How Many Ways Can Things Be The Same? Set Theory For Multiple-Site Surveys

Richard G. Clegg, Department of Mathematics, University of York,
YORK, YO10 5DD, *email: richard@manor.york.ac.uk*

Abstract

This paper describes a rigorous set-theoretic framework for analysing the result of surveys which take place over multiple sites and where the surveyor needs to match surveyed items between more than two of those sites. In the analysis of roadside survey data, it is often desirable to analyse matches between several data sets simultaneously. For example, we might wish to answer questions of the general type “How many drivers are seen at point A, point B and point C?” or “How many vehicles are seen on all five survey days?” This paper attempts to create a general framework for the analysis of matching between data from more than two surveys. The framework is then applied to the specific case of false matching in partial licence plate surveys (that is non-matches which are mistaken for matches because only part of the licence plate is observed). It should be stressed throughout that the framework outlined is applicable to any data series where matches are sought between two or more distinct data sets.

In the first part of the paper, the general problem is described and a brief review is given of other work on licence plate surveys and the false match problem.

In the second part of the paper a framework is developed which formalises the concept of *type of match* using very basic set theory and the notion of the equivalence class. The set $\mathcal{M}_n$ (the set of all possible types of match for n sites) is described and a formal procedure is given for calculating it and for enumerating its members. This general framework can be used to investigate a number of types of question about matches between data in multiple data sets. The problem is placed in the context of partitions of natural numbers and some simple results are given showing how many types of match there are for n survey sites.

In the third part of the paper, the framework is extended by introducing a *partial ordering* and it is shown how this applies to the problem of false matches. An algorithm is given for the estimation of the number of true matches.

In the fourth part of the paper simulations are run using this algorithm to remove false matches in artificially created data sets. The algorithm is shown to be successful in estimating the number of true matches though the variance can be large for some cases.

1 Introduction

This paper describes a general framework for analysing problems in matching data across multiple data sets. The method developed is useful for situations where analysis is to be performed on seveal data sets containing information about unique individuals. The method answers questions of the type “How many unique individuals appear in three or more of the five data sets?” and is particularly useful for addressing situations where false matches are possible (that is, where two distinct individuals appear to be the same as a result of observational error).

The problem originally arose during roadside traffic surveys when attempting to locate vehicles using their licence plates at multiple survey sites across a city. It should be emphasised. however, that the framework is sufficiently general that it could prove of use in any situation where it is important to track matches in data items across a small number of different data sets. In the real life situation reported, the number of false matches could often be a significant fraction of the number of matches recorded.

Using set theory, the problem has been placed in the context of lattices of the integer partition and a solution algorithm has been developed. The algorithm answers problems of the type “How many individuals are genuinely seen once each in every data set when the false matches have been excluded?” The algorithm has been implemented in the C++ programming language and tested on simulated data sets. The test results strongly suggest that the method does indeed provide an unbiased estimator for the true number of matches in the data although the variance in the estimate can, unfortunately, be extremely high in some cases. The method has been tested and found useful in removing false matches from real data although the variance on the estimate can be high.

Throughout this paper the term n-*tuple* is used to describe an ordered set of n elements — somewhat akin to an n-*vector* but not usually an element within a vector space. The tuples are ordered sets of general elements, sometimes sets of other sets are used. The notation of making an n-tuple bold will be used and its individual elements will be subscripted: $\mathbf{x} = (x_1, \ldots, x_n)$.

1.1 Background And Context Of The Problem

The problem of tracking individual vehicles on a road network is a well-known and common problem in transport surveys. Several approaches are used, for example GPS location (Jan, Horowitz and Peng 2000), (Quiroga, Henk and Jacobson 2000) or cell-phones and vehicle tags (Dixon and Rilett 2002). However, a common method is the licence plate survey which may be done either manually (using a roadside observer with a note pad or a dictaphone or specialist recording equipment for the purpose) or automatically with roadside cameras (Williams, Kirby, Montgomery and Boyle 1988). In both manual and automatic surveys the problem of errors in the recordings must be considered. (Slavik 1985), (Schaefer 1988) describe some of the difficulties with such surveys. Manual surveys are commonly partial plate surveys (for reasons of time and convenience) and, in addition to the recording errors, the problem of accidental *false matches* between different vehicles which have the same partial plate is an important one.

