# AN APPLICATION OF THE FELLEGI-SUNTER MODEL OF RECORD LINKAGE TO THE 1990 U.S. DECENNIAL CENSUS

William E. Winkler and Yves Thibaudeau
U.S. Bureau of the Census

## ABSTRACT

This paper describes a methodology for computer matching the Post Enumeration Survey with the Census. Computer matching is the first stage of a process for producing adjusted Census counts. All crucial matching parameters are computed solely using characteristics of the files being matched. No a priori knowledge of truth of matches is assumed. No previously created lookup tables are needed. The methods are illustrated with numerical results using files from the 1988 Dress Rehearsal Census for which the truth of matches is known.

Key words and phrases. EM Algorithm; String Comparator Metric; LP Algorithm; Decision Rule; Error Rate.

## 1. INTRODUCTION

This paper describes a particular application of the Fellegi-Sunter (1969) model of record linkage. New computational methods are used for computer matching the Post Enumeration Survey (PES) with the Census. The PES is used to produce adjusted Census counts. Computer matching is the first stage of PES processing.

All crucial matching parameters associated with comparisons of individual fields are computed automatically. The parameters are generally based on characteristics of the files being matched. No a priori knowledge of truth of matches is assumed. Lookup tables that account for the relative frequency of occurrence of different strings are computed using the files being matched. The paper is divided into a number of sections.

The second section consists of five parts. The first part gives background on the Fellegi-Sunter model. The second part describes PES and Census files from the 1988 Dress Rehearsal Census and overall matching procedures. Truth and falsehood of matches is known for the Dress Rehearsal files.

The third part provides details of a modified Expectation-Maximization (EM) Algorithm for estimating probability distributions used in a crucial likelihood ratio (see e.g., Winkler 1988, 1989a; Thibaudeau 1989). In the fourth part, new computational methods for automatically creating frequency tables accounting for the relative distinguishing power of strings such as 'Smith' and 'Zabrinsky' are given. The methods are a special case of Winkler (1989b).

The fifth part describes new string comparator metrics that allow comparison of strings that do not agree on a character-by-character basis. The metrics generalize Damerau-Levenstein and Jaro metrics (see e.g., Winkler 1985, 1989c, 1990b). Methods for modeling how the metric adjusts matching weights between pure agreement and pure disagreement are covered.

A new linear sum assignment algorithm that forces one-to-one assignments is described in the sixth part. It is substantially faster than the algorithm of Burkard and Derigs used by Jaro (1989).

Results are presented in the third section. Two commonly used matching methods are compared with the method used in the 1988 computer matching system and the enhanced methods adopted for 1990.

The fourth section consists of a six part discussion. The first describes how matching accuracy may fall when files are of relatively lower quality than those available in the 1990 Census and PES. In the second part, we cover how the string comparator metrics and weight adjustment methods fit in with the general Fellegi-Sunter model. The third part presents general limitations of the parameters produced via the EM Algorithm.

In the fourth part of the fourth section, we explain why the specific adjustments of the frequency-table building procedures are suitable in the applications. The fifth part covers how the linear sum assignment procedure improves matching efficacy and why decision rules using it are still optimal in the sense given by Fellegi and Sunter (1969, Theorem). In the sixth part, we discuss different ways of determining cutoff weights.

The fifth section consists of a summary, conclusions, and future work.

## 2. BACKGROUND AND DESCRIPTION OF METHODS

### 2.1. Fellegi-Sunter Model of Record Linkage

The Fellegi-Sunter Model uses a decision-theoretic approach establishing the validity of principles first used in practice by Newcombe (Newcombe et al. 1959). To give an overview, we describe the model in terms of ordered pairs in a product space. The description closely follows Fellegi and Sunter (1969, pp. 1184-1187).

There are two populations **A** and **B** whose elements will be denoted by a and b. We assume that some elements are common to **A** and **B**.
Consequently the set of ordered pairs

$$\mathbf{A}\mathrm{X}\mathbf{B} = \{(a,b): a\in\mathbf{A}, b\in\mathbf{B}\}$$

is the union of two disjoint sets of matches

$$M = \{(a,b): a=b, a\in\mathbf{A}, b\in\mathbf{B}\}$$

and nonmatches

$$U = \{(a,b): a\neq b, a\in\mathbf{A}, b\in\mathbf{B}\}.$$

The records corresponding to members of **A** and **B** are denoted by $\alpha(a)$ and $ß(b)$, respectively. The comparison vector $\gamma$ associated with the records is defined by:

$$\gamma[\alpha(a),ß(b)] \equiv \{\gamma^1[\alpha(a),ß(b)],\gamma^2[\alpha(a),ß(b)],\cdots,\gamma^K[\alpha(a),ß(b)]\}.$$

Each of the $\gamma^i$, $i = 1, \cdots, K$, represents a specific comparison. For instance, $\gamma^1$ could represent agreement/disagreement on sex. $\gamma^2$ could represent the comparison that two surnames agree and take a specific value or that they disagree.

Where confusion does not arise, the function $\gamma$ on **AXB** will be denoted by $\gamma(\alpha,ß)$, $\gamma(a,b)$, or $\gamma$. The set of all possible realizations of $\gamma$ is denoted by $\Gamma$.

The conditional probability of $\gamma(a,b)$ if $(a,b)\epsilon M$ is given by

$$m(\gamma) \equiv P\{\gamma[\alpha(a),ß(b)]|(a,b)\epsilon M\}$$

$$= \sum_{(a,b)\epsilon M} P\{\gamma[\alpha(a),ß(b)]\}\cdot P[(a,b)|M].$$

Similarly we denote the conditional probability of $\gamma$ if $(a,b)\epsilon U$ by $u(\gamma)$.

