averaged $0.02 per record processed. When a microcomputer was used for VVE or Comparison at HSRC, then the processing cost was considered zero.
4.4 Error Generation

Exploring the model predictions of final error rates provides insights into the error generation process and relationships between different edit methods. The final error rate is defined as the undetected error rate. In all models the number of undetected errors is represented by the random variable \( Z; Z \sim \beta(N, z) \) in Model 5 and the final error rate is \( z = E(Z) / N \). We will not consider cost differences while investigating error generation. We will show which error sources significantly affect the final error rate and which parameter values can be set to fixed values.

There are of two types error sources; either new errors are created or existing errors remain undetected. Data entry and improper corrections are sources of new errors in all the models. The various edit checking phases are error sources because of undetected errors. Data collection is another source of new errors in Model 6. The importance of an error source is evaluated by considering the associated parameters. Acceptable ranges for the parameter values are established based on Section 4.2. The importance of each parameter is considered relative to other parameters and the effect on final error rates.

A comparison of the final error rates for the eight subsystems of Model 5 is presented. Conclusions about the potential effectiveness of the different edit methods are presented.

4.4.1 Parameter Values

For the prototype situation, establishing useful parameter value ranges is possible as discussed in Section 4.2. The values used for this exploration, shown in Figure 4.3, have very conservative ranges since value ranges are usually much tighter for actual situations. Within these ranges, the question is whether or not a change in the value of a particular parameter will significantly change the final error rate, i.e., whether the final error rate is sensitive to changes in the parameter value. The goal is to divide the
parameters into two categories, depending on whether the final error rate is relatively sensitive or insensitive to the parameter's value.

FIGURE 4.3
Count Parameter Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value Range</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1, t_2$</td>
<td>0.01 - 0.20</td>
<td>Data collection error rates</td>
</tr>
<tr>
<td>$e_1, e_2$</td>
<td>0.01 - 0.10</td>
<td>Data entry error rates</td>
</tr>
<tr>
<td>$h$</td>
<td>0 - 1.00</td>
<td>Compound error rates: $h_{111}, h_{222}, h_{12}, h_{111}$</td>
</tr>
<tr>
<td>$d$</td>
<td>0.10 - 0.90</td>
<td>VVE detection rates: $d_{111}, d_{211}, d_{311}$</td>
</tr>
<tr>
<td>$r_{11}$</td>
<td>0.25 - 0.75</td>
<td>Manual Proofread detection rate</td>
</tr>
<tr>
<td>$r_{10}$</td>
<td>0 - 0.05</td>
<td>Manual Proofread detection rate for non-existing errors</td>
</tr>
<tr>
<td>$m$</td>
<td>0 - 1.00</td>
<td>Matched errors probability</td>
</tr>
<tr>
<td>$c$</td>
<td>0.90 - 1.00</td>
<td>Comparison detection rate</td>
</tr>
<tr>
<td>$k$</td>
<td>0.90 - 1.00</td>
<td>Proper correction rate</td>
</tr>
</tbody>
</table>

The final error rate, $z$, in Model 5 is relatively insensitive to four parameters. The Manual Proofread detection rate for non-existing errors, $r_{10}$, is considered very small. Within the defined range for $r_{10}$, differences in $z$ are negligible for the appropriate subsystems. Therefore, for practical purposes we can simplify the final error rate formulae by using $r_{10} = 0$. The conditional probability of matched paired errors, $m$, does not have a restricted range, but it also does not play an important role in relation to $z$. In the subsystem formulae for $z$, the parameter $m$ always occurs in a term that is of the form $m \times e_1 \times e_j$, $j = 1, 2$. Since data entry error rates are less than or equal to 0.10, a change in the value of $m$ cannot change $z$ by more than 0.01. The maximum effect on $z$ when $e_1 \leq 0.05$ and $e_2 \leq 0.05$ is 0.0025, which is not significant compared to effects of changing other parameters.
The Comparison detection rate, c, and the proper correction rate, k, have limited ranges and are further restricted in their possible effect on the final error rate by their functional relationship to the data entry error rate. For all the subsystems of Model 5, the formulae for z can be expressed in the following form:

\[ z = e_1 - e_1 \times k \times f(\text{prob}) \]

The function f(prob) represents a product of the other count parameters, which are all probabilities less than or equal to one. The data entry error rate is restricted, \( e_1 \leq 0.10 \), and the maximum change in k is 0.10 (1.00 - 0.90). Thus, the maximum change in z, due to a change in k, is less than one percent. As the data entry error rate decreases, the effect of k on z decreases proportionally. A similar functional relationship holds between c and z. Another way to consider the importance of c is to consider Subsystem 2, Redundancy only. In Subsystem 2, by Theorem 3.5.13,

\[ z = e_1 - cke_1 + ckme_1e_2. \]

Suppose the data entry error rates were ten percent, \( e_1 = e_2 = 0.10 \). Then c must increase by 10 - 11% in order to decrease the final error rate by one percent. Since \( 0.90 \geq c \geq 1.00 \), it is obvious that no improvement greater than one percent is possible.

The remaining parameters in Model 5 all have significant and interrelated effects on final error rates. Clearly, final error rates decrease as error rates for sources of new errors decrease and/or detection rates increase. One aspect must be held fixed in order to explore the magnitude of the effect on final error rates of changing the other. Subsystem 1, Manual Proofread only, and Subsystem 3, Primary VVE only, provide examples of the relative importance of these parameters. Suppose we are interested in obtaining a one percent decrease in the final error rate. Using Manual Proofread, the detection rate, \( r_{11} \), must increase by 10 - 20% for data entry error rates of 0.05 - 0.10.

To achieve the same reduction when \( r_{11} = 0.50 \), the entry error rate must decrease by two percent. Similar effects apply to using Primary VVE. In practice, however, raising
the effectiveness of VVE is much more difficult than raising the effectiveness of proofreading. Reducing entry error rates appears to be the most important way to reduce final error rates for these subsystems. A small decrease in entry error rate is generally easier to achieve than a relatively large increase in a detection rate.

In the data collection model, Model 6, the important parameters are the collection and entry error rates. The compound error rates \((h_{11}, h_{22}, h_{12}, h_{111})\) have relatively little effect on the final error rate for reasons similar to those discussed for \(m\) in Model 5. The Comparison detection rate, \(c_6\), and the proper correction rate, \(k\), are also relatively insignificant; these parameters correspond to \(c\) and \(k\) in Model 5.

4.4.2 Model 5 Final Error Rates

The eight subsystems of Model 5 provide an easy way to compare the effectiveness of the different edit methods. Since final error rates are relatively insensitive to some parameters, we can concentrate on a smaller number of likely situations. Figure 4.4 presents the final error rates for six different situations that are defined by allowing data entry to be poor, average, or good \((e_1 = e_2 = 0.10, 0.05, 0.01)\), and the VVE detection rates to be poor or average \((d_{111} = d_{211} = d_{311} = 0.30, 0.70)\). The expected number of final errors, with standard deviations, are shown for \(N = 1000\). We can compare the results to discover which subsystem is more effective.

When only considering final error rates, there is some duplication within the eight subsystems. If VVE detection rates are considered identical for all VVE phases, then Subsystem 5 and Subsystem 8 have identical final error rates. Subsystem 4 and Subsystem 7 have very similar formulae for final error rates. The two are identical if VVE detection rates are the same \((d_{111} = d_{311})\) and Manual Proofread does not include the possibility of detecting non-existent errors \((r_{10} = 0)\). Subsystems 7 and 8 are of interest because they model different situations from Subsystems 4 and 5. The difference is between using Primary VVE or Final VVE; Primary VVE can be accomplished
by using intelligent data entry while Final VVE is usually done using a batch program. Cost difference would be the deciding factor in choosing a subsystem for these two pairs of subsystems.

The most effective edit method is based on redundant data entry. Figure 4.5 is a graph of the final error rates shown in Figure 4.4. When only one edit method is used, simple Redundancy (Subsystem 2) is better than either Manual Proofread or VVE (Subsystems 1 and 3). An interesting characteristic of Subsystem 2 is that the final error rate does not change much when the data entry rate decreases from 0.10 to 0.05. Thus, the predicted effectiveness of Redundancy is relatively insensitive to data entry error rates. All of the best subsystems are based on using Redundancy together with VVE (Subsystems 5, 6 and 8). Even with a high entry error rate of ten percent, subsystems that include Redundancy have final error rates of one percent or less.

The final error rate difference between subsystems is more pronounced for higher data entry error rates. As expected, when few errors are made during entry, e = 0.01, all subsystems produce "clean" data. Even so, using Redundancy results in the lowest final error rates, at least twice as low as non-Redundancy subsystems.

One purpose of the model is to show that acceptable levels of data quality can be achieved using easily implemented edit methods. The most effective subsystem, Subsystem 6, can have a final error rate less than 0.5% in many prototype situations and is based on edit methods that are increasingly available. The most appropriate RDM system based on Subsystem 6 would use Redundancy with intelligent data entry for both Primary and Secondary Entry. Both intelligent data entry and comparison of paired records can be accomplished using relatively inexpensive, widely available, micro-computer programs without highly trained computer programmers.
**FIGURE 4.4**

Model 5 Final Error Rates

| N  = 1000 | Number of records |
| c = 0.95 | Conditional detection rate at Comparison |
| m = 0.50 | Conditional probability of matched errors |
| r_{11} = 0.50 | Conditional detection rate for Manual Proofread given error exists |
| r_{10} = 0 | Conditional detection rate for Manual Proofread given no error |
| k = 0.95 | Conditional probability of proper correction |

Data entry rates: \( e_1 = e_2 = e \)

<table>
<thead>
<tr>
<th>Subsystem phases</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Proofread only</td>
<td>0.0053</td>
<td>5.3</td>
<td>2.3</td>
<td>0.0053</td>
<td>5.3</td>
<td>2.3</td>
</tr>
<tr>
<td>Redundancy only</td>
<td>0.0010</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0010</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Primary VVE only</td>
<td>0.0072</td>
<td>7.2</td>
<td>2.7</td>
<td>0.0034</td>
<td>3.4</td>
<td>1.8</td>
</tr>
<tr>
<td>Primary VVE, Manual Proofread</td>
<td>0.0038</td>
<td>3.8</td>
<td>1.9</td>
<td>0.0018</td>
<td>1.8</td>
<td>1.3</td>
</tr>
<tr>
<td>Primary VVE, Redundancy</td>
<td>0.0007</td>
<td>0.7</td>
<td>0.8</td>
<td>0.0003</td>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>Prim. &amp; Sec. VVE, Redundancy</td>
<td>0.0007</td>
<td>0.7</td>
<td>0.8</td>
<td>0.0003</td>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>Manual Proofread, Final VVE</td>
<td>0.0038</td>
<td>3.8</td>
<td>1.9</td>
<td>0.0018</td>
<td>1.8</td>
<td>1.3</td>
</tr>
<tr>
<td>Redundancy, Final VVE</td>
<td>0.0007</td>
<td>0.7</td>
<td>0.8</td>
<td>0.0003</td>
<td>0.3</td>
<td>0.5</td>
</tr>
</tbody>
</table>

VVE detection rates: \( d_{111} = d_{211} = d_{311} = d \)

Data entry rates: \( e_1 = e_2 = e \)

<table>
<thead>
<tr>
<th>Subsystem phases</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Proofread only</td>
<td>0.026</td>
<td>26.3</td>
<td>5.1</td>
<td>0.026</td>
<td>26.3</td>
<td>5.1</td>
</tr>
<tr>
<td>Redundancy only</td>
<td>0.006</td>
<td>6.0</td>
<td>2.4</td>
<td>0.006</td>
<td>6.0</td>
<td>2.4</td>
</tr>
<tr>
<td>Primary VVE only</td>
<td>0.036</td>
<td>35.8</td>
<td>5.9</td>
<td>0.017</td>
<td>16.8</td>
<td>4.1</td>
</tr>
<tr>
<td>Primary VVE, Manual Proofread</td>
<td>0.019</td>
<td>18.8</td>
<td>4.3</td>
<td>0.009</td>
<td>8.8</td>
<td>3.0</td>
</tr>
<tr>
<td>Primary VVE, Redundancy</td>
<td>0.004</td>
<td>4.3</td>
<td>2.1</td>
<td>0.002</td>
<td>2.0</td>
<td>1.4</td>
</tr>
<tr>
<td>Prim. &amp; Sec. VVE, Redundancy</td>
<td>0.004</td>
<td>4.1</td>
<td>2.0</td>
<td>0.002</td>
<td>1.8</td>
<td>1.3</td>
</tr>
<tr>
<td>Manual Proofread, Final VVE</td>
<td>0.019</td>
<td>18.8</td>
<td>4.3</td>
<td>0.009</td>
<td>8.8</td>
<td>3.0</td>
</tr>
<tr>
<td>Redundancy, Final VVE</td>
<td>0.004</td>
<td>4.3</td>
<td>2.1</td>
<td>0.002</td>
<td>2.0</td>
<td>1.4</td>
</tr>
</tbody>
</table>

Data entry rates: \( e_1 = e_2 = e \)

<table>
<thead>
<tr>
<th>Subsystem phases</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
<th>( z )</th>
<th>E(Z)</th>
<th>s.d.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Proofread only</td>
<td>0.053</td>
<td>52.5</td>
<td>7.1</td>
<td>0.053</td>
<td>52.5</td>
<td>7.1</td>
</tr>
<tr>
<td>Redundancy only</td>
<td>0.014</td>
<td>14.3</td>
<td>3.8</td>
<td>0.014</td>
<td>14.3</td>
<td>3.8</td>
</tr>
<tr>
<td>Primary VVE only</td>
<td>0.072</td>
<td>71.5</td>
<td>8.1</td>
<td>0.034</td>
<td>33.5</td>
<td>5.7</td>
</tr>
<tr>
<td>Primary VVE, Manual Proofread</td>
<td>0.038</td>
<td>37.5</td>
<td>6.0</td>
<td>0.018</td>
<td>17.6</td>
<td>4.2</td>
</tr>
<tr>
<td>Primary VVE, Redundancy</td>
<td>0.010</td>
<td>10.2</td>
<td>3.2</td>
<td>0.005</td>
<td>4.8</td>
<td>2.2</td>
</tr>
<tr>
<td>Prim. &amp; Sec. VVE, Redundancy</td>
<td>0.009</td>
<td>9.3</td>
<td>3.0</td>
<td>0.004</td>
<td>3.8</td>
<td>1.9</td>
</tr>
<tr>
<td>Manual Proofread, Final VVE</td>
<td>0.038</td>
<td>37.5</td>
<td>6.0</td>
<td>0.018</td>
<td>17.6</td>
<td>4.2</td>
</tr>
<tr>
<td>Redundancy, Final VVE</td>
<td>0.010</td>
<td>10.2</td>
<td>3.2</td>
<td>0.005</td>
<td>4.8</td>
<td>2.2</td>
</tr>
</tbody>
</table>
FIGURE 4.5
Comparison of Model 5 Final Error Rates for Eight Subsystems

Data entry error rate

VVE detection rate

d = 0.30

d = 0.70

Model 5 Subsystems

1 Manual Proofread only
2 Redundancy only
3 Primary VVE only
4 Primary VVE, Manual Pf.
5 Primary VVE, Redundancy
6 Prim. & Sec. VVE, Redundancy
7 Manual Pf., Final VVE
8 Redundancy, Final VVE
4.5 Cost Estimation

The models include cost estimation because cost is an important factor for most RDM decisions, but general results are more difficult to obtain than with error generation. Cost estimates, especially for total cost, are very sensitive to parameter values. Limited ranges for cost parameter values are only possible when considering a particular RDM environment. For different RDM environments, cost values vary by much larger orders of magnitude than error or detection rates. However, when the RDM environment is well known then model cost predictions are useful in two areas: 1) providing objective information about the relative cost of two alternative subsystems, and 2) project budgeting for a chosen RDM system.