A number of researchers have approached the false matching problem for licence plates. (Hauer 1979) provides the earliest approach for removing false matches between observations at two sites. (Maher 1985) describes several methods for approaching the problem including a method for making two point matches between pairs selected from a number of survey sites. (Watling and Maher 1988) gives a graphical method which provides a good visualisation of the problem. (Watling and Maher 1992) describes a further refinement adding journey time information into the mix. (Watling 1994) provides a maximum likelihood estimator for the true matches based upon assumptions about the statistical nature of the inbound traffic and (Liu 2002) extends this method to three sites. However, none of the authors tackles the general problem of removing false matches from matches across n sites.

More generally, a considerable amount of work has been done on “matching problems” in combinatorics — the usual approach being graph theoretic with an edge between two nodes indicating a match. However, in the case of matches across n data sets then the graph theoretic approach is inappropriate since the matches are not just pairwise.

The framework developed in the next section considers problems of the type *“How many individuals occur in three of the five data sets?”* or *“How many individuals are genuinely seen in all five data sets being investigated once all false matches are removed?”*. The framework places the problem in the context of basic set theory (Halmos 1970) and shows how the problem maps onto the well-known mathematical topic, partitions of the natural numbers.

The motivating problem for this paper is a genuine one which arose in the course of road traffic research. The problem arose when partial licence plate data was collected across a number of survey sites (the survey itself is described in (Clegg 2003a)). In the survey undertaken, the researchers wished to know how many vehicles were seen on all of six survey days. Because only partial plate surveys were conducted, false matches occurred. In extreme cases, the number of matches attributed to false matching in data actually exceeded the number of genuine matches. The problem is a surprisingly tricky one since false matches can occur in a huge number of ways. For example, the same partial plate observed on all five weekdays could represent: a single vehicle identified on all five days; five vehicles which by coincidence have the same partial plate, one observed on each day; one vehicle observed on monday and a second vehicle observed on tuesday through until friday; one vehicle observed on monday and tuesday, a second vehicle observed on wednesday and friday and a third vehicle observed only on thursday, and all three having the same partial place; or any of a multiplicity of other ways false matches could occur. Indeed, it is quickly clear that merely enumerating the ways in which a false match can occur is a tricky problem.

1.1.1 Notes On Licence Plate Observation

Throughout this paper, examples will be given using licence plates with a specific format. An example plate would be: `A134SDR`. This type of plate was used in the UK from 1983 up until mid 2001 (Automobile Association 2003). The specific details of the type of plate used are completely irrelevant to the methods developed within this paper, however, chosing parts of a plate to survey for partial plate surveys and estimating the probability of two unique plates matching is not straightforward due to correlations related to year and location identifiers on licence plates. This topic is, however, not of general interest and is not covered here.

2 Setting For The Problem

Assume that there are n data sets (survey sites) and at each site i there exist a set of observations $\mathbf{S}_i$. Each observation is a sighting of one from a set of identifiable, unique individuals $\Omega = \{\omega_1, \ldots, \omega_N\}$ where N is the number of individuals. The n-tuple of all n sites is denoted by $\mathbf{S}$ where $\mathbf{S} = (\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_n)$. It is assumed, initially, that enough information will be recorded in an observation to distinguish between any two members of Ω — this assumption will be relaxed later.

Definition 1. The *observation function*, $f(\omega)$ is a function acting on the members of Ω such that:

$$(i = j) \Leftrightarrow (f(\omega_i) = f(\omega_j)) .$$

In other words, the observation function is a function which uniquely identifies the objects observed. If the objects are different then the result of the observation function is different.

In other words, the observation represented by the function is enough to uniquely determine the object observed and distinguish it from all other such objects. It should be noted that the function $f(\omega)$ need not be real valued. For example, if Ω represents the UK vehicle fleet then $f(\omega)$ could have as its range the set of all possible UK licence plates. The members of the sets $\mathbf{S}_i$ will be observations $f(\omega)$ with $\omega \in \Omega$. Therefore, for each site i:

$$\mathbf{S}_i = \{f(\omega_{j(1)}), f(\omega_{j(2)}), \ldots, f(\omega_{j(N)})\}, \tag{1}$$

where N is the number of observations at site i and $\omega_{j(k)} \in \Omega$ for all $j(k)$. The $j(k)$ are indices representing the place of the observation in the set Ω.