We observe a vector of information $\gamma(a,b)$ associated with pair $(a,b)$ and wish to designate a pair as a link (denote the decision by $A_1$), a possible link (decision $A_2$), or a nonlink (decision $A_3$). A <u>linkage rule</u> L is defined a mapping from $\Gamma$, the comparison space, onto a set of random decision functions $D = \{d(\gamma)\}$ where

$$d(\gamma) = \{P(A_1|\gamma),P(A_2|\gamma),P(A_3|\gamma)\};\ \gamma\epsilon\Gamma$$

and

$$\sum_{i=1}^{3} P(A_i|\gamma) = 1.$$

There are two types of error associated with a linkage rule. A <u>Type I error</u> occurs if an unmatched comparison is erroneously linked. It has probability

$$P(A_1|U) = \sum_{\gamma\epsilon\Gamma} u(\gamma)\cdot P(A_1|\gamma)$$

A <u>Type II error</u> occurs if a matched comparison is erroneously not linked. It has probability

$$P(A_3|U) = \sum_{\gamma\epsilon\Gamma} m(\gamma)\cdot P(A_3|\gamma)$$

Fellegi and Sunter (1969) define a linkage rule $L_0$, with associated decisions $A_1$, $A_2$, and $A_3$, that is optimal in the following sense:

**Theorem** (Fellegi-Sunter 1969). Let $L'$ be a linkage rule with associated decisions $A_1'$, $A_2'$, and $A_3'$ such that it has the same error probabilities $P(A_3'|M) = P(A_3|M)$ and $P(A_1'|U) = P(A_1|U)$ as $L_0$. Then $L_0$ is optimal in that $P(A_2|U) \leq P(A_2'|U)$ and $P(A_2|M) \leq P(A_2'|M)$.

In other words, if $L'$ is any competitor of $L_0$ having the same Type I and Type II error rates (which are both conditional probabilities), then the conditional probabilities (either on set U or M) of not making a decision under rule $L'$ are always greater than under $L_0$.

To describe rule $L_0$, we need the following likelihood ratio

$$R \equiv R[\gamma(a,b)] = m(\gamma)/u(\gamma). \tag{2.1}$$

We observe that, if $\gamma$ represents a comparison of K fields, then there are at least $2^K$ probabilities of form $m(\gamma)$. If $\gamma$ represents agreements of K fields, we would expect this to occur more often for matches M than for nonmatches U. The ratio R would then be large. Alternatively, if $\gamma$ consists of disagreements, the ratio R would be small.

If the numerator is positive and the denominator is zero in (2.1), we assign an arbitrary very large number to the ratio. The Fellegi-Sunter linkage rule $L_0$ takes the form:

If $R >$ UPPER, then denote (a,b) as a link.

If LOWER $\leq R \leq$ UPPER, then denote (a,b) as a possible link.

If $R <$ LOWER, then denote (a,b) as a nonlink.

The cutoffs LOWER and UPPER are determined by the desired error rate bounds.

The Fellegi-Sunter linkage rule is actually optimal with respect to any set Q of ordered pairs in **A**X**B** if we define error probabilities $P_Q$ and a linkage rule $L_Q$ conditional on Q. Thus, it may be possible to define subsets of **A**X**B** on which we make use of differing amounts and types of available information.

For instance, if we have a set of pairs in which telephone number is present, we might use telephone number and a few characters from the name to designate links. With other pairs, we may additionally have to utilize information from the street address and the city name.

Sets of ordered pairs Q on which the Fellegi-Sunter linkage rule is applied are often obtained by blocking criteria. Blocking criteria are sort keys that are used to reduce the number of pairs that are considered. Rather than consider all pairs in **A**X**B**, we might only consider pairs that agree on the first three digits of the ZIP code or on a suitable abbreviation of surname.

### 2.2. Empirical Data Bases and Description of Matching Procedures

Files for three geographic regions, (1) St Louis, MO, (2) Columbia, MO and vicinity, and (3) rural WA, were computer matched during the 1988 Dress Rehearsal Census and Post Enumeration Survey. The PES is a reenumeration of all individuals in all households in a sample of Census subregions. Each subregion, referred to as a block cluster, may be a Census block, a subsample of a Census block, or a group of Census blocks. The Census file consists of the individuals in the Census for the same sampled block clusters and for individuals in blocks or geographic subregions surrounding the PES blocks.

On the first computer matching pass, individuals in the PES are matched only with those individuals in the Census that are in the same block cluster. As almost all true matches that are ultimately found are identified during the first computer matching pass and associated clerical operations, the results of this paper only consider the first pass. Subsequent matching passes against the entire Census file (which is typically 25 to 30 times the size of the PES) are primarily intended to identify individuals who have been counted in the wrong location (i.e., near-by blocks). File sizes for within-block matching are given in

Table 1. File Sizes from the 1988 Dress Rehearsal Census Nonmovers for Within Block Cluster Matching

| City | Census | PES |
|---|---|---|
| St Louis | 15048 | 12072 |
| Columbia | 9794 | 7649 |
| Washington | 3030 | 2214 |

.

Matching within block clusters is mandated by the fact that PES-estimation procedures produce adjusted counts at the block-cluster level and aggregate. For matching within block cluster, we additionally use first character of the surname as a logical blocking criteria. Use of first character of surname reduces computation by a factor of 150 but generally causes 3-5 percent of true matches to be missed. Approximately 2 percent of the missed true matches (about half of the 3-5 percent) are in households for which at least one person is correctly matched by computer procedures and, thus, quickly located during clerical review.

The printouts used for the initial clerical review bring together all individuals in a Census household with all individual in a PES household if at least one individual is designated as a computer match or possible match. This, for instance, allows matching of children that are listed under different last names if at least one of the parents is matched.

The remaining 1-2 percent that are not quickly identified as a result of the computer matching operations are located during additional clerical review procedures. This involves manual comparison of PES and Census lists that have been alphabetically sorted by last name and also by address.

Fields available for matching are first name, middle initial, last name, house number, street name, rural route number, postal box number, conglomerated address, unit, telephone number, age, sex, marital status, relationship to head of household, and race. The last name field receives minor processing to separate out words such as Junior, Senior, and III. The first name field receives processing to change nicknames to standard roots of their given first names. A modified version of the Census Geography Division address standardizer is used for delineating address components.

Generally, if house number and street name are not missing, then rural route and box number are missing and vice-versa. The conglomerated address is used as a default identifier for those addresses failing standardization. As the PES reenumerates hard-to-count regions, many addresses such as those associated with trailer courts, migrant worker camps, and various institutional group quarters such as retirement homes and university dormitories are difficult to standardize.