A model’s ability to predict overall cost is acceptable as long as we can show that overall cost estimates are reasonable when parameter values are well defined. Overall system cost is composed of many components; many parameters, both cost and count, are involved. In a given RDM environment, a model should provide believable cost estimates for future projects when parameter values are available from previous experience. In addition, the definition of overall cost provides a basis for deriving expressions for cost differences between subsystems.

Considering the cost difference between a specific pair of subsystems is simpler than estimating overall cost. The expression for a cost difference always includes fewer parameters since any two subsystems have some system components in common. Suppose we were comparing two subsystems that used Primary Entry followed by different edit methods. Development and processing costs of the entry phase would not be an issue in deciding which subsystem would be less expensive. By providing expressions for subsystem cost differences, the models help focus attention on important parameters. In most situations, the model can provide a valid cost difference prediction, or at least the order of magnitude of the difference.
4.6 Subsystem Comparisons

We will now present examples of conclusions drawn from using a model to compare two subsystems. Concentrating on differences between subsystems leads to discovery of critical parameter values. We are particularly interested in judging the cost effectiveness of simple redundancy in comparison to other edit possibilities. The objective is to discover plausible general recommendations.

There are three main factors in deciding between two RDM systems: 1) final error rate, 2) total cost, and 3) total time required to implement the RDMS and complete data processing. The ideal system would have the fewest undetected errors, cost the least and take the shortest amount of time. In most situations, we must compromise and decide which factor is more important. The models provide a method for obtaining quantitative information about the first two factors, final error rate and cost.

General conclusions about error generation and cost/benefit analyses are discussed in the specific subsystem comparisons. Five subsystem pairs are presented from Model 5. The two subsystems of Model 6, the data collection model, are also considered. By investigating derived expressions for the final error rate difference between two subsystems, we find that some parameters have critical values. If a parameter has a critical value then we can identify when one subsystem is more effective in reducing the number of undetected errors. General statements about cost differences are difficult to make because cost estimates are very sensitive to changes in cost parameter values. However, useful cost comparisons for representative RDM situations are presented.
FIGURE 4.6

Model 5, Subsystem 1: Manual Proofread only

Model 5, Subsystem 2: Redundancy only
4.6.1 Redundancy versus Manual Proofread

For the first subsystem comparison, let us consider Subsystems 1 and 2 of Model 5. As shown in Figure 4.6, Subsystem 1 has Manual Proofread as the only edit phase while Subsystem 2 has redundant data entry and Comparison. These are the same subsystems described by Model 1. Considering the difference in predicted final error rates leads to an interesting conclusion about the Manual Proofread detection rate. The RDM environment at HSRC is used as the basis for judging cost effectiveness.

Let $\text{DIFF}_{21}(z)$ be defined as the difference between the undetected error probability parameters for the two subsystems. As shown in Figure 3.5.3, the formulae for Subsystems 1 and 2 are:

$$z_{51} = r_{01} e_1 + (1 - k)(r_{10}(1 - e_1) + r_{11} e_1)$$  \hspace{1cm} \text{Manual Proofread only,}

$$z_{52} = e_1 - c(k_1 - m_1 e_2)$$  \hspace{1cm} \text{Redundancy only.}

The entry error rates are represented by $e_1$ and $e_2$. Suppose the RDM environment does not allow dependent verification, and let $e_1 = e_2 = e$. The estimate for Subsystem 1 includes conditional detection rates for Manual Proofread. Suppose $r_{10}$, the probability that an error is detected by Manual Proofread when one does not exist, is negligible: $r_{10} = 0$. As defined in Theorem 3.5.9, the other conditional parameter is $r_{11}$, which represents the conditional detection rate of finding an existing error at Manual Proofread ($r_{11} = P(\text{detected} | \text{exists } \& \text{ proofread})$; $r_{01} = (1 - r_{11})$ by definition.

Thus,

$$\text{DIFF}_{21}(z) = z_{52} - z_{51} = [e_1 - c(k_1 - m_1 e_2)] - [r_{01} e_1 + (1 - k)(r_{10}(1 - e_1) + r_{11} e_1)]$$

$$= [e - c(k - m e)] - [(1 - r_{11})e + (1 - k)(r_{10}(1 - e) + r_{11} e)]$$

$$= c k m e^2 + (k r_{11} - c k)e.$$
In order for Subsystem 1, Manual Proofread, to have fewer final errors than Redundancy, \( \text{DIFF}_{21}(z) > 0 \) must hold:

\[
ckme^2 + (kr_{11} - ck)e > 0
\]

implies

\[
ckme + kr_{11} - ck > 0
\]

and

\[
\frac{r_{11}}{} > c - cme.
\]

A minimum value of \( r_{11} \) can be determined from this inequality, given values of the Comparison detection rate, \( c \), and the data entry error rate, \( e \). If \( c \geq 0.90 \) and \( e \leq 0.10 \), then \( r_{11} \) must be greater than 0.81 in order for a Manual Proofread subsystem to have a lower final error rate than Redundancy. The effectiveness of proofreading must increase as the Comparison detection rate increases for Subsystem 1 to remain better. Tighter ranges on the parameter values can usually be defined in a specific RDM situation. For instance, \( c \geq 0.95 \) and \( e = 0.05 \) are appropriate values for many studies at HSRC; in this case the critical value for \( r_{11} \) is 0.90.

Therefore, we can conclude from the model that using Subsystem 2, the Redundancy subsystem, will result in fewer undetected errors unless Manual Proofread is done very carefully. Using Redundancy was always better in the experimental studies since the value of \( r_{11} \) was always less than sixty percent. Of course, for a specific situation the cost difference is usually also considered in choosing between two subsystems.

Let \( \text{DIFF}_{21}(T) = E(T_{52}) - E(T_{51}) \) be the predicted difference in total cost between the Redundancy and Manual Proofread subsystems. Based on Definition 3.5,

\[
\text{DIFF}_{21}(T) = (D_{56} - D_{57}) + N(P_{515} - P_{57}) + NP_{56} + E(W)P_{53} + E(Y - U)P_{54}
\]

Consider the magnitude of these costs for a study at HSRC. Suppose the total number of records expected is 2000. We are only interested in significant cost differences, those greater than $10. For this example, extreme values for the parameters are used to obtain conservative estimates.

The difference in development cost \( (D_{56} - D_{57}) \) between the Comparison and Manual Proofread steps is easy to estimate since Manual Proofread requires very little
development. Based on previous studies at HSRC, $50 is a generous estimate of Comparison development cost. Thus, the maximum difference would be on the order of $50.

The processing costs included in DIFF_{21}(T) are for comparing records, finding corrections after Comparison, and updating error records. The difference in cost between data entry and proofreading (P_{515} - P_{57}) is negligible because dependent verification is not included and students would do both jobs. In the experimental studies, proofreading took essentially the same amount of time as entry. The cost of comparing records depends on whether a mainframe or microcomputer is used. Since microcomputer processing costs are considered negligible at HSRC, we will use mainframe costs. The average cost per record from previous studies was $0.02, which would mean a total (NP_{56}) of $40 for this example. The remaining processing costs for finding and updating error records (E(W)P_{53}, E(Y - U)P_{54}) depend on error rates and detection rates as well as the average cost parameters.

Suppose the average cost for finding or updating a record is $0.25. These are high values, nearly double the values from actual studies at HSRC. Using the conservative parameter values shown below, the maximum cost difference is $125.

\[
\begin{align*}
e &= 0.10 & \text{data entry error rate,} \\
c &= 0.90 \text{ to } 1.00 & \text{Comparison detection rate given error exists,}
\end{align*}
\]

\[
\begin{align*}
r_{11} &= 0.40 \text{ to } 0.60 & \text{Manual Proofread detection rate given error exists,}
\end{align*}
\]

\[
\begin{align*}
r_{10} &= 0 & \text{Manual Proofread detection rate given no error exists.}
\end{align*}
\]

\[
\begin{align*}
E(W)P_{53} &= Nw(0.25) = 2000[2e - (1 + m)e^2](0.25), \\
E(Y - U)P_{54} &= N[y - u](0.25) = 2000[ce - cme^2 - (r_{10} + (r_{11} - r_{10})e)](0.25).
\end{align*}
\]

In this case, the total cost difference estimate comes to $215. Thus, the advantage of a lower final error rate must be balanced against increased cost. For a data entry error rate of ten percent and a proofread detection rate of fifty percent, the Redundancy sub-system has a final error rate from three to five percent less than Manual Proofread.
One approach to justifying the increased cost of a more effective subsystem is based on the "cost" of undetected errors. O'Reagan (1969) defined this "cost" in his model as the cost of a correction done after regular processing was completed. In practice, such a "cost" is difficult to estimate. However, Model 5 estimates can provide additional information on which to base a decision. In order to consider the cost difference in relation to the difference in the number of undetected errors, let \( R_{ij} \) be the cost-error ratio between Subsystem \( i \) and Subsystem \( j \). For Subsystems 1 and 2:

\[
R_{21} = \left| \frac{\text{DIFF}_{21}(T)}{\text{DIFF}_{21}(Z)} \right|, \quad \text{DIFF}_{21}(Z) = N(\text{DIFF}_{21}(z)).
\]

For this example,

\[
R_{21} = \left| \frac{50 + 0 + 2000(0.02) + 2000(0.25)(2 - (1 + m)e + c - cme - r_{11})}{2000(ckme + (kr_{11} - ck))} \right|.
\]

Using the count parameter values specified earlier, the value of \( R_{21} \) is about two. The implication of this statistic is that choosing the more expensive, but better, subsystem would be appropriate if the cost of correcting a record later were greater than $2. Making a correction later, after regular processing is completed, would be necessary if an undetected error were discovered during analysis of the working file.

We can conclude that, for RDM environments similar to HSRC, using a Redundancy subsystem is the better choice if data quality is of any concern. The lower cost of using Manual Proofread may be deceptive due to the potential cost of making corrections later. A RDM system based on Redundancy probably would have lower final error rates with relatively little difference in total system cost.
4.6.2 Manual Proofread versus Primary VVE

In comparing Manual Proofread, Subsystem 1 of Model 5, and Primary VVE, Subsystem 3, we consider both intelligent data entry and batch VVE. As shown in Figure 4.7, neither subsystem includes dependent verification. Investigating final error rate differences leads to an obvious conclusion, while cost examples show that the "best" subsystem depends on the situation.

Manual Proofread has a lower final error rate than Primary VVE when the Manual Proofread detection rate is higher than the VVE detection rate, \( r_{11} > d_{111} \). This follows from \( \text{DIFF}_{13}(z) = e_{1}k(d_{111} - r_{11}) \), for which we assume \( r_{01} = 0 \), and the same entry error rate applies to both subsystems. The type of information being studied is an important factor in determining whether proofreading or VVE is more effective. VVE is probably more effective when most of the fields are categorical, while proofreading may be better for numeric fields that do not have well defined valid ranges.

For cost examples, consider three situations that compare: 1) average proofreading \( (r_{11} = 0.50) \) and intelligent data entry, 2) average proofreading and batch VVE that is equally ineffective \( (r_{11} = 0.50 \text{ and } d_{111} = 0.50) \), or 3) average proofreading and effective batch VVE \( (r_{11} = 0.50 \text{ and } d_{111} = 0.90) \). Suppose \( N = 2000 \), the data entry error rate is ten percent, and conservative cost values similar to those in Section 4.6.1 are used. Estimates of the total cost difference, \( \text{DIFF}_{13}(T) \), are shown below.

<table>
<thead>
<tr>
<th>Situation</th>
<th>Detection Rates</th>
<th>Cost Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Intelligent entry</td>
<td>( r_{11} = 0.50 ), any ( d_{111} )</td>
<td>$125</td>
</tr>
<tr>
<td>2) Poor batch VVE</td>
<td>( r_{11} = 0.50 ), ( d_{111} = 0.50 )</td>
<td>$15</td>
</tr>
<tr>
<td>3) Good batch VVE</td>
<td>( r_{11} = 0.50 ), ( d_{111} = 0.90 )</td>
<td>(-$25)</td>
</tr>
</tbody>
</table>

In the first situation the difference in total cost is \$125, assuming negligible costs for finding corrections and updating records during intelligent data entry.
Proofread is more expensive because costs for proofreading and updating are higher than the development cost of intelligent data entry. The cost difference in the second situation is insignificant since the cost of proofreading is almost completely matched by the cost of developing a batch VVE program and finding corrections after VVE. In the third situation Manual Proofread is cheaper than Primary VVE, but only because forty percent fewer errors are detected.

These examples demonstrate the importance of exploring interrelationships between detection rates and cost. Intuitively, Manual Proofread might appear to be a cost effective subsystem since no programming is required, but the model estimates show that VVE should be seriously considered before a decision is made. In some situations, using Subsystem 3 with batch Primary VVE may not cost much more than Subsystem 1, and may be worth the higher cost if VVE is potentially more effective than proofreading. Intelligent data entry is the obvious choice when it is less expensive and proofreading is expected to be relatively ineffective compared to VVE.

4.6.3 Redundancy versus Standard Processing

Even when a professional entry service is used, Redundancy is a valid alternative to standard processing. In this case, a standard RDM subsystem includes dependent verification followed by batch VVE. In Figure 4.8, Redundancy is Subsystem 2 and Standard Processing is Subsystem 3. Suppose that both the VVE and Comparison programs are run after entry is completed by a professional service.

We can find critical values for the effectiveness of dependent verification by exploring final error rate differences between the two subsystems. Suppose the data entry error rate, $e$, is five percent. Since dependent verification always lowers the data entry error rate, let $e_1 = qe$, $0 < q < 1$, for Standard Processing. For Redundancy, let $e_1 = e_2 = e$. Then

$$\text{DIFF}_{32}(z) = e(q - d_{111}kq - 1 + ck - cmke),$$
and Standard Processing is more effective than Redundancy if $\text{DIFF}_{32}(z) < 0$. Critical values for $q$, which indicates the effectiveness of dependent verification, are apparent when we simplify $\text{DIFF}_{32}(z)$ by fixing the values of the insignificant parameters ($k = 1$, $c = 1$, $m = 1$). If VVE is effective, with a detection rate ($d_{111}$) of ninety percent, then $q < 0.50$ must hold for Standard Processing to be better than Redundancy. If the VVE detection rate is only fifty percent, then Standard Processing is better only if $q < 0.10$. If the data entry error rate is five percent, even when dependent verification is very effective, $q = 0.01$, the actual difference in final error rates is less than 0.005. Redundancy is slightly more effective than Standard Processing when dependent verification detects less than half of the data entry errors. In general, these two subsystems produce very similar final error rates.