A technicality which should be noted in passing is the possibility that some ω_j is observed more than once in a set of observations $\mathbf{S}_i$ (in other words, an individual is observed twice at the same site). This would cause a problem since, formally, a set cannot contain distinct members which are identical as would be the case if, $j(k) = j(l)$ for any $k \neq l$ in equation (1). This problem will be made worse when the requirement that observations uniquely determine individuals is dropped. To prevent this problem, the observations could be, for example, tagged with a time of day or a suffix to denote the order in which the observation was made. This requirement is a pure technicality and will not affect anything which follows nor will it be mentioned again.

Definition 2. An n-*tuple of observations* can be formed by taking one observation from each of the n sites in order. That is:

$$\mathbf{x} = (x_1, \ldots, x_n),$$

where $x_i \in \mathbf{S}_i$.

To make this more concrete, consider the following three sets of observations:

$$\mathbf{S}_1 = \{\texttt{A123XYZ}, \texttt{B256ABC}\}$$
$$\mathbf{S}_2 = \{\texttt{A123XYZ}, \texttt{C232SAD}, \texttt{B256ABC}\}$$
$$\mathbf{S}_3 = \{\texttt{C789ABC}, \texttt{A123XYZ}, \texttt{A543OPQ}\}.$$

Three possible n-tuples of observations are:

$$\mathbf{x} = (\texttt{A123XYZ}, \texttt{A123XYZ}, \texttt{C789ABC}) \tag{2}$$
$$\mathbf{y} = (\texttt{A123XYZ}, \texttt{A123XYZ}, \texttt{A123XYZ}) \tag{3}$$
$$\mathbf{z} = (\texttt{B256ABC}, \texttt{B256ABC}, \texttt{A123XYZ}). \tag{4}$$

Definition 3. The set $\mathcal{S}$ is the set of all possible such n-tuples across the observations in the set of sites $\mathbf{S}$. This is given by the Cartesian product:

$$\mathcal{S} = \mathbf{S}_1 \times \mathbf{S}_2 \times \ldots \times \mathbf{S}_n = \prod_{i=1}^{n} \mathbf{S}_i.$$

It follows immediately that the number of possible n-tuples $\#\mathcal{S}$ is given by $\prod_{i=1}^{n} \#\mathbf{S}_i$.

2.1 Types Of Match

Consider the tuples, $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ as given by equations (2), (3) and (4). It is clear that in some sense that $\mathbf{x}$ and $\mathbf{z}$ are in some sense the same type of tuple (they are observations of the same vehicle at sites one and two and a different vehicle at site three). It is equally clear that $\mathbf{x}$ and $\mathbf{y}$ are in some sense a different type of tuple. This concept of type of match is formalised by an equivalence relation.

Definition 4. Two n-tuples of observations $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ are the same *type of match* if and only if $\mathbf{x} \sim \mathbf{y}$ where $\sim$ is the equivalence relation defined by:

$$(\mathbf{x} \sim \mathbf{y}) \text{ if and only if } (x_i = x_j) \Leftrightarrow (y_i = y_j) \text{ for all } i, j \in 1, 2, \ldots, n.$$ [1]

[1] For simplicity the limits $i, j \in 1, 2, \ldots, n$ on indices will usually be omitted where, as in this case, they are obvious.

In other words, two n-tuples of observations are the same type of match if they match in the same places as each other and differ in the same places. For example:

$$(1, 2, 2, 4) \sim (5, 1, 1, 4),$$

and

$$(\mathrm{pear}, \mathrm{pear}, \mathrm{apple}) \sim (\alpha, \alpha, \eta),$$

but

$$(\circ, \circ, \diamond, \diamond) \not\sim (1, 2, 1, 2).$$

It must now be shown that Definition 4 is, in fact, an equivalence relation (reflexive, symmetric and transitive).

Reflexive: [$\mathbf{x} \sim \mathbf{x}$] follows immediately since clearly $(x_i = x_j) \Leftrightarrow (x_i = x_j)$.