For each PES individual, their true match status and, where appropriate, to whom they were matched is known. The quality of the matching status is very high. The quality is covered in the discussion.

Telephone number, although generally a very good identifier in many applications, is only a somewhat good identifier for the applications in this paper. For instance, in St Louis, only 66 percent of those individuals that were eligible for computer matching had a nonblank telephone number agreeing with the telephone number of the individual to whom matched. For some individuals who were ultimately correctly matched, say those living in trailer courts, telephone numbers sometimes disagree.

### 2.3. Expectation-Maximization Algorithm

Applying the Fellegi-Sunter model involves determining estimates of the conditional probabilities $m(\gamma)$ and $u(\gamma)$. To obtain maximum likelihood estimates we use the EM Algorithm.

For record pairs $r_j$, $j = 1, 2, \cdots, N$, from $Q$, index the comparison vectors $\gamma_j^i$ as follows:

$\gamma_j^i = 1$ if field $i$ agrees for record pair $r_j$

$= 0$ if field $i$ disagrees for record pair $r_j$.

The elements in $Q = (Q \cap M) U (A \cap U)$ are distributed according to a finite mixture with the unknown parameters $\Phi = (m, u, p)$ where $p$ is the proportion of matched pairs in $Q$. Let $\mathbf{x}$ be the complete data vector
$g = <\gamma_j, g_j>$ where

$g_j = (1,0)$ if $r_j \in M \cap Q$ and

$g_j = (0,1)$ if $r_j \in U \cap Q$.

Then the complete data log-likelihood (Dempster, Laird, and Rubin 1977, pp. 15-16) is given by

$$\ln f(\mathbf{x} \mid \Phi) = \sum_{j=1}^{N} g_j \cdot <\ln P(\gamma_j \mid M \cap Q), \ln P(\gamma_j \mid U \cap Q)>$$

$$+ \sum_{j=1}^{N} g_j \cdot <\ln p, \ln(1-p)>.$$

Fitting using the EM Algorithm will be performed under the following conditional independence assumption: There exist vector constants $m \equiv (m_1, m_2, \cdots, m_K)$ and $u \equiv (u_1, u_2, \cdots, u_K)$ such that, for all $\gamma \in \Gamma$,

$$P(\gamma \mid M\cap Q) = \prod_{i=1}^{K} m_i^{\gamma^i} (1-m_i)^{(1-\gamma^i)}$$

and (2.2)

$$P(\gamma \mid U\cap Q) = \prod_{i=1}^{K} u_i^{\gamma^i} (1-u_i)^{(1-\gamma^i)} .$$

Probabilities $m_i$ and $u_i$, $i = 1, 2, \cdots, K$, are constant for all representations $\gamma$ of pairs in $Q$. To avoid trivialities, we assume that $0 < m_i, u_i < 1$, $i = 1, 2, \cdots, K$.

We begin the EM Algorithm with estimates of the unknown parameter

$<\hat{m}, \hat{u}, \hat{p}>$. For the E-step under (2.2), replace $g_j$ with $<P(M\cap Q|\gamma_j), P(U\cap Q|\gamma_j)>$ where

$$P(M\cap Q|\gamma_j) \equiv \frac{\hat{p} \prod_{i=1}^{K} \hat{m}_i^{\gamma_j^i} (1-\hat{m}_i)^{(1-\gamma_j^i)}}{\hat{p} \prod_{i=1}^{K} \hat{m}_i^{\gamma_j^i} (1-\hat{m}_i)^{(1-\gamma_j^i)} + (1-\hat{p}) \prod_{i=1}^{K} \hat{u}_i^{\gamma_j^i} (1-\hat{u}_i)^{(1-\gamma_j^i)}}$$

and

$$P(U\cap Q|\gamma_j) \equiv \frac{(1-\hat{p}) \prod_{i=1}^{K} \hat{u}_i^{\gamma_j^i} (1-\hat{u}_i)^{(1-\gamma_j^i)}}{\hat{p} \prod_{i=1}^{K} \hat{m}_i^{\gamma_j^i} (1-\hat{m}_i)^{(1-\gamma_j^i)} + (1-\hat{p}) \prod_{i=1}^{K} \hat{u}_i^{\gamma_j^i} (1-\hat{u}_i)^{(1-\gamma_j^i)}}.$$

For the $M$ step, the complete data log-likelihood can be separated into three maximization problems. Setting the partial derivatives equal to zero and solving for $\hat{m}_i$,

$i = 1, 2, \cdots, K$, yields:

$$\hat{m}_i = \frac{\sum_{j=1}^{N} P(M \cap Q \mid \gamma_j) \cdot \gamma_j^{\,i}}{\sum_{j=1}^{N} P(M \cap Q \mid \gamma_j)}.$$

Estimates $\hat{u}_i$, $i = 1, 2, \cdots, K$, are derived in a similar manner. The matrix of second partial derivatives can be shown to be negative-definite. The estimate of the proportion of matched pairs is given by

$$\hat{p} = \frac{\sum_{j=1}^{N} P(M \cap Q \mid \gamma^{j})}{N}.$$

The m- and u-probabilities obtained from the EM Algorithm do not generally work well in the decision rule of Fellegi-Sunter because the fitting procedures tend to force m-probabilities away from 1 and u-probabilities away from 0.

To get parameters that yield better decision rules (i.e., both lower rates of false matches and smaller regions of clerical pairs), we use the following procedure:

1. Compute probabilities of random agreement for those pairs agreeing on geocode and on a characteristic such as first name. If the random agreement probability is less than the EM-derived u-probability, substitute. If the resultant u-probability is less than 0.005, replace it with 0.005.
2. Replace the m-probabilities for last and first names with 0.9997 and 0.9999, respectively.

The resultant m- and u-probabilities are still obtained automatically. The effect of 2. is negative disagreement weights of greater absolute value. That the substitution 0.9999 for the m-probability of first name improves distinguishing power is intuitive. As family members, in particular, agree on last name, street name, house number, telephone number, and some demographic characteristics, the more negative disagreement weight induced by the substitution helps delineate different individuals in the same family.