Since we are considering a situation in which a professional entry service is used, let $N = 5000$ for the cost example. Suppose development costs for VVE and Comparison, processing costs for VVE and Comparison, and processing costs for data entry and dependent verification are essentially the same. The cost difference, $\text{DIFF}_{32}(T)$, is $100 - 155$ for the following values of the remaining parameters: $m = 1$, $c = 1$, $e = 0.05$, $d_{111} = 0.90$ or 0.50, $q = 0.50$ or 0.10, $P_{53} = 0.25$ (processing cost of finding a correction), and $P_{54} = 0.25$ (processing cost of updating a record). Standard Processing is always more expensive in this situation.

This subsystem comparison applies when professional entry, without intelligent data entry, is the foundation of all available RDM system alternatives. While Standard Processing is probably cost effective under most conditions, using Redundancy without dependent verification may reduce final errors just as well and be less expensive.
FIGURE 4.8
Selected Model 5 Subsystems

Subsystem 2: Redundancy only

Subsystem 3: Standard Processing
(Dependent verification, batch Primary VVE)

Subsystem 6: Redundancy with intelligent data entry
Subsystem 8: Redundancy and Final VVE
4.6.4 Redundancy versus Redundancy and Final VVE

One approach to considering whether using Redundancy alone is enough, is to compare Subsystem 2 of Model 5 with Subsystem 8, which has Redundancy and Final VVE. Subsystem 8 should be more expensive since more steps are included, as shown in Figure 4.8, but would the decrease in the number of final errors pay off in the long run?

Subsystem 8 always has a lower final error rate than Subsystem 2, as shown in Figure 4.5. The difference, \( \text{DIFF}_{28}(z) = k d_{311} e_1(1 - ck + ckme_2) \), is greater than zero for all possible count parameter values. It is more interesting to discover that, while the maximum increase in actual final error rate is only one percent, adding Final VVE can produce a ninety percent improvement over the final error rate achievable by using Redundancy only. This improvement depends directly on the VVE detection rate, \( d_{311} \).

The cost difference is completely due to the additional Final VVE steps. In the RDM environment used for the other cost examples, Subsystem 8 is about $50 more expensive for \( N = 2000 \). Since we expect increased cost, the cost-error ratio defined in Section 4.6.1 is of interest. With effective VVE, a detection rate of ninety percent, \( R_{28} \) ranges from $2 to $10 for entry error rates of ten and five percent.

Considering Subsystem 8 as an alternative RDM system has another advantage in that the final decision can be postponed. The actual entry error rate can easily be computed once the Comparison phase is completed. A better estimate of \( R_{28} \) could then be used to decide if adding Final VVE would be a cost effective method for avoiding later corrections.

4.6.5 Redundancy versus Redundancy and Intelligent Data Entry

For the last subsystem comparison from Model 5, we explore the effect of adding intelligent data entry to Redundancy. The subsystems compared are Subsystem 2 and Subsystem 6, shown in Figure 4.8. Subsystem 2 uses simple Redundancy, no
dependent verification or other edit checking. The only difference in Subsystem 6 is the
addition of Primary and Secondary VVE, accomplished by using intelligent data entry.
As with Subsystem 8, we expect the final error rate to be lower than for simple Redund-
dancy, but we are interested in the cost consequences.

Considering \( \text{DIFF}_{26}(z) \) confirms that the addition of intelligent data entry always
lowers the final error rate,

\[
\text{DIFF}_{26}(z) = (e_1 - e_3) - ck(e_1 - e_3) + ckm(e_1e_2 - e_3e_4).
\]

Since they are probabilities, \( ck < 1 \), and from Theorem 3.5.5 and 3.5.6, \( e_1 > e_3 \),
\( e_2 > e_4 \). Thus, Subsystem 6 is always better because \( \text{DIFF}_{26}(z) > 0 \) always holds.
Although Subsystem 6 is slightly better than Subsystem 8, the maximum actual differ-
ence in final error rate compared to Subsystem 2 is still only one percent.

For cost examples, we consider two cost possibilities. Cost estimates are based
on processing 2000 records in an RDM environment similar to HSRC when micro-
computers are used for all computer-aided steps. Assuming negligible costs for finding
corrections and updating records during intelligent data entry, \( a_1 = a_2 = 0 \), then

\[
\text{DIFF}_{26}(T) = -D_{55} - 2NP_{55} + NP_{53}(w_{52} - w_{56}) + NP_{54}(y_{52} - y_{56})
\]
in which \( w_{52}, y_{52}, \) and \( w_{56}, y_{56} \) are the appropriate probability parameters for
Subsystem 2 and Subsystem 6, respectively. The development cost for VVE, \( D_{55} \), is
about $20 (one programmer hour) when simple microcomputer software is used for
intelligent data entry. Suppose the average added cost of entering records using intel-
ligent data entry, \( P_{55} \), is $0.005; for 2000 records this comes to $20. The two cost
possibilities differ in the processing costs for finding corrections and updating records.
The first has relatively high costs, \( P_{53} = $0.25 \) and \( P_{54} = $0.25 \), and the second is more
reasonable, \( P_{53} = $0.10 \) and \( P_{54} = $0.05 \). Total cost differences (\( k = 1, m = 1, c = 1 \))
for data entry rates of ten or five percent and VVE detection rates of ninety or fifty
percent are shown below (P.P. means Primary Procesing, F.C. means Finding
Corrections).
<table>
<thead>
<tr>
<th>Entry</th>
<th>VVE</th>
<th>P.P.</th>
<th>Non-match</th>
<th>Correction</th>
<th>F.C.</th>
<th>Update</th>
<th>DIFF_{26}(T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>d_{311}</td>
<td>e_{3}</td>
<td>w_{52} - w_{56}</td>
<td>y_{52} - y_{56}</td>
<td>$0.25$</td>
<td>$0.25$</td>
<td>$80.00$</td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>0.90</td>
<td>0.010</td>
<td>0.160</td>
<td>0.080</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.05</td>
<td>0.90</td>
<td>0.005</td>
<td>0.085</td>
<td>0.043</td>
<td></td>
<td></td>
<td>24.00</td>
</tr>
<tr>
<td>0.10</td>
<td>0.50</td>
<td>0.050</td>
<td>0.085</td>
<td>0.043</td>
<td></td>
<td></td>
<td>24.00</td>
</tr>
<tr>
<td>0.05</td>
<td>0.50</td>
<td>0.025</td>
<td>0.046</td>
<td>0.023</td>
<td></td>
<td></td>
<td>- 5.50</td>
</tr>
<tr>
<td>0.10</td>
<td>0.90</td>
<td>0.010</td>
<td>0.160</td>
<td>0.080</td>
<td>$0.10$</td>
<td>$0.05$</td>
<td>$0$</td>
</tr>
<tr>
<td>0.05</td>
<td>0.90</td>
<td>0.005</td>
<td>0.085</td>
<td>0.043</td>
<td></td>
<td></td>
<td>- 18.70</td>
</tr>
<tr>
<td>0.10</td>
<td>0.50</td>
<td>0.050</td>
<td>0.085</td>
<td>0.043</td>
<td></td>
<td></td>
<td>- 18.70</td>
</tr>
<tr>
<td>0.05</td>
<td>0.50</td>
<td>0.025</td>
<td>0.046</td>
<td>0.023</td>
<td></td>
<td></td>
<td>- 28.50</td>
</tr>
</tbody>
</table>

These cost examples show that the additional costs of using intelligent VVE ($-D_{55} - 2NP_{55} = -40$) are offset because fewer records require processing after Comparison than with simple Redundancy. When the VVE detection rate is high and costs are high for finding corrections and updating records then Subsystem 6 can even be less expensive than Subsystem 2. When the additional cost of intelligent data entry is relatively low, less than $50, Subsystem 6 is probably a better choice than Subsystem 2 regardless of the entry error rate or the VVE detection rate.

4.6.6 Redundant Collection versus Primary Collection only

Model 6, the data collection model, only has one pair of interesting subsystems. Both subsystems include redundant entry, but Subsystem 1 also has redundant data collection (see Figure 3.6.2). The additional collection phase decreases the final error rate, but the associated cost increase may be unacceptable.

An expression for DIFF_{12}(z) can be derived, using Theorem 3.6.5, that shows Subsystem 1 is always better, i.e., DIFF_{12}(z) < 0 for all parameter values. Since the expression is complex, Figure 4.9 presents numerical results for DIFF_{12}(z) for selected values of the count parameters.


**FIGURE 4.9**

**Model 6 Final Error Rates**

The following parameter values were used for this table:

- \( c_6 = 0.95 \)  
  Conditional detection rate at Comparison
- \( m_6 = 0.50 \)  
  Conditional probability of matched errors
- \( h = 0.50 \)  
  Conditional compound error probability: \( h_{11} = h_{22} = h_{12} = h_{111} = h \)
- \( k = 1.00 \)  
  Conditional probability of proper correction

<table>
<thead>
<tr>
<th>Input Error Rates</th>
<th>Final Error Rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection ( t_1 = t_2 = t )</td>
<td>Entry ( e_1 = e_2 = e )</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>0.10</td>
<td>0.10</td>
</tr>
<tr>
<td>0.05</td>
<td>0.01</td>
</tr>
<tr>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>0.05</td>
<td>0.10</td>
</tr>
<tr>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>0.01</td>
<td>0.10</td>
</tr>
<tr>
<td>0.05</td>
<td>0.03</td>
</tr>
<tr>
<td>0.01</td>
<td>0.01</td>
</tr>
</tbody>
</table>

A simple cost example demonstrates that, while total cost difference may be substantial, final error differences should also be evaluated. Suppose a study requires coding information from 1000 existing diagrams. From Definition 3.6, the cost difference is

\[
\text{DIFF}_{12}(T) = \text{NP}_{60} + \text{NP}_{63}(w_{61} - w_{62}) + \text{NP}_{64}(y_{61} - y_{62}).
\]

Suppose people who are paid $5 an hour can code about ten records an hour, which means the average data collection processing cost, \( P_{60} \), is $0.50. Consider an extreme situation with high processing costs for finding corrections and updating records, \( P_{63} = $0.25 \) and \( P_{64} = $0.25 \). Let the data collection and entry error rates be the same for both Primary and Secondary Processing \((t_1 = t_2 \text{ and } e_1 = e_2)\). Comparisons are shown below for data entry error rates of five percent, \( e_1 = e_2 = 0.05 \).
\[ t_1 = t_2 = t \]

<table>
<thead>
<tr>
<th></th>
<th>( \text{DIFF}_{12}(T) )</th>
<th>( \text{DIFF}_{12}(Z) )</th>
<th>( R_{12} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10</td>
<td>$538</td>
<td>37</td>
<td>$15</td>
</tr>
<tr>
<td>0.05</td>
<td>520</td>
<td>19</td>
<td>27</td>
</tr>
<tr>
<td>0.01</td>
<td>504</td>
<td>4</td>
<td>126</td>
</tr>
</tbody>
</table>

The overall cost difference is due mainly to the additional cost of Secondary Collection \((N_{P60} = 500)\), but the large variation in the cost-error ratio shows the importance of including final error differences in any decision. When few errors are made during data collection \((t = 0.01)\), redundant collection may not be worthwhile since \(R_{12}\) is over $100. However, when the data collection error rate is high, greater than ten percent, the using Redundant Collection could prove cost effective in the long run.

Since \( \text{DIFF}_{12}(T) \) is very sensitive to data collection processing cost, a reasonable collection cost value must be available before Model 6 cost estimates are useful in making an informed decision. However, estimating the subsystem cost difference is straightforward when cost values are definable. A table of probability parameter values \((w_{61} - w_{62}, y_{61} - y_{62})\) is provided in Appendix A for common count parameter values.
CHAPTER 5
APPLICATIONS

In this chapter the models are applied to actual RDM situations. The validity of the modeling approach is demonstrated by comparing the results of implementing a subsystem with model estimates. The descriptions of the experimental studies provide additional examples of using the modeling approach as a decision tool. The results of a controlled experiment designed to investigate Model 1 are presented. Comparisons of model predictions and actual results are discussed for two other studies for which Model 3 is appropriate (Primary and Secondary VVE, Comparison, or Manual Proofread).

5.1 RDM Environment

As mentioned in Chapter 4, the experimental studies were done at HSRC. The staff at HSRC (less than thirty people) do multiple, concurrent, statistical research projects. Statistical analysis is generally performed using SAS software on a mainframe. When required, data entry is usually done in-house on microcomputers (Apple® //e, IBM® PC, IBM PC AT), although professional entry services are also used for larger (more than 5000 records) studies. Simple file programs, with limited validation capabilities, are used for microcomputer data entry, e.g., PC-File III. In-house data entry and manual proofreading are done mostly by undergraduate “work-study” students who are hired for general services, not specifically to do data processing. When necessary, research staff members also enter data and do proofreading. Professional
computer programmers are responsible for RDM tasks, which include planning and supervising data entry.

At HSRC, a study that requires data entry may be part of a complex project. Each study involving data entry is treated separately since no formal RDM system exists for data preparation from original information. Typically, all RDM system development and processing takes less than a month for one study; work is never done without the distraction of other activities.

5.2 Model 1 Experiment

The BAC study was an experiment to investigate simple Redundancy and Manual Proofread. The two RDM systems implemented correspond to the subsystems of Model 1, which are also Subsystems 1 and 2 of Model 5 (see Figure 2.2 and Figure 3.5.2). Half of the data were processed with Manual Proofread as the edit phase and the other half were processed using redundant data entry and Comparison.

5.2.1 RDM Situation

The BAC study was done in 1985, as part of a complex HSRC project, and completed in about two weeks. Students did data entry and proofreading. A simple filing program (PDMS) was used for entry on an Apple //e; no dependent verification was possible. All other processing was done by one programmer on a mainframe (TSO QED, SAS). The Manual Proofread subsystem processed 612 records and the Redundancy subsystem had 614 records.

The original information consisted of three simple fields: Driver's License Number (7 digits), Accident Date (DDMMYY, 6 digits), and Blood Alcohol Content (BAC, 2 digits). The input data were taken from a computer listing on which handwritten BAC values had been added for most lines (about 30 lines per page). The data
for one record came from one line of information. Only lines with BAC values were entered and additional fields on the listing were ignored. Rounding of BAC values was necessary for about twenty percent of the records.

5.2.2 Subsystem Comparison

Model predictions for the two subsystems were made before implementation. Proposed parameter values were based on previous experience at HSRC. We expected data entry error rates between one and five percent. Manual Proofreading was considered poor, with a detection rate ranging from fifty to ninety percent (0.50 ≤ r₁₁₁ ≤ 0.90). The conditional probability for detecting a non-existent error during Manual Proofreading was considered negligible. After Comparison, error detection for non-matches (Secondary Proofread in Model 1, Finding Corrections in Model 5) was considered good, with a detection rate of ninety percent or higher (c₁ ≥ 0.90). No limitations were placed on the conditional probability of a matched error (0 ≤ m₁ ≤ 1). Proper correction after updating was considered very likely (k = 0.99).