Symmetric: [$(\mathbf{x} \sim \mathbf{y}) \Rightarrow (\mathbf{y} \sim \mathbf{x})$] follows by assuming the converse. If $\mathbf{x} \sim \mathbf{y}$ and $\mathbf{y} \not\sim \mathbf{x}$ then there exists some i and j where $y_i = y_j$ but $x_i \neq x_j$, a contradiction if $\mathbf{x} \sim \mathbf{y}$.

Transitive: [$\mathbf{x} \sim \mathbf{y}$ and $\mathbf{y} \sim \mathbf{z}$ together imply $\mathbf{x} \sim \mathbf{z}$] follows because if $\mathbf{x} \sim \mathbf{y}$ and $\mathbf{y} \sim \mathbf{z}$ for all i and j then $x_i = x_j$ implies $y_i = y_j$ which in turn implies $z_i = z_j$. The same chain of reasoning means that $z_i = z_j$ implies $x_i = x_j$ and therefore the relationship is transitive.

2.2 The Set Of All Types Of Match, $\mathcal{M}_n$

An obvious next question to ask is "For n sites, how many *types of match* exist?" To answer this question, consider the equivalence relation given by Definition 4 as a partition of the set of all possible n-tuples. A *transversal* is a set containing one and only one representative for each partition. This *transversal* will be referred to as $\mathcal{M}_n$ and by definition has the properties that no distinct members of $\mathcal{M}_n$ are equivalent under Definition 4 but any n-tuple is equivalent to some member of $\mathcal{M}_n$. The notation $\mathbf{x}_n^{\mathcal{M}}$ will be used to designate n-tuples which are members of $\mathcal{M}_n$.

Definition 5. An n-tuple $\mathbf{x}_n^{\mathcal{M}} = (x_1, \ldots, x_n) \in \mathcal{M}_n$ if and only if $x_i \in \mathbb{N}$ and:

$$x_i \in \begin{cases} 1 & i = 1 \\ x_j \text{ for some } j < i & i > 1 \quad \text{or} \\ 1 + \max_{j<i}(x_j) & i > 1 \end{cases}$$

Theorem 1. The set $\mathcal{M}_n$ of all possible $\mathbf{x}_n^{\mathcal{M}}$ meeting the conditions of Definition 5 is a transversal of the set of all possible n-tuples partitioned by the equivalence relation in Definition 4.

Proof. It is necessary to establish two things:

1. For any n-tuple $\mathbf{x}$ there exists some $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$.
2. No two distinct elements of $\mathcal{M}_n$ are equivalent.

To prove the first part define a procedure to calculate $\mathbf{y}_n^{\mathcal{M}}$ from $\mathbf{x} = (x_1, \ldots, \mathbf{x}_n)$ such that $\mathbf{y}_n^{\mathcal{M}} \sim \mathbf{x}$. Such a procedure is defined in Table 1.

1. Set $y_1 = 1$.
2. Set r to 2.
3. If $x_r = x_i$ where $(i < r)$ then $y_r = y_i$
4. Otherwise $y_r = \max_{i<r}(y_i) + 1$
5. If $r < n$ then increment r and go back to step 3.

Table 1: Procedure for forming $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$.

This procedure will create some n-tuple $\mathbf{y}_n^{\mathcal{M}}$ given an n-tuple $\mathbf{x}$. It remains to prove that $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ and $\mathbf{y}_n^{\mathcal{M}} \sim \mathbf{x}$. Since $y_1 = 1$ and either $y_i = y_j$ for some $(j < i)$ or $y_i = \max_{j<i}(y_j) + 1$ then, clearly $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$. It is also clear that if the above procedure is followed $\mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$. From step three in the procedure it must always be true that $(x_i = x_j) \Rightarrow (y_i = y_j)$ and from step four then $(x_i \neq x_j) \Rightarrow (y_i \neq y_j)$. Therefore $(x_i = x_j) \Leftrightarrow (y_i = y_j)$ and so, from Definition 4, $\mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$.

For the second part of the proof, it must be shown that no two distinct elements of $\mathcal{M}_n$ are equivalent. Or alternatively, that if two elements of $\mathcal{M}_n$ are equivalent then they must also be equal. That is, for all $\mathbf{x}_n^{\mathcal{M}}, \mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ then $(\mathbf{x}_n^{\mathcal{M}} \sim \mathbf{y}_n^{\mathcal{M}}) \Rightarrow (\mathbf{x}_n^{\mathcal{M}} = \mathbf{y}_n^{\mathcal{M}})$.