### 2.4. Frequency-Based Weighting

In this section, we also closely follow the terminology of Fellegi and Sunter (1969, section 3.3.1). Let the true frequencies of occurrence of a specified string in files **A** and **B**, respectively, be

$$f_1, f_2, \cdots, f_m \,;\ \sum_{j=1}^{m} f_j = N_A$$

and

$$g_1, g_2, \cdots, g_m \,;\ \sum_{j=1}^{m} g_j = N_B.$$

Let the corresponding true frequencies in **A∩B** be

$$h_1, h_2, \cdots, h_m \,;\ \sum_{j=1}^{m} h_j = N_{AB}.$$

We note that $h_j \le \min(f_j,g_j)$, $j = 1, 2, \cdots, m$. For the empirical examples of section 3.2, for $j = 1, 2, \cdots, m$, we will generally use

$$h_j = \min(f_j,g_j) \quad \text{if } f_j>1 \text{ or } g_j>1$$

$$h_j = 2/3 \qquad \text{otherwise.}$$

The latter part of the definition implicitly means that, if we observe only one pair agreeing on a specific string, the pair has 2/3 chance of being a match and 1/3 chance of being a nonmatch.

The following additional notation is needed

$e_A$ or $e_B$ the respective probabilities of a name being misreported in **A** or **B** (assumed independent of a particular name);

$e_{A0}$ or $e_{B0}$ the respective probabilities of a name not being reported in **A** or **B** (name independent);

$e_T$ the probability that a name is differently (but correctly) reported in the two files.

Then we have the following representations:

P(string agrees & jth string| M)

$$= h_j\,(1-e_A)(1-e_B)(1-e_T)(1-e_{A0})(1-e_{B0})/N_{AB}$$

$\approx h_j\,(1\text{-}e_A\text{-}e_B\text{-}e_T\text{-}e_{A0}\text{-}e_{B0})/N_{AB}$;

P(string disagrees | M)

$= [1\text{-}(1\text{-}e_A)(1\text{-}e_B)(1\text{-}e_T)](1\text{-}e_{A0})(1\text{-}e_{B0}) \approx e_A + e_B + e_T$;

P(string missing on either file | M)

$= 1 - (1\text{-}e_{A0})(1\text{-}e_{B0}) \approx e_{A0} + e_{B0}$;

P(string agrees & jth string| U)

$= (f_j \cdot g_j - h_j)(1\text{-}e_A)(1\text{-}e_B)(1\text{-}e_T)(1\text{-}e_{A0})(1\text{-}e_{B0})/(N_A \cdot N_B - N_{AB})$

$\approx (f_j \cdot g_j - h_j)(1\text{-}e_A\text{-}e_B\text{-}e_T\text{-}e_{A0}\text{-}e_{B0})/(N_A \cdot N_B - N_{AB})$;

P(string disagrees | U)

$$= [1\text{-}(1\text{-}e_A)(1\text{-}e_B)(1\text{-}e_T) \sum_{J=1}^{m} (f_j \cdot g_j - h_j)/(N_A \cdot N_B - N_{AB})](1\text{-}e_{A0})(1\text{-}e_{B0})$$

$$\approx [1\text{-}(1\text{-}e_A\text{-}e_B\text{-}e_T) \sum_{J=1}^{m} (f_j \cdot g_j - h_j)/(N_A \cdot N_B - N_{AB})](1\text{-}e_{A0}\text{-}e_{B0}); \text{ and}$$

P(string missing on either file | U)

$= 1 - (1\text{-}e_{A0})(1\text{-}e_{B0}) \approx e_{A0} + e_{B0}$.

We define the weight for agreement on the jth specific string, j = 1, 2, ···, m, by

$\text{wgt}(j) = h_j \cdot (N_A \cdot N_B\text{-}N_{AB}) / ((f_j \cdot g_j\text{-}h_j) \cdot N_{AB})$,

if either $f_j>1$ or $g_j>1$ and by

$\text{wgt}(j) = 2 \cdot (N_A \cdot N_B\text{-}N_{AB})/N_{AB}$

if $f_j=1$ and $g_j=1$. The weight represents the probability of agreement over the entire product space.

We observe that $e_{A0}$ and $e_{B0}$ can be estimated directly using file characteristics. To estimate $e_T$, $e_A$ and $e_B$, we need to know $\mathbf{A} \cap \mathbf{B}$. They cannot be estimated directly. In practice, guesses based on past experience are often used.

Use of the EM-Algorithm, within the limitations of the previous section, allows direct estimation of P(string disagrees|M) and, thus, approximate estimation of the sum $e_A+e_B+e_T$.

To assure that frequency weights do not overwhelm simple agree/disagree weights given by the EM procedure, we use for the agreement weights

$$c_1 \cdot \ln( c_2 \cdot P(\text{jth string agrees} \mid M) / P(\text{jth string agrees} \mid U) ).$$

For last name $c_1$ and $c_2$ are 0.03125 and 0.5, respectively. For first name $c_1$ and $c_2$ are 0.0625 and 0.7, respectively.

General methods of directly computing $c_1$ and $c_2$ are given in Winkler (1989b) but do not work well for all types of files. For the narrow classes of files we consider in the applications of this paper, we can choose $c_1$ and $c_2$ a priori based on modeling files with similar characteristics.

2.5. String Comparator Metrics

When human beings review a pair, they often do not make simple yes/no decisions related to the weight of agreement associated with a fixed string. Typically, they will assign various degrees of maybe to a pair of strings exhibiting typographical variation (Table2).

Table 2. Hypothetical Human Decisions Based on Typographical Errors
Full Agreement = 10, Full Disagreement = -10

| Pair | Jonathon<br>Johathon | Bellieu<br>Baliew | Ovid<br>Ouid | Wilansk<br>Wolansky | Doret<br>Doris |
|---|---|---|---|---|---|
| Weight | 10 | 4 | 1 | -3 | -10 |

Jaro (see e.g., Winkler 1985, 1989c, 1990b) introduced a string comparator measure that gives values of partial agreement between two strings. The string comparator accounts for length of strings and partially accounts for the types of errors typically made in alphanumeric strings by human beings. It is used in adjusting exact agreement weights when two strings do not agree on a character-by-character basis.