Estimates were made for several combinations of count parameter values within the ranges specified. Count estimates were calculated by a simple program (written in Microsoft® BASIC). The results for a five percent data entry error rate (e₁ = e₁ = 0.05) are shown in Figure 5.1.

The final error rate estimates, with a five percent entry error rate, are 0.5 – 2.5% using Manual Proofread and 0.1 – 0.8% using Redundancy. As we expected from the results in Section 4.6.2, a Redundancy subsystem has lower predicted final error rates. Manual Proofread has final error rates as low as Redundancy only when proofreading is very good (r₁₁₁ ≥ 0.90).
FIGURE 5.1
Predicted Final Error Rates for BAC Study

<table>
<thead>
<tr>
<th>Manual Proofread Subsystem</th>
<th>Redundancy Subsystem</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{11}$</td>
<td>Final Error Rate</td>
</tr>
<tr>
<td>0.50</td>
<td>0.025</td>
</tr>
<tr>
<td>0.75</td>
<td>0.013</td>
</tr>
<tr>
<td>0.90</td>
<td>0.005</td>
</tr>
</tbody>
</table>

|                |                    | 0.95  | 0      | 0.003 |
|                |                    | 0.95  | 0.50   | 0.004 |
|                |                    | 0.95  | 1.00   | 0.005 |

|                |                    | 0.99  | 0      | 0.001 |
|                |                    | 0.99  | 0.50   | 0.002 |
|                |                    | 0.99  | 1.00   | 0.003 |

The difference in predicted cost between the two subsystems depends mainly on the development and processing costs of the Comparison phase. The total difference is the sum of the Comparison phase costs and the additional cost of processing a larger number of records with detected errors. Using high average manual processing costs, $0.50 for finding a correction and $0.25 for updating an error record, the additional costs for these steps are $30 and $10, respectively. Together with the cost of running the comparison program, the maximum estimate of cost difference for processing is about $50. From previous experience at HSRC, the costs associated with Comparison were estimated as $45 for labor and $15 for computer charges.

Based on model predictions the Redundancy subsystem was the better choice for the BAC study. Data quality was more important than cost, assuming a reasonable cost difference. The predicted total cost difference, which is extremely inflated, of about $100 was an acceptable price for fewer errors. The cost-error ratio has a maximum predicted value of $11. Redundancy was potentially a very cost-effective subsystem for this study; correcting a record at a later stage of the project would require repeating a relatively expensive (at least $50) matching process.
5.2.3 Actual Results

The implementation of both subsystems was closely monitored. Information was kept about time spent and computer costs. Undetected errors and total costs were determined after processing was completed. Undetected errors were discovered by extremely careful proofreading.

Actual count parameter values did not differ unreasonably from the values used for model predictions. The actual data entry error rates (Primary Entry for the Manual Proofread subsystem, Primary and Secondary Entry for the Redundancy subsystem) were between two and three percent, compared to the proposed range of one to five percent. For Manual Proofreading, the actual detection rate was below the worst proposed value of fifty percent. Students detected four error records, while seven were undetected, for a detection rate of thirty-six percent (4 / 11). No records with non-existent errors were detected by proofreading. Using Redundancy, all corrections were found for non-matched primary records (c1 = 1.00), and no matched paired errors occurred. All updates were done correctly.

The results of the BAC study experiment compare favorably with model estimates. The expected number of undetected errors for a Manual Proofread subsystem is $9 \pm 3$ s.d. ($e_1 = 0.03$, $r_{111} = 0.40$); the actual count was seven. No undetected errors were discovered in the final dataset of the Redundancy subsystem; the expected number is $1 \pm 1$ s.d. The expected number of errors found ($E(Y)$) for the Comparison phase is $14.5 \pm 3.8$ s.d., using a data entry error rate of three percent ($e_1 = e_2 = 0.03$). Fifteen error records were actually found using Redundancy. The actual difference in total cost was $30$, which is less than the maximum difference predicted. The actual cost-error ratio was $4$ for the data processed by the Manual Proofread subsystem, assuming Redundancy would have detected all errors.
5.2.4 Discussion

The BAC study experiment shows that the modeling approach provides useful decision information. Model estimates are reasonable and could have been used to choose the better subsystem. Considering error generation using a record as the measurement unit was no problem with only three fields. Although cost estimates were less accurate than count estimates, modeling provided objective evidence about the cost-effectiveness of the more expensive subsystem.
5.3 Using Redundancy and Batch VVE

Valid value editing, by a batch program, and Redundancy were used for the Nevada study. Originally, the only edit phase for detecting entry errors was Primary VVE. Secondary Entry was performed several months later. In addition to comparing count estimates with actual results, we consider "actual" results for subsystems other than the one implemented. Relationships between the types of mistakes made during data entry and detection rates are discussed.

5.3.1 RDM Situation

The Nevada Study was based on a survey of people at Driver's License stations in Nevada; 937 survey forms were processed. Data processing began in 1985 and was completed in 1986. As in the BAC Study, students entered data on an Apple //e using a simple filing program, which did not allow dependent verification or valid value edit checking. A simple batch VVE program for univariate valid value edit checking was written on the mainframe (SAS). Primary and secondary records were compared on an IBM PC AT (BASS Compare Procedure). Updating was accomplished on a microcomputer using a filing program (PC-File III).

The survey form consisted of thirteen questions on one legal-sized page (see Appendix B). With one exception, the questions had multiple-choice answers. The exception requested a numeric response between 0 – 100, and most responses were common values such as 50 or 100. Responses circled by respondents (a, b, c, etc.) were entered as capital letters (A, B, C, etc. with CAPS LOCK on). A sequential identification number (four digits) was also entered.
5.3.2 Error Generation

The implemented subsystem was Subsystem 5 of Model 5 (Redundancy after Primary VVE), but additional processing was done to allow comparisons of count estimates with actual results for several subsystems. Figure 5.2 provides count estimates, using standard parameter values, for Subsystem 2 (simple Redundancy) and Subsystem 5. Suppose data entry error rates are the same, \( e_1 = e_2 = e \), and \( e = 0.05 \) or \( e = 0.10 \). VVE detection is considered as poor, fair, or good (\( d_{111} = 0.25, 0.50, 0.75 \)). The following parameters have fixed values: \( m = 0.50 \), \( c = 0.95 \), \( k = 0.95 \) (see Appendix A: Model 5 Count Parameters). The actual data entry error rates for Primary and Secondary Entry (\( e_1, e_2 \)) were 0.08 and 0.09 (76 and 82 records), respectively. Primary and secondary records were checked separately by VVE for actual detection rates of 0.34 and 0.37. Of eight paired records in which both had errors, only one was a matched error (\( m = 1 / 8 = 0.125 \)).

**FIGURE 5.2**

Count Estimates for Redundancy and Primary VVE (\( N = 937 \))

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Entry Rate</th>
<th>VVE</th>
<th>Final Error Rate</th>
<th>Error Count E(Z) s.d.</th>
<th>Non-match Count E(W) s.d.</th>
<th>VVE Detected Count E(V_1) s.d.</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.05</td>
<td></td>
<td>0.006</td>
<td>6 2.4</td>
<td>90 9.0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.05 0.25</td>
<td></td>
<td>0.005</td>
<td>5 2.1</td>
<td>80 8.5</td>
<td>12 3.4</td>
</tr>
<tr>
<td>5</td>
<td>0.05 0.50</td>
<td></td>
<td>0.003</td>
<td>3 1.8</td>
<td>69 8.0</td>
<td>23 4.8</td>
</tr>
<tr>
<td>5</td>
<td>0.05 0.75</td>
<td></td>
<td>0.002</td>
<td>2 1.3</td>
<td>59 7.5</td>
<td>35 5.8</td>
</tr>
<tr>
<td>2</td>
<td>0.10</td>
<td></td>
<td>0.014</td>
<td>14 3.7</td>
<td>173 11.9</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.10 0.25</td>
<td></td>
<td>0.011</td>
<td>11 3.3</td>
<td>154 11.4</td>
<td>23 4.8</td>
</tr>
<tr>
<td>5</td>
<td>0.10 0.50</td>
<td></td>
<td>0.007</td>
<td>7 2.7</td>
<td>136 10.8</td>
<td>47 6.7</td>
</tr>
<tr>
<td>5</td>
<td>0.10 0.75</td>
<td></td>
<td>0.004</td>
<td>4 2.0</td>
<td>117 10.1</td>
<td>70 8.0</td>
</tr>
</tbody>
</table>
We can investigate four subsystems of Model 5 using the Nevada study. After all regular processing was complete, "actual" results were determined for three subsystems (2, 3, 6) as if each had been implemented. For Primary VVE alone, Subsystem 3, we expected between 14 and 38 error records in the final dataset for \( e_1 = 0.05 \) and \( d_{111} \) from 0.25 to 0.75 (\( z_{53} = (1 - kd_{111})e_1 \)). The actual number detected by VVE was 26 for the primary records (\( e_1 = 0.08, d_{111} = 0.30 \)) and 30 for the secondary records (\( e_2 = 0.09, d_{211} = 0.37 \)). Using Redundancy only, Subsystem 2, the actual number of final errors, four records, compared reasonably with the model estimates. Compared to predictions of 90 and 173 (\( e = 0.05, 0.10 \)) non-matches, 153 occurred, which is plausible since the actual data entry rate was about eight percent. When Comparison was done after Primary VVE, Subsystem 5, the actual number of final errors was four, and 132 non-matches occurred. If we eliminate the errors detectable by Secondary VVE, 104 non-matches remain, which is within the predicted range for Subsystem 6 with an eight percent entry error rate and a thirty percent VVE detection rate.

5.3.3 Discussion

Analyzing the types of mistakes in the Nevada data provides insights about error detection capabilities of the different subsystems. More than half of the mistakes can be traced to response problems, e.g., unclear choices or unanswered questions. Another major problem involved column shifting, which was aggravated by the microcomputer software used for entry. Most of the records in which a shift occurred had multiple mistakes as a result.

If only one edit method was used, Redundancy was clearly more effective for this data than valid value editing alone. The type of mistakes detected by VVE were mainly shift mistakes (88%); mistakes due to unclear responses or entry switch errors, which were important, were not detected. Redundancy, with or without VVE, also had the
advantage that ambiguous responses were detectable. Since the researcher was made aware of ambiguous responses (seven detected by Comparison), the effect of decisions made by the students during data entry was reduced. Entry mistakes in the sequential identification number were detectable by either Redundancy or VVE; however, no special programming was needed in the Comparison program to detect errors in this field.

Batch VVE, with its associated development cost, was not cost-effective for the Nevada data. Adding Primary VVE to Redundancy was not worthwhile since all of the 26 error records detected by Primary VVE were detectable by Redundancy alone (Subsystem 2). A cost analysis of Subsystem 2 and Subsystem 3 (Primary VVE alone) showed Subsystem 2 would be about $35 more expensive, based on actual costs. The higher cost was due to the cost of developing the comparison program and dealing with more detected errors. The cost of Secondary Entry in Subsystem 2 and the development cost for VVE in Subsystem 3 cancelled each other out. Subsystem 3 had 50 undetected errors and Subsystem 2 had four. Thus, the cost-error ratio was less than $1. When valid value editing is ineffective because of the data types involved, using simple Redundancy instead of a batch VVE program should be considered seriously as a RDM alternative.
5.4 Intelligent Data Entry, Redundancy, or Manual Proofread

The Bike study at HSRC provided information about situations in which inexpensive intelligent data entry is available. The advantages of Redundancy over Manual Proofread in this case are discussed.

5.4.1 RDM Situation

The input data for the Bike study was a computer listing containing information about 247 bicycle accidents. One line corresponded to one record and all data fields were numeric. All data processing was done on an IBM PC AT. Data was entered by a programmer trainee using a filing program (PC-File III) with limited valid value editing capabilities. Range checks were possible for single digit fields and other fields were screened for non-numeric characters. Of the 31 fields, only four were more than one digit. Developing the "masking" for intelligent data entry took the programmer trainee thirty minutes (@$7 per hour). The effectiveness of interactive VVE was limited because 22 of the fields had only two values, zero or one. Undetectable transposition or switch errors were likely to occur during data entry. The comparison program was created easily (less than thirty minutes using the Compare Procedure of BASS).

5.4.2 Discussion

Model estimates suggested that using Redundancy to detect errors in the Bike data would entail minimal additional processing costs. The expected number of non-matches that would require further processing was $18 \pm 4$ for a five percent entry error rate and a twenty-five percent VVE detection rate. Finding corrections and updating was a quick, inexpensive process. Original information was easily accessible and the filing program used for entry also allowed efficient updating. The actual entry error rate was six percent
and 30 non-matches occurred. Finding corrections and updating took about thirty minutes, for a cost of less than $5 (@$7 per hour).

For this data, Redundancy with intelligent data entry was obviously better than using Manual Proofread. After processing was completed, extremely careful proofreading was done to search for undetected errors for system evaluation purposes. The total amount of time required for proofreading was essentially the same as that required for Secondary Entry. Thus, the cost of an effective Manual Proofread subsystem would not be less than Redundancy. In addition, proofreading could not be done effectively in the same elapsed time as data entry. The detection capabilities of interactive VVE was questionable in this case, but the low development cost meant there was little to lose. Although intelligent data entry was not sufficient as the only edit method, the combination of interactive VVE with Redundancy was a cost-effective RDM system for the Bike study.
CHAPTER 6
CONCLUSIONS AND SUGGESTIONS

This research presents a modeling approach for investigating RDM alternatives applicable to relatively small studies. We have created a model framework within which a variety of models are developed. The models are used to explore error generation in data processing and for cost/benefit analysis of the modeled RDM alternatives. The models are designed for a target situation (see Figure 1.1) that is increasingly common in applied statistical research. People are encouraged to do more small studies utilizing microcomputers with data entry and statistical analysis capabilities. Such simple RDM environments provide opportunities for developing useful models because of the inherent limitation on the number of available methods for reducing errors. Our choice of RDM systems reflects a special interest in evaluating the use of redundancy as an edit method, for which implementation is relatively simple.

6.1 Conclusions

The flexibility of the model framework is an advantage. Considering a greater variety of RDM situations is possible because we are not limited to one fixed model. The model extensions in Chapter 3 follow naturally from the basic model presented in Chapter 2. By separating system components, one model can apply to newer computer-aided data collection and data entry methods, as well as to more standard RDM methods. Examples are discussed in Chapters 4 and 5 of modeling intelligent data entry by choosing appropriate parameter values.
An important aspect of these models is their utility. The probability structure of the models is validated by comparing model predictions to actual results. By treating a record as the measurement unit, the models remain tractable. The calculations required to estimate final error rates or perform a cost/benefit analysis are straightforward. Although many parameter values must be known, the discussion in Chapter 4 showed that appropriate values are not difficult to obtain. Relying on real RDM experience provides a much better basis for system evaluation than complex theoretical arguments. As shown by the experimental studies discussed in Chapter 5, using this modeling approach can provide objective evidence for making informed RDM decisions.