If $\mathbf{x}_n^{\mathcal{M}} \neq \mathbf{y}_n^{\mathcal{M}}$ then there must be some earliest element r of the n-tuples at which they differ. Therefore, define r as the earliest element of $\mathbf{x}_n^{\mathcal{M}}$ such that $x_r \neq y_r$. Assume without loss of generality that $x_r < y_r$. By Definition 5, either $y_r = y_i$ for some $i < r$ or $y_r = \max_{i<r}(y_i) + 1$.

In the first case, $y_r = y_i$, however, $x_r \neq y_r$ (by the definition of r) and therefore, since $y_i = x_i$ and $x_r \neq x_i$ by Definition 4, $\mathbf{x}_n^{\mathcal{M}} \not\sim \mathbf{y}_n^{\mathcal{M}}$.

In the second case, $y_r = \max(y_i) + 1$. Since $x_r \neq y_r$, it is clear that there is some element x_i with $(i < r)$ such that $x_r = x_i$ but $y_r \neq y_i$ and therefore $\mathbf{x}_n^{\mathcal{M}} \not\sim \mathbf{y}_n^{\mathcal{M}}$.

Therefore it has been proved that, if element r exists, the two classes are not equivalent. If there is no such element r then obviously $x_i = y_i$ for all i and $\mathbf{x}_n^{\mathcal{M}} = \mathbf{y}_n^{\mathcal{M}}$. □

The procedure defined by Table 1 can be thought of as a map from the set of all possible n-tuples to the set $\mathcal{M}_n$. An example of this map in use is:

$$(\circ, \Box, \circ, \diamond, \diamond) \mapsto (1, 2, 1, 3, 3).$$

Thus it has been shown that $\mathcal{M}_n$ in Definition 5 is a transversal of the equivalence classes in definition 4 for all n-tuples. Table 1 defines a procedure which will convert any n-tuple of observations $\mathbf{x}$ into $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n : \mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$.

Definition 6. The *matching class* of an n-tuple $\mathbf{x}$ is the member of $\mathcal{M}_n$ to which it is equivalent. That is, the matching class of an n-tuple $\mathbf{x}$ is $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n : \mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}}$.

Definition 7. The height $H(\mathbf{x}_n^{\mathcal{M}})$ of an n-tuple $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$ is the value of its maximal element:

$$H(\mathbf{x}_n^{\mathcal{M}}) = \max(x_i).$$

Definition 8. A *true match* $\mathcal{M}_n(\mathcal{T})$ is the member of $\mathcal{M}_n$ with height 1. That is, $\mathcal{M}_n(\mathcal{T}) = (1, 1, \ldots, 1)$. This represents an observation of the same individual at every one of n sites. A *false match* $\mathcal{M}_n(\mathcal{F})$ is the member of $\mathcal{M}_n$ with height n. That is, $\mathcal{M}_n(\mathcal{F}) = (1, 2, \ldots, n)$. This represents an observation of n different individuals, one each at every one of n sites.

2.3 Mapping $\mathcal{M}_n$ to the set of partitions of the first n integers

A partition of the first n integers is a set $\mathcal{P}$ of non-empty sets Y_i (that is $\mathcal{P} = \{Y_1, \ldots, Y_m\}$) where each of the first n integers is a member of one and only one of the sets Y_i. Call the set of all possible such partitions of the first n integers $\mathcal{P}_n$.

Theorem 2. The set $\mathcal{M}_n$ has the same number of elements as the set P_n, the set of all possible partitions of the first n integers.

This is proved in (Clegg 2003b).

It is well-known (see (van Lint and Wilson 2001, pages 119–128)) that the number of members of $\mathcal{P}_n$ can be counted using Bell Numbers and Stirling numbers of the second kind.