Specifically, if $c > 0$, the Jaro string comparator is

$$\Phi = W_1 \cdot c/d + W_2 \cdot c/r + W_t \cdot (c-\tau)/c,$$

where

$W_1$ = weight associated with characters in the first of two files,
$W_2$ = weight associated with characters in the second of two files,
$W_t$ = weight associated with transpositions,
$d$ = length of string in first file,
$r$ = length of string in second file,
$\tau$ = number of transpositions of characters, and
$c$ = number of characters in common in pair of strings.

If $c = 0$, then $\Phi = 0$.

Two characters are considered in common only if they are no further apart than $(m/2 - 1)$ where $m = \max(d,r)$. Characters in common from two strings are assigned; remaining characters unassigned. Each string has the same number of assigned characters.

The number of transpositions is computed as follows: The first assigned character on one string is compared to the first assigned character on the other string. If the characters are not the same, half of a transposition has occurred. Then the second assigned character on one string is compared to the second assigned character on the other string, etc. The number of mismatched characters is divided by two to yield the number of transpositions.

If two strings agree on a character-by-character basis, then the Jaro string comparator $\Phi$ is set to $W_1+W_2+W_t$, which is the maximum value that $\Phi$ can assume. The minimum value that the $\Phi$ can assume is 0, which occurs when the two strings have no characters in common (subject to the above definition of common).

For present matching applications, $W_1$, $W_2$, and $W_t$ are arbitrarily set to 1/3. The new string comparator metric basically modifies the basic string comparator according to whether the first few characters in the strings being compared agree. Specifically, for $i = 1, 2, 3, 4$,

$$\Phi_n = \Phi + i \cdot 0.1 \cdot (1 - \Phi) \quad \text{if the first } i \text{ characters agree.}$$

If $w_a$ and $w_d$ are the estimated agreement and disagreement weights for a specific field, respectively, then the Jaro adjusted matching weight $w_{am}$ used in the total weight calculation is given by

$$w_{am} = \begin{array}{ll} w_a & \text{if } \Phi = 1, \text{ and} \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi)\cdot(9/2), w_d\} & \text{if } 0 \leq \Phi < 1. \end{array}$$

The constant 9/2 controls how quickly decreases in partial agreement values force the adjusted weight to the disagreement weight.

Instead of assuming that the same adjustment procedure works for different fields such as first name, last name, and house number, procedures for modeling the weight adjustment as a piecewise linear function were developed. The procedures necessitate having representative sets of pairs for which the truth of matches is known. The new adjusted weights $w_{na}$ take the form

$$w_{na} = \begin{array}{ll} w_a & \text{if } \Phi_n \geq b_1 \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi_n)\cdot(a_1), w_d\} & \text{if } b_2 \leq \Phi_n < b_1. \\ \max\{w_a-(w_a-w_d)\cdot(1-\Phi_n)\cdot(a_2), w_d\} & \text{if } \Phi_n < b_2. \end{array}$$

The constants $a_1$, $a_2$, $b_1$, and $b_2$ depend on the specific type of string (such as first name) to which the weight adjustment is applied. Generally, $a_1 < a_2$. The specific constants used are given in Table 3.

Table 3. Constants used in piecewise linear weight adjustments

| Field | $a_1$ | $a_2$ | $b_1$ | $b_2$ |
|---|---|---|---|---|
| first | 1.5 | 3.0 | .92 | .75 |
| last | 3.0 | 4.5 | .96 | .88 |
| house # | 4.5 | 7.5 | .98 | .83 |

Table 4 provides examples of string comparator values for pairs of last names and for pairs of first names. The abroms-abrams example with string comparator value .9333 in contrast to the lampley-campley with value .9048 shows that the string comparator gives a higher value to the pair that differs by a single character further from the first position. The martha-marhta example with value .9667 in contrast to the jonathon-jonathan example with value .9583 shows that transposition of two characters causes less of a downweighting than differing by one

Table 4. Examples of String Comparator Values for Various Pairs

| | | |
|---|---|---|
| shackleford | shackelford | .9848 |
| cunningham | cunnigham | .9833 |
| campell | campbell | .9792 |
| nichleson | nichulson | .9630 |
| massey | massie | .9444 |
| abroms | abrams | .9333 |
| galloway | calloway | .9167 |
| lampley | campley | .9048 |
| dixon | dickson | .8533 |
| | | |
| frederick | fredrick | .9815 |
| michele | michelle | .9792 |
| jesse | jessie | .9722 |
| marhta | martha | .9667 |
| jonathon | jonathan | .9583 |
| julies | juluis | .9333 |
| jeraldine | geraldine | .9246 |
| yvette | yevett | .9111 |
| tanya | tonya | .8933 |
| dwayne | duane | .8578 |

### 2.6. Linear Sum Assignment Algorithm

A linear sum assignment procedure similar to the one introduced by Jaro (1989) is used to force one-to-one assignments. The actual algorithm, due to Errol Rowe (1987), requires less storage than the one of Burkard and Derig (see e.g., Jaro 1989, page 418). Basically, the Rowe algorithm works with rectangular arrays while the Burkard-Derig algorithm only works with square arrays. The algorithm also makes use of some of the characteristics of the arrays of weights. For the arrays of weights occurring when the set of PES records remaining from within-block matching are matched against Census files covering extended areas, the new algorithm is more than 100 times as fast as the previously used algorithm.

## 3. RESULTS

The section presents a comparison of matching results. The basic strategies are the 1988 strategy used during the Dress Rehearsal Census and the 1990 computer matching strategy. The differences and similarities between the two strategies are presented in Table 5.

Table 5. Comparison of 1988 and 1990 Matching Strategies

| | 1988 | 1990 |
|---|---|---|
| logical blocking | cluster+ soundex surname | cluster+ first char surname |
| pairs for parameter estimation | cluster+ soundex surname | cluster+ soundex surname |
| multi string comparison adjust | no | yes |
| frequency-based | no | yes |

. This section also presents comparisons with several other strategies which will be described.

A comparison of matching results is given in Tables 6, 7, and 8 for St Louis, Columbia, and Washington, respectively. To understand the tables, we need describe the types of matching procedures. The simplest procedure, crude, merely uses an ad hoc guess for matching parameters and does not use string comparators. The ad hoc guess consists of assigning matching weights of ±2 to agreement/disagreement on fields such as first name, house number, and age and assigning matching weights of ±1 to less important fields such as marital status and sex.