Exploring RDM concerns with a model has advantages. We find modeling focuses investigations into cost/benefit issues. The modular framework of the models allows us to consider individual processing steps more easily. In defining specific parameters for each processing stage, we are forced to speculate on the effect of changing the values of those parameters. Investigations can concentrate on the important RDM factors since relatively insignificant parameters can be identified and held at reasonable, fixed values. An interesting insight gained from exploring the models is that better data quality is not always achieved by spending more money. Modeling allows us to investigate quantitatively many more RDM situations than would be possible otherwise.

6.2 Suggestions for Further Research

Further extensions of this modeling approach could prove very useful. Our intent is to encourage research in RDM areas by presenting the foundations of a modeling framework and demonstrating model estimation capabilities. More complex models, with additional processing phases, remain to be defined and investigated. The inclusion of phases for inventory functions and more complicated update processes would create models applicable to larger studies. Considering time as a factor could produce an even more useful decision model. Investigation of other subsystem comparisons using the
existing models is suggested, either of different pairs or for different RDM situations. Since the lowest predicted final error rates in Model 5 are for Subsystem 6 (intelligent data entry and Redundancy), complete cost/benefit analysis for more subsystems and Subsystem 6 may be worthwhile. The modeling approach could be applied to studies in appropriate RDM environments that have more records (20,000 – 100,000) than the examples provided.

More research is needed into the nature of non-sampling data collection error rates, data entry error rates, and error detection rates. These rates are crucial to any model of error generation processes. Since past attempts at analytical treatments have not yielded practical results, the problem remains unsolved. In the absence of theoretical knowledge, we strongly suggest that data processing error rates and error detection rates be reported for studies involving RDM. For reported rates to be most useful, the measurement unit must be clearly defined and a brief description of the data fields should be included.

6.3 Redundancy as a RDM Principle

Our model definitions and investigations are influenced by an interest in regarding redundancy as a RDM principle. The various final error rate comparisons and cost/benefit analysis examples clearly demonstrate the effectiveness of using redundant data processing steps in detecting errors. We focus on the use of simple Redundancy to provide initial conclusions about the effectiveness of redundancy alone; possible interactions with other edit methods are avoided.

The advantages of redundancy in RDM go beyond a simple cost/benefit argument. Analyses of error records have shown that transposition and shift mistakes are major sources of errors. Standard edit methods often cannot detect such mistakes and can never detect switch errors unless multivariate edit tests are possible. Redundancy is a simple edit method with the potential to detect mistakes of all types, including
transpositions, shifts, and switches. Implementing a redundancy RDM system is more workable in some environments because less sophisticated computer knowledge is required. Packaged programs are available that compare datasets and report discrepancies, e.g., BASS Compare Procedure (microcomputers under MS/PC DOS), SAS PROC COMPARE (microcomputers under PC DOS, minicomputers, IBM mainframes), or a SAS macro %COMPARE (Rosen and Kanciruk (1985)). An original comparison program is generally simpler to create and test than an effective valid value edit program. Manual proofreading is easy to implement, but using redundancy is usually more effective in detecting errors. Thus, in many situations a redundancy edit method is both more effective and easy to implement.

As intelligent data entry becomes relatively inexpensive, due to increased availability of microcomputers, the importance of redundancy as a RDM principle for edit detection will increase. People who enter data using inexpensive microcomputer database programs, with valid value editing during entry, are more inclined to think that no further editing is necessary. One reason for proposing our modeling approach to RDM system evaluation is to provide quantitative evidence to the contrary.
REFERENCES


GLOSSARY

batch VVE: when VVE is accomplished by a separate computer program

Combination: the computer-aided process of creating a dataset of all processed records

Comparison: the computer-aided process of comparing corresponding primary and secondary records in a RDM system with redundant processing

Comparison correction: a correction found from investigating primary records that do not match at Comparison

component: a single RDM process, such as data entry, that may appear more than once in a system

compound error: an error produced by 1) a data collection error, and 2) a data entry error in the same record

correct record: a record that accurately reflects original input information

cost-error ratio: the absolute value of the ratio of the expected subsystem cost difference and the difference in the number of records with errors in the final dataset; denoted $R_{ij}$ for Subsystems i and j.

count parameter: an error creation or detection rate, defined on a per record basis

count random variable: defined to estimate the number of records for a particular step of an overall system

data collection: the manual process of recording information for eventual computer-aided analysis; may be separate from or combined with entry

data entry: the manual process, which may be part of intelligent data entry, of keying information into machine readable form; considered a separate process from dependent verification

dependent verification: the manual process of detecting errors by reprocessing data with both original information and initial processing results available to the verifier

development costs: fixed costs for designing and implementing computer programs, training, and overhead for a RDM system

entry: a manual process of entering data; may be done with or without dependent verification and/or intelligent data entry

error: a record that has one or more mistakes is counted as an error
error record: a record with one or more mistakes

final error rate: error rate based on the number of records with errors in the final dataset

finding corrections: the manual process of using original information to investigate records with detected errors

improper update: when an updated record is still an error record

independent verification: the manual process of detecting errors by reprocessing data with initial processing results unavailable to the verifier

intelligent data entry: computer-assisted data entry with valid value editing capabilities

initial records: edited primary records after Comparison and Manual Proofread corrections are made

Initial Processing: all steps used to produce initial records

interactive VVE: when VVE is accomplished by intelligent data entry

matched error: when paired records have identical mistake(s)

Merge: see Combination

paired records: corresponding primary and secondary records that are compared after a redundant process is done

overall system: all RDM processing steps available for creating a file ready for computer analysis

phase: a complete RDM process involving a one or more system components

Primary Collection: the data collection process that leads to primary records

Primary Entry: the entry process (data entry with or without dependent verification) that leads to primary records

Primary Processing: all RDM steps used to produce primary records

Primary VVE: the valid value edit process that checks primary records and may be part of Primary Processing

primary records: records produced by Primary Processing

processing costs: variable costs for processing records in a RDM system

proofreading: the manual process of comparing entry results with original information to detect errors

proper correction: when an updated record becomes a correct record
RDM: research data management

RDMS: research data management system

Redundancy: the combination of Secondary Entry and Comparison in the data entry models, Models 1-5, or the use of Secondary Collection in Model 6

redundant data entry: records are entered a second time using the same data entry process that was used for initial data entry

Secondary Collection: the data collection process that leads to secondary records

Secondary Entry: the entry process (data entry with or without dependent verification) that leads to secondary records

Secondary Processing: all RDM steps used to produce secondary records

Secondary VVE: the valid value edit process that checks secondary records and may be part of Secondary Processing

secondary records: records produced by Secondary Processing

switch error: an error that occurs when the value of a data field is valid, but not correct

subsystem: a subset of phases, from an overall system, that corresponds to an implemented RDM system

target study: a relatively simple research study that uses RDM methods to prepare data for computer analysis, see Figure 1.1

total cost: the sum of development and processing costs for a RDM system

Update: the process of updating records that have detected mistake(s)

undetected error: a record with one or more mistakes after being processed by an edit phase

valid value: a value for a data field that satisfies previously established guidelines

valid value edit: a simple edit check that compares the value of a field with the valid values for that field.

valid value edit process: the complete process that begins with a computer edit, which does valid value edits for all applicable fields, and ends with the update of records for which corrections are found

VVE: valid value edit process
### Count Parameters

<table>
<thead>
<tr>
<th>Phase</th>
<th>Model 1</th>
<th>Model 2</th>
<th>Model 3</th>
<th>Model 4</th>
<th>Model 5</th>
<th>Model 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Collection: Primary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>t₁</td>
<td></td>
</tr>
<tr>
<td>Data Collection: Secondary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>t₂</td>
<td></td>
</tr>
<tr>
<td>Data Entry: Primary</td>
<td>e₁</td>
<td>e₁</td>
<td>e₁</td>
<td>e₁</td>
<td>e₁</td>
<td>e₁</td>
</tr>
<tr>
<td>Data Entry: Secondary</td>
<td>e₂</td>
<td>e₂</td>
<td>e₂</td>
<td>e₂</td>
<td>e₂</td>
<td>e₂</td>
</tr>
<tr>
<td>VVE: Primary</td>
<td>d₁₁, ...</td>
<td>d₃₁₁₁, ...</td>
<td>d₁₁₁₁ ...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v₂₁</td>
<td>v₃₁</td>
<td>v₁</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE update: Primary</td>
<td>s₂₁</td>
<td>s₃₁</td>
<td>s₁</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>f₂₁ = v₂₁</td>
<td>f₃₁ = v₃₁</td>
<td>f₁ = v₁</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE: Secondary</td>
<td>d₃₂₁₁ ...</td>
<td></td>
<td>d₂₁₁ ...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v₃₂</td>
<td>v₂</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE update: Secondary</td>
<td>s₃₂</td>
<td>s₂</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>f₃₁ = v₃₂</td>
<td>f₂ = v₂</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Primary Processing</td>
<td>e₂₃</td>
<td>e₃₃</td>
<td>e₃</td>
<td>e₆₃, h₁₁</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secondary Processing</td>
<td>e₃₄</td>
<td></td>
<td>e₄</td>
<td>e₆₄, h₁₂, h₂₂</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comparison</td>
<td>m₁</td>
<td>m₂</td>
<td>m₃</td>
<td>m₁</td>
<td>m</td>
<td>m₆, h₁₁₁, e₆₁₁</td>
</tr>
<tr>
<td></td>
<td>w₁</td>
<td>w₂</td>
<td>w₃</td>
<td>w₄ = w₁</td>
<td>w</td>
<td>w₆</td>
</tr>
<tr>
<td>Comparison update</td>
<td>c₁</td>
<td>c₂</td>
<td>c₃</td>
<td>c₁</td>
<td>c</td>
<td>c₆</td>
</tr>
<tr>
<td></td>
<td>y₁</td>
<td>y₂</td>
<td>y₃</td>
<td>y₄ = y₁</td>
<td>y</td>
<td>y₆</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>r₁₁₁ ...</td>
<td>r₂₁₁ ...</td>
<td>r₃₁₁ ...</td>
<td>r₁₁₁ ...</td>
<td>r₁₁ ...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>u₁</td>
<td>u₂</td>
<td>u₃</td>
<td>u₄ = u₁</td>
<td>u</td>
<td></td>
</tr>
<tr>
<td>Initial Processing</td>
<td>e₄₅</td>
<td></td>
<td>e₅</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE: Final</td>
<td>d₄₁₁ ...</td>
<td>d₃₁₁ ...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v₄₃</td>
<td>v₃</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VVE update: Final</td>
<td>s₄₃</td>
<td>s₃</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>f₄₃ = v₄₃</td>
<td>f₃ = v₃</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Proper correction</td>
<td>k</td>
<td>k</td>
<td>k</td>
<td>k</td>
<td>k</td>
<td>k</td>
</tr>
<tr>
<td>Final errors</td>
<td>z₁</td>
<td>z₂</td>
<td>z₃</td>
<td>z₄</td>
<td>z</td>
<td>z₆</td>
</tr>
</tbody>
</table>
### Cost Parameters

<table>
<thead>
<tr>
<th>Cost Component</th>
<th>Model 1</th>
<th>Model 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Collection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entry</td>
<td>$D_{11}, P_{11}$</td>
<td>$D_{21}, P_{21}$</td>
</tr>
<tr>
<td>Dependent Verification</td>
<td>$D_{12}, P_{12}$</td>
<td>$D_{22}, P_{22}$</td>
</tr>
<tr>
<td>Find correction</td>
<td>$D_{14}, P_{14}$</td>
<td>$D_{23}, P_{23}$</td>
</tr>
<tr>
<td>Update</td>
<td>$D_{15}, P_{15}$</td>
<td>$D_{24}, P_{24}$</td>
</tr>
<tr>
<td>VVE</td>
<td>$D_{16}, P_{16}$</td>
<td>$D_{25}, P_{25}$</td>
</tr>
<tr>
<td>Comparison</td>
<td>$D_{13}, P_{13}$</td>
<td>$D_{26}, P_{26}$</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>$D_{14}, P_{14}$</td>
<td>$D_{27}, P_{27}$</td>
</tr>
<tr>
<td>Create complete file</td>
<td>$D_{16}, P_{16}$</td>
<td>$D_{28}, P_{28}$</td>
</tr>
<tr>
<td>VVE indicator</td>
<td></td>
<td>b, a</td>
</tr>
<tr>
<td>Redundancy indicator</td>
<td>s</td>
<td>s</td>
</tr>
<tr>
<td>Manual Proofread indicator</td>
<td>t</td>
<td>t</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cost Component</th>
<th>Model 5</th>
<th>Model 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Collection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entry</td>
<td>$D_{51}, P_{51}$</td>
<td>$D_{60}, P_{60}$</td>
</tr>
<tr>
<td>Dependent Verification</td>
<td>$D_{52}, P_{52}$</td>
<td>$D_{61}, P_{61}$</td>
</tr>
<tr>
<td>Find correction</td>
<td>$D_{53}, P_{53}$</td>
<td>$D_{62}, P_{62}$</td>
</tr>
<tr>
<td>Update</td>
<td>$D_{54}, P_{54}$</td>
<td>$D_{63}, P_{63}$</td>
</tr>
<tr>
<td>VVE</td>
<td>$D_{55}, P_{55}$</td>
<td>$D_{64}, P_{64}$</td>
</tr>
<tr>
<td>Comparison</td>
<td>$D_{56}, P_{56}$</td>
<td>$D_{65}, P_{65}$</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>$D_{57}, P_{57}$</td>
<td>$D_{66}, P_{66}$</td>
</tr>
<tr>
<td>Create complete file</td>
<td>$D_{68}, P_{68}$</td>
<td></td>
</tr>
<tr>
<td>VVE indicator</td>
<td>$b_1, b_2, b_3, a_1, a_2, a_3$</td>
<td></td>
</tr>
<tr>
<td>Redundancy indicator</td>
<td>s</td>
<td>$s_6$</td>
</tr>
<tr>
<td>Manual Proofread indicator</td>
<td>t</td>
<td>t</td>
</tr>
</tbody>
</table>

Cost estimation for Models 3 and 4 is based on Definition 3.5 from Model 5.
Bernoulli Random Variables

\[ \beta(\text{argument}) = \begin{cases} 1 & \text{if argument is true,} \\ 0 & \text{otherwise (i.e., argument is false).} \end{cases} \]

Each model only uses some of the following random variables.

- \( T_{1i} = \beta(\text{error made in record i during Primary Data Collection}) \)
- \( T_{2i} = \beta(\text{error made in record i during Secondary Data Collection}) \)
  
  \( T_{1i} \sim B(1, t_1), T_{2i} \sim B(1, t_2), \) all \( T_{1i} \) and \( T_{2i} \) assumed independent.

- \( X_{1i} = \beta(\text{error made in record i during Primary Data Entry}) \)
- \( X_{2i} = \beta(\text{error made in record i during Secondary Data Entry}) \)
  
  \( X_{1i} \sim B(1, e_1), X_{2i} \sim B(1, e_2), \) all \( X_{1i} \) and \( X_{2i} \) assumed independent.