2.3.1 Bell Numbers and Stirling Numbers of the Second Kind (enumerating $\mathcal{M}_n$)

Definition 9. Stirling numbers of the second kind are defined by the recursive relationship:

$$S(n,k) = \begin{cases} kS(n-1,k) + S(n-1,k-1) & n > 0 \text{ and } 0 < k \leq n \\ 1 & n = k = 0 \\ 0 & \text{otherwise.} \end{cases}$$

Definition 10. The Bell numbers $B(n)$ are given by:

$$B(n) = \sum_{k=1}^{n} S(n,k) \text{ for all } n > 0.$$

Theorem 3. Given the definitions of $S(n,k)$ and $B(n)$ above:

1. The total number of members of $\mathcal{P}_n$ which are partitions into k sets is given by $S(n,k)$.
2. The total number of members of $\mathcal{P}_n$ (and therefore $\mathcal{M}_n$) is given by the Bell number $B(n)$.

Proof. The first part is proved in (van Lint and Wilson 2001, page 125). The second part follows from the fact that the Bell numbers are the sum over all possible Stirling numbers for a given n and the already established fact that $\#\mathcal{M}_n = \#\mathcal{P}_n$. □

3 Inducing a Partial Ordering on the Set $\mathcal{M}_n$

A useful *partial ordering* can be induced on the set $\mathcal{M}_n$ as follows:

Definition 11. For two n-tuples $\mathbf{x}_n^{\mathcal{M}}, \mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ a partial ordering relation $\succsim$ can be defined by:

$$\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}} \text{ if and only if } (x_i = x_j) \Rightarrow (y_i = y_j).$$

To be a partial ordering, the relation must be reflexive, anti-symmetric and transitive and, again, these properties are easily proved.

Refexive [$\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{x}_n^{\mathcal{M}}$ for all $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$]: This is trivially true since $x_i = x_j \Rightarrow x_i = x_j$.

Anti-Symmetric [$\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}}$ and $\mathbf{y}_n^{\mathcal{M}} \succsim \mathbf{x}_n^{\mathcal{M}}$ together imply $\mathbf{x}_n^{\mathcal{M}} = \mathbf{y}_n^{\mathcal{M}}$ for all $\mathbf{x}_n^{\mathcal{M}}, \mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$]: This trivially follows since if both conditions together apply then $(x_i = x_j) \Leftrightarrow (y_i = y_j)$ and hence $\mathbf{x}_n^{\mathcal{M}} \sim \mathbf{y}_n^{\mathcal{M}}$ from Definition 4. It has already been shown that this implies $\mathbf{x}_n^{\mathcal{M}} = \mathbf{y}_n^{\mathcal{M}}$.

Transitive [$\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}}$ and $\mathbf{y}_n^{\mathcal{M}} \succsim \mathbf{z}_n^{\mathcal{M}}$ together imply $\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{z}_n^{\mathcal{M}}$ for all $\mathbf{x}_n^{\mathcal{M}}, \mathbf{y}_n^{\mathcal{M}}, \mathbf{z}_n^{\mathcal{M}} \in \mathcal{M}_n$]: This follows since, if $x_i = x_j$ implies $y_i = y_j$ and $y_i = y_j$ implies $z_i = z_j$ then clearly $x_i = x_j$ implies $z_i = z_j$.

Note that this definition is identical to the original equivalence relation in Definition 4 except that the implication is only in one direction. Note also that this partial ordering applies only to members of the set $\mathcal{M}_n$ not to a general n-tuple of observations. This is because the property of anti-symmetry would not hold for general n-tuples for example $(1, 2) \succsim (\alpha, \beta)$ and $(\alpha, \beta) \succsim (1, 2)$ but $(1, 2) \neq (\alpha, \beta)$.

The symbol $\succ$ will be used to mean strictly succeeds. That is $\mathbf{x} \succ \mathbf{y}$ means $\mathbf{x} \succsim \mathbf{y}$ and $\mathbf{x} \not\sim \mathbf{y}$. The symbol $\succ\!\succ$ will be used to mean *immediate successor* that is, if $\mathbf{x} \succ\!\succ \mathbf{z}$ then $\mathbf{x} \succ \mathbf{z}$ but there is no $\mathbf{y}$ such that $\mathbf{x} \succ \mathbf{y} \succ \mathbf{z}$. The symbols $\prec$, $\precsim$ and $\prec\!\prec$ will have their obvious meanings.

Lemma 1. If $\mathbf{x}_n^{\mathcal{M}} = (x_1, \ldots, x_n) \in \mathcal{M}_n$ then $\mathbf{x}_m^{\mathcal{M}} = (x_1, \ldots, x_m)$ is a member of $\mathcal{M}_m$ for all $1 \leq m \leq n$.