The next, param, does not use string comparators but does estimate the probabilities $m(\gamma)$ and u(γ). Such probabilities are often estimated through an iterative procedure that involves manual review of matching results and successive reuse of the reestimated parameters. The third type, param2, uses the same probabilities as param and the basic string comparators.

The fourth type, em, uses an EM-Algorithm for estimating matching parameters (see e.g., Winkler 1988, Thibaudeau 1990) and uses the basic Jaro string comparator. The fourth type is the 1988 matching strategy. The fifth type, em2, uses the EM-derived weights and the new string comparator and new weight adjustments. The final type, freq, replaces simple agree/disagree weights for first name and last name with frequency-based weights (see e.g., Winkler 1989) and also makes adjustments for joint dependencies of agreements on first name, sex, and age. The final type is the 1990 matching strategy.

In each table, the number of matches is determined by a false match rate of 0.002. The crude and param types are allowed to rise slightly above the 0.002 level because they generally have higher error levels.

Table 6. True Matches in Computer Categories
Various Procedures, St Louis
10291 True Matches, 12072 Records
Pairs Agreeing on Cluster and First
Character Surname 1/

| | --computer designation-- | | | |
|---|---|---|---|---|
| | match | | clerical | |
| truth-> | match | non-match | match | non-match |
| crude | 310/ | 1 | 9344/ | 794 |
| param | 7899/ | 16 | 1863/ | 198 |
| param2 | 9276/ | 23 | 545/ | 191 |
| em (1988) | 9587/ | 23 | 271/ | 192 |
| em2 | 9639/ | 24 | 215/ | 189 |
| freq (1990) | 9801/ | 24 | 52/ | 94 |

1/ Approximately 400 true matches disagree on first character of surname and are not eligible for inclusion in the table. Of the 400, approximately 200 reside in households where some is correctly matched.

By examining the tables we observe that a dramatic improvement in matches can occur when string comparators are first used (from param to param2). The basic reason is that disagreements (on a character-by-character basis) are replaced by partial agreements. The improvements due to the new string comparators and weighting adjustments (from em to em2) are quite minor.

The main matching results are given in Tables 6, 7, and 8 for St Louis, Columbia, and Washington files, respectively. For St Louis files, with the 1990 strategy, 9801 of 10291 true nonmover matches are designated as matches by the computer with an error rate of 0.2 percent. An additional 55 true matches are in the set of designated clerical pairs and also quickly obtained. Thus, the computer matcher allows 96 percent (= 9856/10291) of the true nonmover matches to be obtained quickly. For Columbia and Washington, the corresponding percentages are 97 (= 6798/6984) and 97 (= 1899/1950), respectively.

Table 7. True Matches in Computer Categories
Various Procedures, Columbia
6984 True Matches, 7649 Records
Pairs Agreeing on Cluster and First
Character Surname 1/

| | --computer designation-- | | | |
|---|---|---|---|---|
| | match | | clerical | |
| truth-> | match | non-match | match | non-match |
| crude | 2429/ | 7 | 4327/ | 119 |
| param | 6449/ | 22 | 327/ | 92 |
| param2 | 6655/ | 13 | 135/ | 35 |
| em (1988) | 6719/ | 13 | 78/ | 22 |
| em2 | 6762/ | 13 | 37/ | 20 |
| freq (1990) | 6792/ | 11 | 6/ | 9 |

1/ Approximately 180 true matches disagree on first character of surname and are not eligible for inclusion in the table. Of the 180, approximately 80 reside in households where some is correctly matched.

Because printouts for clerical review and by PES household and some of the individuals not designated as matches or clerical pairs reside in households having individuals containing matching individuals, the additional individuals are also found during clerical review. The

Table 8. True Matches in Computer Categories
Various Procedures, Washington
950 True Matches, 2214 Records
Pairs Agreeing on Cluster and First
Character Surname 1/

| | --computer designation-- | | | |
|---|---|---|---|---|
| | match | | clerical | |
| truth-> | match | non-<br>match | match | non-<br>match |
| crude | 1307/ | 3 | 564/ | 98 |
| param | 1250/ | 5 | 614/ | 88 |
| param2 | 1765/ | 4 | 134/ | 41 |
| em (1988) | 1749/ | 4 | 149/ | 29 |
| em2 | 1795/ | 3 | 107/ | 29 |
| freq (1990) | 1892/ | 4 | 7/ | 9 |

1/ Approximately 40 true matches disagree on first character of surname and are not eligible for inclusion in the table. Of the 40, approximately 20 reside in households where some is correctly matched.

percentages of these individuals are 2, 1-2, and 1-2 percent in St Louis, Columbia, and Washington, respectively. Thus, individuals quickly found via computer match/printout procedures are 98, 98-99, and 98-99 percent in St Louis, Columbia, and Washington, respectively.

## 4. DISCUSSION

### 4.1. How the Quality of Files Effects Matching Results

The overall success of the procedures introduced in this paper is highly dependent on the facts that individual fields contain relatively few typographical variations and that there are redundant fields. Comparing strings having relatively severe typographical variations yields disagreement weights rather than the agreement weights that would occur without the typographical variation. If fields were without typographical variation, then having three crucial fields such as last name, first name, and house number might be sufficient to identify uniquely an individual. Having additional matching fields such as street name, age, and telephone number would be redundant. For those records having typographical variation or missing fields, the redundancy provides the extra information needed so that the computerized procedures can delineate many more pairs as matches.

Winkler (1989b, section 3.5) provides an example of matching techniques that are quite similar to the techniques of this paper for files of administrative records. On an absolute basis, overall matching results are much worse than the results of this paper. The reasons are that there is relatively less information for matching and available information is often missing or inaccurate. For these files, matching procedures that use EM-derived parameters, frequency-based techniques, and string comparator metrics also yield better results than those procedures that do not.

The data of this paper are suitable for evaluating matching procedures becauses essentially all matches were found and correctly identified. The identification is with codes specifying to which record a record is matched. All basic identifying information was carefully checked and rechecked. In particular, virtually no matches were found among the set of code-identified nonmatches using a variety of procedures.