- \( V_{1i} = \beta(\text{error detected in record i by Primary Valid Value Edit}) \)
- \( F_{1i} = \beta(\text{correction found for record i after Primary Valid Value Edit detection}) \)

- \( V_{2i} = \beta(\text{error detected in record i by Secondary Valid Value Edit}) \)
- \( F_{2i} = \beta(\text{correction found for record i after Secondary Valid Value Edit detection}) \)

- \( X_{3i} = \beta(\text{error exists in record i after Primary Processing}) \)
- \( X_{4i} = \beta(\text{error exists in record i after Secondary Processing}) \)

- \( W_i = \beta(\text{no match for record i at Comparison}) \)
- \( Y_i = \beta(\text{correction found for record i after Comparison detection}) \)

- \( U_i = \beta(\text{error detected for record i by Manual Proofread}) \)

- \( X_{5i} = \beta(\text{error exists in record i after Initial Processing}) \)

- \( V_{3i} = \beta(\text{error detected in record i by Valid Value Edit}) \)
- \( F_{3i} = \beta(\text{correction found for record i after Valid Value Edit detection}) \)

- \( Z_i = \beta(\text{error exists in Final Data for record i}) \)
Model 5 Count Parameters

Data Entry and Dependent Verification

\[ e_1 \quad = \quad \text{Primary Entry error rate} \quad = \quad P(X_{1i} = 1) \]
\[ e_2 \quad = \quad \text{Secondary Entry error rate} \quad = \quad P(X_{2i} = 1) \]

Comparison

\[ m \quad = \quad \text{conditional probability of matched paired error records} \]
\[ = \quad \text{Prob(match | both Primary and Secondary records have errors)} \]
\[ = \quad P(\text{same bad value(s)} | X_{3i} = 1 \& X_{4i} = 1) \]
\[ c \quad = \quad \text{conditional Comparison detection rate} \quad = \quad P(Y_i = 1 | X_{3i} = 1 \& W_i = 1) \]

Manual Proofread

\[ r_{11} \quad = \quad \text{conditional Manual Proofread detection rate given error exists} \]
\[ = \quad P(U_i = 1 | X_{3i} = 1 \& W_i = 0) \]
\[ r_{01} \quad = \quad \text{conditional Manual Proofread detection rate given no error exists} \]
\[ = \quad P(U_i = 1 | X_{3i} = 0 \& W_i = 0) \]

Valid Value Edit (VVE)

\[ d_{111} \quad = \quad \text{conditional Primary VVE detection rate} \quad = \quad P(V_{1i} = 1 | X_{1i} = 1) \]
\[ d_{211} \quad = \quad \text{conditional Secondary VVE detection rate} \quad = \quad P(V_{2i} = 1 | X_{2i} = 1) \]
\[ d_{311} \quad = \quad \text{conditional Final VVE detection rate} \quad = \quad P(V_{3i} = 1 | X_{3i} = 1) \]
\[ g_1 \quad = \quad \text{Prob(found | error exists & detected)} \quad = \quad P(F_{1i} = 1 | V_{1i} = 1 \& X_{1i} = 1) = 1 \]
\[ g_2 \quad = \quad \text{Prob(found | error exists & detected)} \quad = \quad P(F_{2i} = 1 | V_{2i} = 1 \& X_{2i} = 1) = 1 \]
\[ g_3 \quad = \quad \text{Prob(found | error exists & detected)} \quad = \quad P(F_{3i} = 1 | V_{3i} = 1 \& X_{3i} = 1) = 1 \]

Correction

\[ k \quad = \quad \text{conditional proper correction rate} \]
\[ = \quad \text{Prob(proper correction | error detected & correction found)} \]
\[ 1 \quad - \quad k \quad = \quad \text{conditional probability of an improper update} \]
Model 5 Theorems

3.5.1 \( V_1 \sim B(N, \nu_1), \) errors detected by Primary Valid Value Edit

\[ (5.1) \quad \nu_1 = d_{111}e_1 \]

3.5.2 \( F_1 \sim B(N, f_1), \) updates required by Primary Valid Value Edit

\[ f_1 = \nu_1 \]

3.5.3 \( V_2 \sim B(N, \nu_2), \) errors detected by Secondary Valid Value Edit

\[ (5.2) \quad \nu_2 = d_{211}e_2 \]

3.5.4 \( F_2 \sim B(N, f_2), \) updates required by Secondary Valid Value Edit

\[ f_2 = \nu_2 \]

3.5.5 \( X_3 \sim B(N, e_3), \) errors after Primary Processing

\[ (5.3) \quad e_3 = (1 - kd_{111})e_1 \]

3.5.6 \( X_4 \sim B(N, e_4), \) errors after Secondary Processing

\[ (5.4) \quad e_4 = (1 - kd_{211})e_2 \]

Lemma 3.5.1. All \( X_{3i} \) and \( X_{4i} \) are independent.

3.5.7 \( W \sim B(N, w), \) non-matches from Comparison

\[ (5.5) \quad w = e_3 + e_4 - (1 + m)e_3e_4 \quad \text{Redundancy} \]

3.5.8 \( Y \sim B(N, y), \) updates required by Comparison

\[ (5.6) \quad y = c [e_3(1 - e_4) + (1 - m)e_3e_4] \quad \text{Redundancy} \]
Model 5 Theorems (continued)

3.5.9  \( U \sim B(N, u) \), errors detected by Manual Proofread

\[
(5.7) \quad u_1 = r_{10}(1 - e_3) + r_{11}e_3 \quad \text{No redundancy}
\]

\[
(5.8) \quad u_2 = r_{10}(1 - e_3)(1 - e_4) + r_{11}me_3e_4 \quad \text{Both}
\]

3.5.10  \( X_5 \sim B(N, e_3) \), errors after Initial Processing

\[
(5.9) \quad e_{51} = e_3 \quad \text{No redun., no Manual Pf.}
\]

\[
(5.10) \quad e_{52} = r_{01}e_3 + (1 - k)(r_{10}(1 - e_3) + r_{11}e_3) \quad \text{No redun., Manual Pf.}
\]

\[
(5.11) \quad e_{53} = e_3 - ck(e_3 - me_3e_4) \quad \text{Redun., no Manual Pf.}
\]

3.5.11  \( V_3 \sim B(N, v_3) \), errors detected by Final Valid Value Edit

\[
(5.12) \quad v_3 = d_{311}e_5 \quad \text{For all subsystems}
\]

\[
(5.13) \quad v_{31} = d_{311}e_3 \quad \text{No redun., no Manual Pf.}
\]

\[
(5.14) \quad v_{32} = d_{311}[r_{01}e_3 + (1 - k)(r_{10}(1 - e_3) + r_{11}e_3)] \quad \text{No redun., Manual Pf.}
\]

\[
(5.15) \quad v_{33} = d_{311}[e_3 - ck(e_3 - me_3e_4)] \quad \text{Redun., no Manual Pf.}
\]

3.5.12  \( F_3 \sim B(N, f_3) \), updates required by Final Valid Value Edit

\[
f_3 = v_3
\]

3.5.13  \( Z \sim B(N, z) \), undetected errors in final dataset

\[
(5.16) \quad z = (1 - kd_{311})e_5 \quad \text{For all subsystems}
\]

\[
(5.17) \quad z_1 = (1 - kd_{311})e_3 \quad \text{No redun., no Manual Pf.}
\]

\[
(5.18) \quad z_2 = (1 - kd_{311})[r_{01}e_3 + (1 - k)(r_{10}(1 - e_3) + r_{11}e_3)] \quad \text{No redun., Manual Pf.}
\]

\[
(5.19) \quad z_3 = (1 - kd_{311})[e_3 - ck(e_3 - me_3e_4)] \quad \text{Redun., no Manual Pf.}
\]
Model 6 Count Parameters

Data Collection

\[ t_1 = \text{Primary Collection error rate} = P(T_{1i} = 1) \]
\[ t_2 = \text{Secondary Collection error rate} = P(T_{2i} = 1) \]

Data Entry and Verification

\[ e_1 = \text{Primary Entry error rate} = P(X_{1i} = 1) \]
\[ e_2 = \text{Secondary Entry error rate} = P(X_{2i} = 1) \]

Compound errors

\[ h_{11} = \text{Primary compound error probability} = P(\text{compound error} \mid T_{1i} = 1 \& X_{1i} = 1) \]
\[ h_{22} = \text{Secondary compound error probability} = P(\text{compound error} \mid T_{2i} = 1 \& X_{2i} = 1) \]

No Secondary Collection

\[ h_{12} = \text{compound error probability for Secondary Entry from Primary Collection} = P(\text{compound error} \mid T_{1i} = 1 \& X_{2i} = 1) \]
\[ h_{111} = \text{paired compound error probability} = P(\text{both compound errors} \mid T_{1i} = 1 \& X_{1i} = 1 \& X_{2i} = 1) \]

Comparison

\[ m_6 = \text{conditional probability of matched paired error records} = \text{Prob(match \mid both Primary and Secondary records have errors)} = P(\text{same bad value(s)} \mid X_{3i} = 1 \& X_{4i} = 1) \]
\[ c_6 = \text{conditional Comparison detection rate} = P(Y_i = 1 \mid X_{3i} = 1 \& W_i = 1) \]

Correction

\[ k = \text{conditional proper correction rate} = \text{Prob(proper correction \mid error detected & correction found)} \]
\[ 1 - k = \text{conditional probability of an improper update} \]
Model 6 Theorems

3.6.1 \( X_3 \sim B(N, e_{63}) \), errors after Primary Processing

\[
(6.1) \quad e_{63} = t_1(1 - e_1) + (1 - t_1)e_1 + h_{11}t_1e_1
\]

3.6.2 \( X_4 \sim B(N, e_4) \), errors after Secondary Processing

\[
(6.2) \quad e_{641} = t_2(1 - e_2) + (1 - t_2)e_2 + h_{22}t_2e_2 \quad \text{Redundant collection}
\]

\[
(6.3) \quad e_{642} = t_1(1 - e_2) + (1 - t_1)e_2 + h_{12}t_1e_2 \quad \text{No redundant collection}
\]

Lemma 3.6.1 With redundant collection, all \( X_{3i} \) and \( X_{4i} \) are independent

Lemma 3.6.2 With Primary collection only, then

\[
(6.4) \quad e_{611} = P(X_{3i} = 1 \& X_{4i} = 1)
= (1 - t_1)e_1e_2
+ t_1[(1 - e_1)(1 - e_2) + h_{11}e_1(1 - e_2) + h_{12}(1 - e_1)e_2 + h_{111}e_1e_2]
\]

\[
(6.5) \quad e_{610} = P(X_{3i} = 1 \& X_{4i} = 0)
= e_{63} - e_{611}
\]

3.6.3 \( W \sim B(N, w_6) \), non-matches from Comparison

\[
(6.6) \quad w_{61} = e_{63} + e_{641} - (1 + m_6)e_{63}e_{641} \quad \text{Redundant collection}
\]

\[
(6.7) \quad w_{62} = 1 - [(1 - t_1)(1 - e_1)(1 - e_2) + m_6e_{611}] \quad \text{No redundant collection}
\]

3.6.4 \( Y \sim B(N, y_6) \), updates required by Comparison

\[
(6.8) \quad y_{61} = c_6(e_{63} - m_6e_{63}e_{641}) \quad \text{Redundant collection}
\]

\[
(6.9) \quad y_{62} = c_6(e_{63} - m_6e_{611}) \quad \text{No redundant collection}
\]

3.6.5 \( Z \sim B(N, z_6) \), undetected errors in final dataset

\[
(6.10) \quad z_{61} = e_{63} - c_6k(e_{63} - m_6e_{63}e_{641}) \quad \text{Redundant collection}
\]

\[
(6.11) \quad z_{62} = e_{63} - c_6k(e_{63} - m_6e_{611}) \quad \text{No redundant collection}
\]
## Model 6 Probability Parameter Values

### Parameters
- $t = 0.01, 0.05, 0.10$  
  Data collection error rate: $t_1 = t_2 = t$
- $e = 0.01 - 0.10$  
  Entry error rate: $e_1 = e_2 = e$
- $c_6 = 0.95$  
  Conditional detection rate at Comparison
- $m_6 = 0.50$  
  Conditional probability of matched errors
- $h = 0.50$  
  Compound error rate: $h_{11} = h_{12} = h_{11} = h$
- $k = 1.00$  
  Conditional probability of proper correction

### Cost Difference

$$\text{DIFF}_{12}(T) = \text{NP}_{60} + \text{NP}_{63}(w_{61} - w_{62}) + \text{NP}_{64}(y_{61} - y_{62})$$

- $P_{60}$: average processing cost per record for data collection
- $P_{63}$: average processing cost per record for finding a correction
- $P_{64}$: average processing cost per record for updating