Proof. If Definition 5 holds for x_i with $1 \leq i \leq n$ then clearly it holds for x_i with $1 \leq i \leq n$ if $m \leq n$. Therefore, $\mathbf{y}_m^{\mathcal{M}} \in \mathcal{M}_m$. □

This theorem can be thought of as stating that the n-tuple obtained by choosing only the first m members of a matching class is itself a matching class. (Note that this is not the case if the last m members are chosen. For example the last member of $(1, 2) \in \mathcal{M}_2$ is (2) which is not a member of $\mathcal{M}_1$).

Lemma 2. For all $\mathbf{x}_n^{\mathcal{M}}, \mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$, if $\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}}$ then $\mathbf{x}_r^{\mathcal{M}} \succsim \mathbf{y}_r^{\mathcal{M}}$ for all $r \leq n$.

Proof. By Definition 11 then since $(x_i = x_j) \Rightarrow (y_i = y_j)$ for all $i, j < n$ this is also true for all $i, j < r$ if $r \leq n$ by the same reasoning as for the previous lemma. □

3.1 A Consistent Enumeration For the Partial Ordering

Definition 12. A *consistent enumeration* of a partially ordered set S is a real valued function $f(\mathbf{x})$ where $\mathbf{x} \in S$ with the property that, for all $\mathbf{x}, \mathbf{y} \in S$ then $\mathbf{x} \succ \mathbf{y}$ implies $f(\mathbf{x}) > f(\mathbf{y})$.

Theorem 4. The function $H(\mathbf{x}_n^{\mathcal{M}})$ provides a consistent enumeration of $\mathcal{M}_n$.

Corollary 1. If $H(\mathbf{x}_n^{\mathcal{M}}) = H(\mathbf{y}_n^{\mathcal{M}})$ then either $\mathbf{x}_n^{\mathcal{M}} = \mathbf{y}_n^{\mathcal{M}}$ or $\mathbf{x}_n^{\mathcal{M}} || \mathbf{y}_n^{\mathcal{M}}$, where the symbol $||$ means non-comparable, that is neither $\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}}$ nor $\mathbf{y}_n^{\mathcal{M}} \succsim \mathbf{x}_n^{\mathcal{M}}$.

This theorem and its corollary are proved in (Clegg 2003b).

3.2 The Hasse Diagram

A Hasse diagram is a way of visualising a partially ordered set. A Hasse diagram is constructed by plotting a partially ordered set S graphically in such a way that for all $\mathbf{x}, \mathbf{y} \in S$ if $\mathbf{x} \prec \mathbf{y}$ then $\mathbf{x}$ is further to the top of the diagram than $\mathbf{y}$. Further, if $\mathbf{x} \succ\!\succ \mathbf{y}$ then an arrow is drawn from $\mathbf{x}$ to $\mathbf{y}$.

Every Hasse diagram for $\mathcal{M}_n$ will have discrete levels defined by $H(\mathbf{x}_n^{\mathcal{M}})$ (since this has been shown to provide a consistent enumeration) and will have singular upper and lower levels defined respectively by $\mathcal{M}_n(\mathcal{F})$ (the only possible n-tuple in $\mathcal{M}_n$ with height n) and $\mathcal{M}_n(\mathcal{T})$ (the only possible n-tuple in $\mathcal{M}_n$ with height 1). As an example, the Hasse diagram for $\mathcal{M}_4$ is shown in Figure 1.