### 4.2. String Comparison and Weight Adjustment in the Fellegi-Sunter Model

For matching applications of files having significantly different characteristics (i.e., matching fields) from those of the files of this paper, string comparator weighting adjustments may have to be remodeled.

In all matching situations, it seems likely that modeling partial agreement should improve matching efficacy because the proportions of exact agreement on key matching fields can be quite low. For the files of this paper, the proportions of true matches agreeing on a character-by-character basis ($\Phi_n$=1.0) are approximately 76 percent for first name and approximately 86 percent for last name (Table 9).

Table 9. Proportional Agreement by
String Comparator Values
Key Fields by Geography

|  | StL | Col | Wash |
|---|---|---|---|
| First |  |  |  |
| $M_n=1.0$ | 0.75 | 0.82 | 0.75 |
| $M_n\geq 0.6$ | 0.93 | 0.94 | 0.93 |
| Last |  |  |  |
| $M_n=1.0$ | 0.85 | 0.88 | 0.86 |
| $M_n\geq 0.6$ | 0.95 | 0.96 | 0.96 |

Most of the reason that the crude and param methods (Tables 6, 7, and 8) perform relatively poorly is that they only assign the full agreement weight for first name and surname to the 86 and 76 percent, respectively, of the pairs that are truly matches and agree on a character-by-character basis. The remaining 24 and 14 percent, respectively, of the true matches get the full disagreement even though a number of them differ by only one or two characters.

### 4.3. Weights Produced by the EM Algorithm

The obvious reason that probabilities and associated agreement and disagreement weights produced by the EM Algorithm do not work that well is the failure of the independence assumption. Decision rules that incorporate interaction effects into the weighting on either an ad hoc or formal basis may yield improved matching efficacy and/or automation.

### 4.3.1. Weights From General Statistical Fitting Procedures

Work by Winkler (1989a) and Thibaudeau (1989) using the formal model of three-way interactions (Bishop, Fienberg, and Holland 1975) and six matching variables have yielded normalized chi-square fits that are generally 200 times as accurate as fits under the independence model. Matching software, however, that uses three-way weights generally has not performed any better than the software of section 3 that utilizes simple agree/disagree weights.

Only with highly expert adjustments, does the 3-way matching software make slight improvements over the em results of section 3. The computation of 3-way probabilities, using theoretical and computational results generalizing results of Haberman (1977, 1976), are excessively slow to converge. There are two chief difficulties. The first is that the maximization step of the EM Algorithm for 3-way interactions is via an iterative fitting procedure whereas for independent situations the maximization step is closed form. The second is that different starting points typically yields different limiting solutions in the 3-way model whereas the limiting solutions are typically unique in the independent case.

Recently, Thibaudeau (1990) developed a scoring algorithm approach for fitting 3-way interaction models that converges much more rapidly than the previous fitting methods. In terms of chi-square statistics, the fits are only 20 times as good as the independent fits and would be rejected by any reasonable hypothesis test. They do, however, yield decision rules that are better than those using parameters produced by the independent EM Algorithm. While the resultant parameters do not yield decision rules that work as well as the best of this paper, they do yield improvements without the expert, file-specific adjustments that are used for the results of section 3. The reason for the improved decision rules is that the fitting procedures are better are finding local maxima of the likelihood and at rejecting unacceptable starting points.

### 4.3.2. Comparison With Existing Weighting Methods

All other existing computer matching systems of which we are aware use time-consuming, trial-and-error procedures for estimating matching parameters. Generally, initial guesses of matching parameters are used, matched pairs are reviewed, new matching parameters guessed at and the matching is repeated until satisfactory matching results obtained. In the hands of experts, such procedures are generally quite robust.

As the Generalized Iterative Record Linkage System of Statistics Canada (1983) is the only system to systematize the iterative review process and, thus, make it relatively usable by nonexperts, we use it as a point of comparison. Because we know truth and falsehood of matches, we merely assume that the known marginals P(agree specific field | Match) are the result of the iterative procedure. The Statistics Canada iterative procedures assume that convergence is generally to P(agree specific field | Match). Statistics Canada uses random agreement weights for P(agree specific field | Nonmatch) just as we do.

The matching results (Tables 6, 7, and 8) for the param procedure correspond roughly to a Statistics Canada procedure that does not use frequency-based weights and the param2 procedure adds the string comparator metrics. The comparisons between the em procedure and the param2 show that the EM-produced weights perform better. The reason is that, while both set of parameters are associated with independent distributions, the EM-produced weights are not constrained to satisfy additional marginal constraints P(agree specific field | Match).

4.4. Frequency-Based Weights

For the matching applications of this paper frequency-based weights make a noticeable improvement (e.g., section 3) over simple agree/disagree weights. The improvement is basically due to pairs having relatively rare first names and surnames that are designated as possible matches using simple agree/disagree weights but are designated as matches using frequency-based matches.

For applications to files having relatively poor distinguishing information (e.g., Winkler 1989b, section 3.5) relative improvements can be even greater. In such applications, only first and last name are generally available and they have relatively greater typographical variation. With simple agree/disagree weights very few pairs can be reliably designated as matches or clerical pairs. With frequency-based weighting a moderate percentage of pairs change from nonmatch to clerical pairs.

4.5. Linear Sum Assignment Procedure

The reason the assignment works well for the particular applications considered in this paper is the nature of the data for individuals in households that are brought together. Generally, all the individuals will have good information or they will all have substantial missing or erroneous data (e.g., as a result of proxy information). In both cases clerical review sets are necessarily much larger.

In the first case, say, the assignment procedure eliminates all pairs (such as brother-sister or husband-wife) that might have moderately high weights. The UPPER cutoff would have to be raised to force them into the set of clerical pairs. If such pairs were retained, additional pairs based having only moderately good identifying information (e.g., with some typographical differences) would no longer be in the set of computer-designated matches.

We observe that if one reasonable assumption is made, then the decision rules that incorporates the linear sum assignment procedure are still optimal. The assumption is that the basic weighting procedure (prior to the linear sum assignment) always makes the best assignment (either to the true match or the more closely agreeing nonmatch). Then the decision rule modified with the linear sum assignment procedure is still optimal. Indeed, under the assumption, the second through nth best assignments will always be associated with true nonmatches and the likelihood ratio associated with them should be forced to 0.