<table>
<thead>
<tr>
<th>Input Error Coll.</th>
<th>Entry $t$</th>
<th>Subsystem Differences $z_{61} - z_{62}$</th>
<th>$w_{61} - w_{62}$</th>
<th>$y_{61} - y_{62}$</th>
<th>Subsystem 1 Redundant Collection $w_{61}$ $y_{61}$ $z_{61}$</th>
<th>Subsystem 2 Primary Collection $w_{62}$ $y_{62}$ $z_{62}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.01</td>
<td>-0.0046</td>
<td>0.014</td>
<td>0.005</td>
<td>0.019 $0.001$ 0.025</td>
<td>0.014 $0.006$ 0.006</td>
</tr>
<tr>
<td>0.01</td>
<td>0.02</td>
<td>-0.0044</td>
<td>0.014</td>
<td>0.004</td>
<td>0.058 $0.028$ 0.002</td>
<td>0.044 $0.023$ 0.006</td>
</tr>
<tr>
<td>0.01</td>
<td>0.03</td>
<td>-0.0043</td>
<td>0.014</td>
<td>0.004</td>
<td>0.077 $0.037$ 0.003</td>
<td>0.063 $0.033$ 0.007</td>
</tr>
<tr>
<td>0.01</td>
<td>0.04</td>
<td>-0.0042</td>
<td>0.013</td>
<td>0.004</td>
<td>0.095 $0.046$ 0.004</td>
<td>0.082 $0.042$ 0.008</td>
</tr>
<tr>
<td>0.01</td>
<td>0.05</td>
<td>-0.0040</td>
<td>0.013</td>
<td>0.004</td>
<td>0.113 $0.055$ 0.005</td>
<td>0.101 $0.051$ 0.009</td>
</tr>
<tr>
<td>0.01</td>
<td>0.06</td>
<td>-0.0039</td>
<td>0.012</td>
<td>0.004</td>
<td>0.131 $0.063$ 0.006</td>
<td>0.119 $0.059$ 0.010</td>
</tr>
<tr>
<td>0.01</td>
<td>0.07</td>
<td>-0.0038</td>
<td>0.012</td>
<td>0.004</td>
<td>0.149 $0.072$ 0.007</td>
<td>0.137 $0.068$ 0.011</td>
</tr>
<tr>
<td>0.01</td>
<td>0.08</td>
<td>-0.0036</td>
<td>0.011</td>
<td>0.004</td>
<td>0.166 $0.081$ 0.008</td>
<td>0.154 $0.077$ 0.012</td>
</tr>
<tr>
<td>0.01</td>
<td>0.09</td>
<td>-0.0035</td>
<td>0.011</td>
<td>0.004</td>
<td>0.183 $0.089$ 0.010</td>
<td>0.172 $0.086$ 0.013</td>
</tr>
<tr>
<td>0.01</td>
<td>0.10</td>
<td>-0.0034</td>
<td>0.011</td>
<td>0.003</td>
<td>0.199 $0.097$ 0.011</td>
<td>0.189 $0.094$ 0.014</td>
</tr>
<tr>
<td>0.05</td>
<td>0.01</td>
<td>-0.0219</td>
<td>0.069</td>
<td>0.022</td>
<td>0.113 $0.055$ 0.005</td>
<td>0.044 $0.033$ 0.027</td>
</tr>
<tr>
<td>0.05</td>
<td>0.02</td>
<td>-0.0212</td>
<td>0.067</td>
<td>0.021</td>
<td>0.130 $0.063$ 0.006</td>
<td>0.063 $0.042$ 0.027</td>
</tr>
<tr>
<td>0.05</td>
<td>0.03</td>
<td>-0.0206</td>
<td>0.065</td>
<td>0.021</td>
<td>0.146 $0.071$ 0.007</td>
<td>0.081 $0.050$ 0.027</td>
</tr>
<tr>
<td>0.05</td>
<td>0.04</td>
<td>-0.0199</td>
<td>0.063</td>
<td>0.020</td>
<td>0.163 $0.079$ 0.008</td>
<td>0.100 $0.059$ 0.028</td>
</tr>
<tr>
<td>0.05</td>
<td>0.05</td>
<td>-0.0193</td>
<td>0.061</td>
<td>0.019</td>
<td>0.179 $0.087$ 0.009</td>
<td>0.118 $0.068$ 0.029</td>
</tr>
<tr>
<td>0.05</td>
<td>0.06</td>
<td>-0.0187</td>
<td>0.059</td>
<td>0.019</td>
<td>0.194 $0.095$ 0.011</td>
<td>0.135 $0.076$ 0.029</td>
</tr>
<tr>
<td>0.05</td>
<td>0.07</td>
<td>-0.0181</td>
<td>0.057</td>
<td>0.018</td>
<td>0.210 $0.103$ 0.012</td>
<td>0.153 $0.085$ 0.030</td>
</tr>
<tr>
<td>0.05</td>
<td>0.08</td>
<td>-0.0175</td>
<td>0.055</td>
<td>0.018</td>
<td>0.225 $0.110$ 0.014</td>
<td>0.170 $0.093$ 0.031</td>
</tr>
<tr>
<td>0.05</td>
<td>0.09</td>
<td>-0.0169</td>
<td>0.053</td>
<td>0.017</td>
<td>0.240 $0.118$ 0.015</td>
<td>0.187 $0.101$ 0.032</td>
</tr>
<tr>
<td>0.05</td>
<td>0.10</td>
<td>-0.0164</td>
<td>0.051</td>
<td>0.016</td>
<td>0.255 $0.126$ 0.017</td>
<td>0.203 $0.109$ 0.033</td>
</tr>
<tr>
<td>0.10</td>
<td>0.01</td>
<td>-0.0415</td>
<td>0.131</td>
<td>0.041</td>
<td>0.199 $0.097$ 0.011</td>
<td>0.068 $0.056$ 0.052</td>
</tr>
<tr>
<td>0.10</td>
<td>0.02</td>
<td>-0.0402</td>
<td>0.127</td>
<td>0.040</td>
<td>0.213 $0.105$ 0.012</td>
<td>0.086 $0.064$ 0.053</td>
</tr>
<tr>
<td>0.10</td>
<td>0.03</td>
<td>-0.0390</td>
<td>0.123</td>
<td>0.039</td>
<td>0.227 $0.112$ 0.014</td>
<td>0.104 $0.073$ 0.053</td>
</tr>
<tr>
<td>0.10</td>
<td>0.04</td>
<td>-0.0378</td>
<td>0.119</td>
<td>0.038</td>
<td>0.241 $0.119$ 0.015</td>
<td>0.122 $0.081$ 0.053</td>
</tr>
<tr>
<td>0.10</td>
<td>0.05</td>
<td>-0.0366</td>
<td>0.115</td>
<td>0.037</td>
<td>0.255 $0.126$ 0.017</td>
<td>0.139 $0.089$ 0.053</td>
</tr>
<tr>
<td>0.10</td>
<td>0.06</td>
<td>-0.0354</td>
<td>0.112</td>
<td>0.035</td>
<td>0.268 $0.133$ 0.018</td>
<td>0.156 $0.097$ 0.054</td>
</tr>
<tr>
<td>0.10</td>
<td>0.07</td>
<td>-0.0343</td>
<td>0.108</td>
<td>0.034</td>
<td>0.281 $0.139$ 0.020</td>
<td>0.173 $0.105$ 0.054</td>
</tr>
<tr>
<td>0.10</td>
<td>0.08</td>
<td>-0.0332</td>
<td>0.104</td>
<td>0.033</td>
<td>0.294 $0.146$ 0.022</td>
<td>0.189 $0.113$ 0.055</td>
</tr>
<tr>
<td>0.10</td>
<td>0.09</td>
<td>-0.0321</td>
<td>0.101</td>
<td>0.032</td>
<td>0.306 $0.153$ 0.024</td>
<td>0.205 $0.121$ 0.056</td>
</tr>
<tr>
<td>0.10</td>
<td>0.10</td>
<td>-0.0310</td>
<td>0.097</td>
<td>0.031</td>
<td>0.319 $0.159$ 0.026</td>
<td>0.221 $0.128$ 0.057</td>
</tr>
</tbody>
</table>
APPENDIX B

STUDY SUMMARIES

BAC Study
An experiment at HSRC to compare Manual Proofreading with Redundancy. Records consisting of three numeric fields (Driver's License Number, Birth Date, BAC level) were divided into two groups of about 600 records each. Undergraduate students did data entry and proofreading. Microcomputers were used for data entry and a mainframe for Comparison and updating.

Bike Study
An experiment at HSRC with intelligent data entry and Redundancy. Information on 247 bicycle accidents was taken from a computer listing. All 31 fields were short numeric fields and most had only two values (0, 1). All processing was done on a microcomputer. Careful proofreading was also done to find any undetected errors.

Glaucoma Study
An experiment to investigate Redundancy and VVE, both interactive and batch. There were 393 forms divided into two batches with 337 records in the first batch. The input forms were very messy and had a total of 23 fields, which included name fields (last, first and middle), simple categorical information related to hypertension, two measurements of blood pressure (3-digit), and two measurements of ocular tension (2-digit). Some coding was required during data entry. Data entry was done by a consultant using microcomputer; other processing done using a mainframe.
NC Belt Study
A study at HSRC that used professional data entry services. Simple categorical information about front seat passengers of motor vehicles (seat belt usage, race) have been collected on several occasions. The first batch had 18,432 records. Two different entry services have been used.

Nevada Study
An experiment at HSRC using Redundancy and batch VVE. Information taken from survey forms with thirteen categorical questions produced 937 records. Undergraduate students did data entry using a microcomputer. A batch VVE program was written in SAS on a mainframe. Updating was done using several methods, both microcomputer and mainframe.

Registered Drivers Study
A study at HSRC that involved Manual Proofreading. A total of 1713 records consisting of six fields (two ID, four data) were proofread two ways. The data fields were counts that could be checked individually or by checking their sum. Data entry was done using a microcomputer.

Southwest Study
A small study done at HSRC with 325 records, consisting of eleven categorical fields and an eight-digit case number. Primary and Secondary Entry were done by undergraduate students and a research associate using a microcomputer. Comparison was done using a mainframe computer by modifying an existing program based on SAS %COMPARE macro.
Truck Correlation Study

A limited experiment at HSRC using Redundancy. The data consisted of identification information (name, address, SSN#, DL#) and a few categorical fields taken from the top half of a form. The information was collected from four companies; the number of forms ranged from 150 to 350. Undergraduate students did data entry using a microcomputer. Comparison and updating were done using a mainframe.
<table>
<thead>
<tr>
<th>Name</th>
<th>BAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subsystems</td>
<td>Manual Proofread, Redundancy</td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
</tr>
<tr>
<td>Experiment</td>
<td>Data divided into two groups, processed differently, see Section 5.2</td>
</tr>
<tr>
<td>Data description</td>
<td>Computer listing with handwritten info added</td>
</tr>
<tr>
<td></td>
<td>Entry of selected lines and fields only</td>
</tr>
<tr>
<td></td>
<td>Blood Alcohol Level with identifying information</td>
</tr>
<tr>
<td>Records</td>
<td>612 for Manual Proofread subsystem</td>
</tr>
<tr>
<td></td>
<td>614 for Redundancy subsystem</td>
</tr>
<tr>
<td>Fields</td>
<td>ID</td>
</tr>
<tr>
<td></td>
<td>Driver's License number, 7 digits</td>
</tr>
<tr>
<td></td>
<td>Date, DDMMYY</td>
</tr>
<tr>
<td></td>
<td>BAC, 2 digits, some rounding</td>
</tr>
<tr>
<td>RDM elements</td>
<td>PDMS on Apple //e, undergraduate students</td>
</tr>
<tr>
<td></td>
<td>Undergraduate students</td>
</tr>
<tr>
<td></td>
<td>Simple complete record comparison</td>
</tr>
<tr>
<td></td>
<td>Mainframe (TSO QED)</td>
</tr>
<tr>
<td></td>
<td>Mainframe processing after entry</td>
</tr>
<tr>
<td>Cost availability</td>
<td>Detailed records kept of time spent and computer cost</td>
</tr>
<tr>
<td>Error rates</td>
<td>11 / 612 = 0.018 entry error rate</td>
</tr>
<tr>
<td>Proofread</td>
<td>1 record with multiple errors</td>
</tr>
<tr>
<td></td>
<td>15 / 614 = 0.024</td>
</tr>
<tr>
<td></td>
<td>1 record with multiple errors</td>
</tr>
<tr>
<td></td>
<td>11 / 614 = 0.018</td>
</tr>
<tr>
<td>Detection rates</td>
<td>4 detected errors</td>
</tr>
<tr>
<td>Proofread</td>
<td>7 undetected errors</td>
</tr>
<tr>
<td></td>
<td>7 / 11 = 0.636 cond. detection rate</td>
</tr>
<tr>
<td>Comparison</td>
<td>15 detected errors</td>
</tr>
<tr>
<td></td>
<td>0 undetected errors</td>
</tr>
<tr>
<td></td>
<td>2 records with errors found but not corrected</td>
</tr>
<tr>
<td>Name</td>
<td>Bike</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Subsystems</td>
<td>Redundancy, Manual Proofread</td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
</tr>
<tr>
<td>Experiment</td>
<td>Special proofreading after regular processing, see Section 5.4</td>
</tr>
<tr>
<td>Data description</td>
<td>Computer listing from SAS PROC PRINT</td>
</tr>
<tr>
<td>Form style</td>
<td>Simple information about bicycle accidents</td>
</tr>
<tr>
<td>Information</td>
<td></td>
</tr>
<tr>
<td>Records</td>
<td>247</td>
</tr>
<tr>
<td>Fields</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Sequence number, 3 digits</td>
</tr>
<tr>
<td>Categorical</td>
<td>29 fields, mostly (0,1) values</td>
</tr>
<tr>
<td>Numeric</td>
<td>Age, Hour, Hospital (1 – 11)</td>
</tr>
<tr>
<td>RDM elements</td>
<td></td>
</tr>
<tr>
<td>Data entry</td>
<td>PC-File on IBM PC AT, programmer trainee</td>
</tr>
<tr>
<td>Intelligent entry</td>
<td>Masking in PC-File</td>
</tr>
<tr>
<td>Proofreading</td>
<td>For experimental purposes</td>
</tr>
<tr>
<td>Comparison</td>
<td>BASS COMPARE procedure</td>
</tr>
<tr>
<td>Update</td>
<td>Microcomputer (PC-File)</td>
</tr>
<tr>
<td>Computer use</td>
<td>Microcomputer only</td>
</tr>
<tr>
<td>Cost availability</td>
<td>Records kept on time spent</td>
</tr>
<tr>
<td>Error rates</td>
<td></td>
</tr>
<tr>
<td>Primary Entry</td>
<td>$16 / 247 = 0.0648$</td>
</tr>
<tr>
<td></td>
<td>7 multiple errors, 2 ID errors</td>
</tr>
<tr>
<td>Secondary Entry</td>
<td>$15 / 247 = 0.0607$</td>
</tr>
<tr>
<td></td>
<td>11 multiple errors</td>
</tr>
<tr>
<td>Detection rates</td>
<td></td>
</tr>
<tr>
<td>Comparison</td>
<td>No matched errors</td>
</tr>
<tr>
<td>Manual Proofread</td>
<td>0.87 or 0.93</td>
</tr>
<tr>
<td>Proper correction rate</td>
<td>$14 / 15 = 0.93$</td>
</tr>
<tr>
<td>Name</td>
<td>Glaucoma</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>Subsystems</td>
<td>Redundancy, VVE</td>
</tr>
<tr>
<td>Environment</td>
<td>Consultant</td>
</tr>
<tr>
<td>Experiment</td>
<td>Multiple entry, intelligent data entry, batch VVE</td>
</tr>
<tr>
<td>Data description</td>
<td>2-page forms, very messy</td>
</tr>
<tr>
<td></td>
<td>Simple hypertension, ocular tension information</td>
</tr>
<tr>
<td>Records</td>
<td>337 first batch, 56 second batch</td>
</tr>
<tr>
<td>Fields</td>
<td>Sequence number</td>
</tr>
<tr>
<td></td>
<td>6 alpha value fields (Race, sex, smoking, etc.)</td>
</tr>
<tr>
<td></td>
<td>Some coding needed (Hypertension history)</td>
</tr>
<tr>
<td>ID</td>
<td>BP, Ocular tension</td>
</tr>
<tr>
<td>Categorical</td>
<td>Date, Birthdate</td>
</tr>
<tr>
<td>Numeric</td>
<td>Name</td>
</tr>
<tr>
<td>Alpha</td>
<td></td>
</tr>
<tr>
<td>RDM elements</td>
<td>PC-File on IBM PC, with and without VVE</td>
</tr>
<tr>
<td>Data entry</td>
<td>SAS program</td>
</tr>
<tr>
<td>VVE batch</td>
<td>SAS Macro %COMPARE</td>
</tr>
<tr>
<td>Comparison</td>
<td>Mainframe (TSO QED)</td>
</tr>
<tr>
<td>Update</td>
<td>Mainframe after entry</td>
</tr>
<tr>
<td>Computer use</td>
<td></td>
</tr>
<tr>
<td>Cost availability</td>
<td>Detailed records kept of time spent and computer cost</td>
</tr>
<tr>
<td></td>
<td>Overall cost estimates</td>
</tr>
</tbody>
</table>

Error rates

- **Primary Entry**
  - \(\frac{66}{337} = 0.196\), detected only
  - Multiple errors, 17% \((11/66)\) detected by Comparison
- **Secondary Entry**
  - \(\frac{57}{337} = 0.169\), detected only