4.6. Determination of Cutoff Weights

Fellegi and Sunter (1969, section 3.3.2 and 3.7; see also Jaro 1989, section 3.4) provide methods for calculating the cutoff weights (or threshold values) UPPER and LOWER provided the probabilities m( ) and u( ) can be given a strict probabilistic interpretation. In such a situation, if we are given an upper bound on the false match rate, we should be able to use m( ) and u( ) directly to computer UPPER and LOWER.

Our experience has been that the probabilities m( ) and u( ) for each of the methods of section 3 (e.g., param, em, etc.) are never suitable for direct estimation of the cutoffs because the estimated m( ) and u( ) deviate much too severely from the true underlying probabilities. Part of the deviation is due to the failure of the independence assumption that is typically used. Other deviations can be caused by having a small subpopulation that has characteristics significantly different the the population as a whole.

In practice, we determine UPPER and LOWER by reviewing a set of pairs that are printed by decreasing matching weight. Based on experience, it is quite rapid to determine the cutoffs with a rough a priori bound on the error rates.

Belin and Rubin (1990) have introduced a method for directly determining the cutoffs at desired error levels for any weighting curve. Their method necessitates having a training set of weights associated with matches and nonmatches. Under the assumption that the weighting curve is a mixture of normals that can be transformed to normality via Box-Cox techniques, they provide an EM-type Algorithm to estimate the curves for new data sets. The a priori information is used primarily to get the shapes of the curves. Their method yields estimates of the error rate and estimates of the corresponding confidence intervals.

## 5. SUMMARY AND CONCLUSIONS

### 5.1. Summary

This paper provides a methodology for computer matching that generally falls under the formal decision-theoretic procedures given by Fellegi and Sunter. The two crucial ideas for improving matching efficacy are having parameters estimated automatically via a modified version of the Expectation-Maximization Algorithm and using string comparator metrics for assigning adjusted weights to pairs of strings that agree almost exactly rather than using full disagreement weights.

### 5.2. Conclusion

At present, the very high quality of the matching results appears very dependent on having files similar to those that will be available during 1990 Post Enumeration Survey processing.

### 5.3. Future Work

New research on more general, theoretically valid string comparator metrics and associated weighting adjustments is just beginning. Research into decision rules that allow for general interactions rather than independent iteractions continues and shows promise.

Applications to general administrative lists and to lists of businesses is just beginning. Matching of business lists is highly dependent on the quality of name and address parsing software.

## REFERENCES

Belin, T. R. and Rubin, D. B. (1990), "Calibration of Errors in Computer Matching for Census Undercount," *American Statistical Association, Proceedings of the Section on Government Statistics*, to appear.

Bishop, Y. M. M., Fienberg, S. E., and Holland, P. W. (1975), *Discrete Multivariate Analysis*, MIT Press, Cambridge, MA.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977) "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*, **B**, **39,** 1-38.

Fellegi, I. P., and Sunter, A. B. (1969), "A Theory for Record Linkage," *Journal of the American Statistical Association*, **40**, 1183-1210.

Haberman, S. J. (1976), "Iterative Scaling for Log-Linear Model for Frequency Tables Derived by Indirect Observation, *American Statistical Association, Proceedings of the Section on Statistical Computing*, 45-50.

Haberman, S. J. (1977), "Product Models for Frequency Tables Involving Indirect Observation," Annals of Statistics, **5** 1124-1147.

Haberman, S. (1979), *Analysis of Qualitative Data*, Academic Press. New York.

Jaro, M. A. (1989), "Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida,"*Journal of the American Statistical Association,* **89**, 414-420.

Newcombe, H. B. (1988) *Handbook of Record Linkage: Methods for Health and Statistical Studies, Administration, and Business*. Oxford: Oxford University Press.

Newcombe, H. B., Kennedy, J. M., Axford, S. J., and James, A. P. (1959), "Automatic Linkage of Vital Records," *Science*, **130** 954-959.

Rowe, E. (1987), private communication of FORTRAN and Pascal versions of linear sum assignment algorithm.

Statistics Canada (1983) "Generalized Iterative Record Linkage System", Systems Development Division.

Thibaudeau, Y. (1989), "Fitting Log-Linear Models When Some Dichotomous Variables are Unobservable,"*American Statistical Association, Proceedings of the Section on Biometrics,* to appear.

Thibaudeau, Y. (1990), "Improving the Performance of Computer Matching Algorithms Through Better Choices of Parameters," paper presented at Annual *American Statistical Association* meeting in Anaheim, California.

Winkler, W. E. (1985), "Preprocessing of Lists and String Comparison," in *Record Linkage Techniques- 1985*, edited by W. Alvey and B. Kilss, U.S. Internal Revenue Service, Publication 1299 (2-86), 181-187.

Winkler, W. E. (1988), "Using the EM Algorithm for Weight Computation in the Fellegi-Sunter Model of Record Linkage," *American Statistical Association, Proceedings of the Section on Survey Research Methods*, 667-671.

Winkler, W. E. (1989a), "Near Automatic Weight Computation in the Fellegi-Sunter Model of Record Linkage," *Proceedings of the Fifth Census Bureau Annual Research Conference,* 145-155.

Winkler, W. E. (1989b), "Frequency-based Matching in the Fellegi-Sunter Model of Record Linkage," *American Statistical Association, Proceedings of the Section on Survey Research Methods,* 788-793.

Winkler, W. E. (1989c, "The Interaction of Record Linkage Theory and Practice," Ottawa, Ontario, Canada, August 1989, *Proceedings of the Record Linkage Sessions and Workshop*, 139-148.

Winkler, W. E. (1990a), "Bootstrap Evaluation of Calibration Procedures used for Decision Rules in the Fellegi-Sunter Model of Record Linkage," *Computing Science and Statistics, Proceedings of Interface '90,* edited by LePage, R. and Billard, L., to appear

Winkler, W. E. (1990b), "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage," *American Statistical Association, Proceedings of the Section on Survey Research Methods*, to appear.