Detection rates

- **Valid value edit**
  - Intelligent data entry (PC-Fie masking)
    - Primary: \(\frac{88}{337}\)
    - Duplicate: \(\frac{40}{337}\)
  - Primary VVE (masking & edit) detected: 97
    - \(\frac{97}{(97 + 66)} = 0.595\)
- **Comparison**
  - 24 non-matches in which both errors
<table>
<thead>
<tr>
<th>Name</th>
<th>NC Belt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environment</td>
<td>HSRC, professional collection, professional entry</td>
</tr>
<tr>
<td>Experiment</td>
<td>Not controlled: professional data entry service</td>
</tr>
<tr>
<td>Data description</td>
<td>Designed data collection form</td>
</tr>
<tr>
<td></td>
<td>20 lines per form, one record per line</td>
</tr>
<tr>
<td>Information</td>
<td>Seatbelt use and race for front seat occupants</td>
</tr>
<tr>
<td>Records</td>
<td>18432 first batch</td>
</tr>
<tr>
<td>Fields</td>
<td>Site-Observer-Page number</td>
</tr>
<tr>
<td></td>
<td>Vehicle type for non-cars</td>
</tr>
<tr>
<td></td>
<td>Belt use, Race for 3 seating positions</td>
</tr>
<tr>
<td>RDM elements</td>
<td>Professional data entry services</td>
</tr>
<tr>
<td>Data entry</td>
<td>Durham service</td>
</tr>
<tr>
<td>Interactive VVE</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Computer use</td>
<td></td>
</tr>
<tr>
<td>Cost availability</td>
<td>Raleigh</td>
</tr>
<tr>
<td></td>
<td>$1 per page (20 records) for entry</td>
</tr>
<tr>
<td></td>
<td>$0.33 per page for dependent verification</td>
</tr>
<tr>
<td></td>
<td>Durham</td>
</tr>
<tr>
<td></td>
<td>$0.02 per record for entry</td>
</tr>
<tr>
<td></td>
<td>$0.02 per record for dependent verification</td>
</tr>
<tr>
<td>Error rates</td>
<td>89 / 18432 = 0.0048 without dependent verification</td>
</tr>
<tr>
<td>Name</td>
<td>Nevada</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
</tr>
<tr>
<td>Subsystems</td>
<td>Redundancy, VVE</td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
</tr>
<tr>
<td>Experiment</td>
<td>Error type analysis, see Section 5.3</td>
</tr>
<tr>
<td>Data description</td>
<td>Multiple choice survey, single legal-sized page (see p. S-9)</td>
</tr>
<tr>
<td></td>
<td>Questions related to drunk driving in Nevada</td>
</tr>
<tr>
<td>Records</td>
<td>937</td>
</tr>
<tr>
<td>Fields</td>
<td>Sequence number</td>
</tr>
<tr>
<td>ID</td>
<td>18 fields (letter values a – g)</td>
</tr>
<tr>
<td>Categorical</td>
<td>Chance Of Arrest, 2-digit</td>
</tr>
<tr>
<td>Numeric</td>
<td></td>
</tr>
<tr>
<td>RDM elements</td>
<td>PDMS on Apple //e, undergraduate students</td>
</tr>
<tr>
<td>Data entry</td>
<td>SAS program</td>
</tr>
<tr>
<td>VVE batch</td>
<td>BASS COMPARE procedure</td>
</tr>
<tr>
<td>Comparison</td>
<td>Mainframe (TSO QED), microcomputer (PC-File)</td>
</tr>
<tr>
<td>Update</td>
<td>Mostly microcomputer, some mainframe</td>
</tr>
<tr>
<td>Computer use</td>
<td></td>
</tr>
<tr>
<td>Cost availability</td>
<td>Records kept of time spent and computer cost</td>
</tr>
<tr>
<td>Results</td>
<td>See p. S-10</td>
</tr>
<tr>
<td>Error rates</td>
<td>0.081</td>
</tr>
<tr>
<td>Primary Entry</td>
<td>0.081</td>
</tr>
<tr>
<td>Secondary Entry</td>
<td></td>
</tr>
<tr>
<td>Detection rates</td>
<td>0.303, 0.366</td>
</tr>
<tr>
<td>VVE batch</td>
<td>matched errors: $1/8 = 0.125$ or $1/3 = 0.333$</td>
</tr>
<tr>
<td>Comparison</td>
<td>detection: at least 0.97</td>
</tr>
<tr>
<td>Proper correction rate</td>
<td>at least 0.96</td>
</tr>
</tbody>
</table>
Nevada Department of Motor Vehicles – Driver's License Division
Survey on Highway Safety Issues (2)

The Nevada Department of Motor Vehicles requests your help in providing information about highway safety issues. Your answers to the following questions will be strictly anonymous and will be used only for statistical purposes to help plan future safety programs.

1. Your sex? (CIRCLE ONE)  Male  Female

2. Your age? (CIRCLE ONE)  a. 16-19  b. 20-29  c. 30-49  d. 50-65  e. 65 & over

3. Why are you at the driver's license office? (CIRCLE ONE)
   a. To get first license  c. To have license reinstated
   b. To renew currently valid license  d. I. D. only
   e. Other (PLEASE SPECIFY):

4. If you are stopped for drunk driving and fail or refuse to take the breath alcohol test, is it possible that you will have to give your license to the police for suspension by the Dept. of Motor Vehicles before going to court on a drunk driving charge? (CIRCLE ONE)
   a. Yes  b. No

5. Suppose you drive after drinking enough to violate Nevada's drunk driving law, what are your chances of being arrested by the police? (CIRCLE ONE)
   a. 0%  b. 1-10%  c. 20-39%  d. 40-59%  e. 60-79%  f. 80-99%  g. 100%

6. What percent of drivers who are stopped for drunk driving and fail or refuse to take a breath alcohol test must give their licenses to the police for suspension by the Dept. of Motor Vehicles before going to court for a drunk driving charge? (CIRCLE ONE)
   a. 0%  b. 1-19%  c. 20-39%  d. 40-59%  e. 60-79%  f. 80-99%  g. 100%

7. How strongly does this chance of turning over your driver's license to the police for suspension by the Dept. of Motor Vehicles before you go to court influence your decision not to drive after drinking enough to violate Nevada's drunk driving law? (CIRCLE ONE)

8. If someone is stopped for drunk driving for the first time and must give his or her license to the police for suspension by the Department of Motor Vehicles before going to court, for how long is the person's license suspended? (If you are not sure, please guess the minimum months' suspension.)
   __________ Months

9. In general, about how often do you drink beer, wine or liquor? (CIRCLE ONE)
   a. Every day  b. Several times a week  c. Once a week  d. Once a month  e. Less than once a month  f. Never

10. How often do you drink alcoholic beverages and then drive within 3 hours? (CIRCLE ONE)
    a. Every day  b. Several times a week  c. Once a week  d. Once a month  e. Less than once a month  f. Never

11. A. Within the last 3 months, how often do you think you may have driven after drinking enough to violate Nevada's drunk driving law? (CIRCLE ONE)
    a. Every day  b. Several times a week  c. Once a week  d. Once a month  e. Less than once a month  f. Never

    B. Compared with three months ago, has your rate of driving after drinking:
       (CIRCLE ONE)
       a. Increased  b. Decreased  c. Stayed the same

    C. If your rate of driving while drinking has changed, please indicate why.
       (CIRCLE ALL THAT APPLY)
       a. Increased enforcement  b. Decreased enforcement  c. Greater chance of being convicted
       d. Lesser chance of being convicted  e. Stronger penalties  f. Weaker penalties  g. Other (PLEASE EXPLAIN)
Nevada Study Results

<table>
<thead>
<tr>
<th>Subsystem 2</th>
<th>Subsystem 3</th>
<th>Subsystem 5</th>
<th>Subsystem 6</th>
<th>Secondary Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>No VVE, Redundancy only</td>
<td>Primary VVE only</td>
<td>Primary VVE, Redundancy (Secondary Entry, Comparison)</td>
<td>Primary and Secondary VVE, Redundancy</td>
<td>Corrections found and updated for additional information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>e₁</th>
<th>e₂</th>
<th>d₁₁₁</th>
<th>d₂₁₁</th>
<th>m</th>
<th>c</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.081</td>
<td>0.088</td>
<td>0.342</td>
<td>0.366</td>
<td>0.125</td>
<td>0.987</td>
<td>0.973</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>X₁</th>
<th>X₂</th>
<th>V₁</th>
<th>V₂</th>
<th>X₃</th>
<th>X₄</th>
<th>W</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>76</td>
<td>76</td>
<td>76</td>
<td>76</td>
<td>76</td>
<td>76</td>
<td>153</td>
<td>75</td>
<td>4</td>
</tr>
<tr>
<td>82</td>
<td>82</td>
<td>26</td>
<td>26</td>
<td>50</td>
<td>82</td>
<td>132</td>
<td>52</td>
<td>50</td>
</tr>
</tbody>
</table>

| z | 0.0043 | 0.0534 | 0.0043 | 0.0043 | 0.0043 |

Undetected errors

<table>
<thead>
<tr>
<th>Matched errors</th>
<th>Primary</th>
<th>Secondary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Improper update</th>
<th>Primary</th>
<th>Secondary</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Not found</th>
<th>Primary</th>
<th>Secondary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td><strong>Registered Drivers</strong></td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>------------------------</td>
<td></td>
</tr>
<tr>
<td>Subsystem</td>
<td>Manual Proofread</td>
<td></td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
<td></td>
</tr>
</tbody>
</table>
| Experiment | Proofread using two different methods:  
    1) standard  
    2) sum |
| Data description | Computer listing  
    Numbers of registered drivers by Race/Sex category |
| Records    | 1713                   |
| Fields     |                        |
| ID         | Calendar year, YYYY    
    Birth year, YYYY |
| Numeric    | 4 counts, 5-digit      |
| RDM elements | PC-File on IBM PC, undergraduate students  
    Check for numeric type only  
    Research assistant  
    Microcomputer only |
| Cost availability | No records kept |
| Error rate |                        |
| Data entry | 84 / 1713 = 0.05       |
| Detection rate | Manual Proofread  
    46 / 84 = 0.55 using standard (checking all fields) |
<table>
<thead>
<tr>
<th>Name</th>
<th>Southwest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subsystem</td>
<td>Redundancy</td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
</tr>
<tr>
<td>Experiment</td>
<td>Not an experiment</td>
</tr>
<tr>
<td>Data description</td>
<td>Designed form, one record per line</td>
</tr>
<tr>
<td>Form style</td>
<td>Coded from police motor vehicle accident report form</td>
</tr>
<tr>
<td>Information</td>
<td></td>
</tr>
<tr>
<td>Records</td>
<td>325</td>
</tr>
<tr>
<td>Fields</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Case number, 8 digits</td>
</tr>
<tr>
<td>Categorical</td>
<td>13 number value fields, mostly 1-digit</td>
</tr>
<tr>
<td>Alpha</td>
<td>Comment (25 character max), for certain records only</td>
</tr>
<tr>
<td>RDM elements</td>
<td></td>
</tr>
<tr>
<td>Data entry</td>
<td>PC-File on IBM PC XT, undergraduate students</td>
</tr>
<tr>
<td>Comparison</td>
<td>SAS Macro %COMPARE</td>
</tr>
<tr>
<td>Update</td>
<td>Mainframe (TSO QED)</td>
</tr>
<tr>
<td>Computer use</td>
<td>Mainframe after entry</td>
</tr>
<tr>
<td>Cost availability</td>
<td>No records kept</td>
</tr>
<tr>
<td>Error rates</td>
<td></td>
</tr>
<tr>
<td>Data entry</td>
<td>15 – 20 records with error(s) in data fields</td>
</tr>
<tr>
<td></td>
<td>15 / 325 = 0.046, 20 / 325 = 0.062</td>
</tr>
<tr>
<td></td>
<td>5 or 6 errors in ID field, some skipped records</td>
</tr>
<tr>
<td></td>
<td>2 records with multiple errors</td>
</tr>
<tr>
<td>Name</td>
<td>Truck Correlation</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Subsystem</td>
<td>Redundancy</td>
</tr>
<tr>
<td>Environment</td>
<td>HSRC</td>
</tr>
<tr>
<td>Experiment</td>
<td>Records kept during regular processing</td>
</tr>
<tr>
<td>Data description</td>
<td>Designed form, one record per form</td>
</tr>
<tr>
<td>Information</td>
<td>Data on truck drivers</td>
</tr>
<tr>
<td></td>
<td>Recorded by researchers from truck company records</td>
</tr>
<tr>
<td>Records</td>
<td>344, Company A</td>
</tr>
<tr>
<td></td>
<td>319, Company B</td>
</tr>
<tr>
<td></td>
<td>204, Company C</td>
</tr>
<tr>
<td></td>
<td>153, Company D</td>
</tr>
<tr>
<td>Fields</td>
<td>Sequence number, 4 digits</td>
</tr>
<tr>
<td>ID</td>
<td>Birthday, DL number, SSN</td>
</tr>
<tr>
<td>Numeric</td>
<td>Name, address</td>
</tr>
<tr>
<td>Alpha</td>
<td>Some records kept of time spent and computer cost</td>
</tr>
<tr>
<td>RDM elements</td>
<td>Overall cost estimates</td>
</tr>
<tr>
<td>Data entry</td>
<td>PDMS on Apple //e, undergraduate students</td>
</tr>
<tr>
<td>Comparison</td>
<td>SAS Macro %COMPARE</td>
</tr>
<tr>
<td>Update</td>
<td>Mainframe (TSO QED)</td>
</tr>
<tr>
<td>Computer use</td>
<td>Mainframe after entry</td>
</tr>
<tr>
<td>Cost availability</td>
<td>Some records kept of time spent and computer cost</td>
</tr>
<tr>
<td>Error rates</td>
<td>Overall cost estimates</td>
</tr>
<tr>
<td>Primary Entry</td>
<td>5 / 344 = 0.015 Sequence, DL# only</td>
</tr>
<tr>
<td></td>
<td>24 / 344 = 0.070</td>
</tr>
<tr>
<td></td>
<td>32 / 319 = 0.100</td>
</tr>
<tr>
<td></td>
<td>19 / 204 = 0.093</td>
</tr>
<tr>
<td></td>
<td>8 / 153 = 0.052</td>
</tr>
<tr>
<td>Secondary Entry</td>
<td>2 / 344 = 0.006 Sequence, DL# only</td>
</tr>
<tr>
<td></td>
<td>42 / 344 = 0.122</td>
</tr>
<tr>
<td></td>
<td>52 / 319 = 0.163</td>
</tr>
<tr>
<td></td>
<td>27 / 204 = 0.132</td>
</tr>
<tr>
<td></td>
<td>9 / 153 = 0.059</td>
</tr>
<tr>
<td>Detection Comparison</td>
<td>Non-matches: W</td>
</tr>
<tr>
<td></td>
<td>7 of 344 Sequence, DL# only</td>
</tr>
<tr>
<td></td>
<td>64 of 344</td>
</tr>
<tr>
<td></td>
<td>77 of 319</td>
</tr>
<tr>
<td></td>
<td>45 of 204</td>
</tr>
<tr>
<td></td>
<td>16 of 153</td>
</tr>
</tbody>
</table>