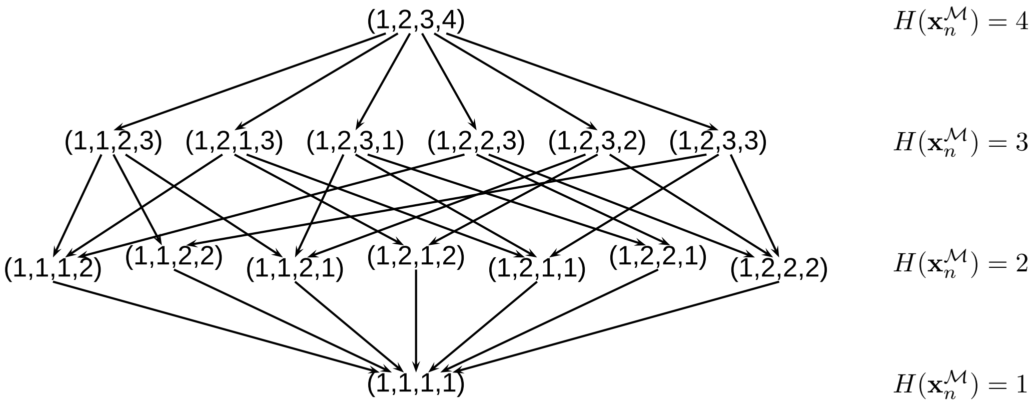
(1,2,3,4)
$H(\mathbf{x}_n^{\mathcal{M}}) = 4$
(1,1,2,3)
(1,2,1,3)
(1,2,3,1)
(1,2,2,3)
(1,2,3,2)
(1,2,3,3)
$H(\mathbf{x}_n^{\mathcal{M}}) = 3$
(1,1,1,2)
(1,1,2,2)
(1,1,2,1)
(1,2,1,2)
(1,2,1,1)
(1,2,2,1)
(1,2,2,2)
$H(\mathbf{x}_n^{\mathcal{M}}) = 2$
(1,1,1,1)
$H(\mathbf{x}_n^{\mathcal{M}}) = 1$

Figure 1: Hasse diagram for $\mathcal{M}_4$.

3.3 Partial (or Censored) Observations Related To Partial Ordering

Definition 13. The *censored observation function*, $C(\mathbf{x})$ is a function which acts on an n-tuple $\mathbf{x} = (x_1, \ldots, x_n)$ (this may be an n-tuple of observations or an n-tuple $\in \mathcal{M}_n$) to produce an n-tuple $\mathbf{y} = (y_1, \ldots, y_n)$ in such a way that if $\mathbf{y} = C(\mathbf{x})$ then:

$$(x_i = x_j) \Rightarrow (y_i = y_j),$$

for all i and j.

The censored observation function is equivalent to the common sense notion of two or more observations of separate individuals which may be confused and appear to be the same individual. An example of a censored observation function would be correctly recording only part of a licence plate. By observing only part of the licence plate the same vehicle can never be recorded differently but different vehicles may be recorded as being the same.

Theorem 5. The matching class of an n-tuple of censored observations $\precsim$ the n-tuple of the original observations. That is, for an n-tuple of observations $\mathbf{z}$:

$$(\mathbf{x}_n^{\mathcal{M}} \sim \mathbf{z} \text{ and } \mathbf{y}_n^{\mathcal{M}} \sim C(\mathbf{z})) \Rightarrow (\mathbf{y}_n^{\mathcal{M}} \succsim \mathbf{x}_n^{\mathcal{M}}).$$

Proof. This follows immediately from the fact that by Definition 4 then $(z_i = z_j) \Leftrightarrow (x_i = x_j)$. By Definitions 4 and 13 $(z_i = z_j) \Rightarrow (y_i = y_j)$. Therefore $(x_i = x_j) \Rightarrow (y_i = y_j)$ which is exactly the condition for the relationship $\mathbf{x}_n^{\mathcal{M}} \succsim \mathbf{y}_n^{\mathcal{M}}$ from Definition 11. □

3.4 The Exact and Relaxed Matching Functions

In this section the exact and relaxed matching functions are introduced and this is used to create an algebra of matching.

Definition 14. The *exact matching function* $X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x})$, where $\mathbf{x}$ is an n-tuple of observations and $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$, is defined as:

$$X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x}) = \begin{cases} 1 & \text{if and only if} \quad \mathbf{x} \sim \mathbf{y}_n^{\mathcal{M}} \\ 0 & \text{otherwise.} \end{cases}$$

This definition can be thought of as an indicator as to whether an observation is a equivalent to a particular matching class. The definition naturally extends from a single n-tuple of observations to a set of n-tuples as shown.

Definition 15. The *exact matching function* $X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{Z})$, (where $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$, all $\mathbf{z}_i$ are n-tuples of observations and $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$) is defined as:

$$X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{Z}) = \sum_{\mathbf{z} \in \mathbf{Z}} X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{z}).$$

In other words, the function counts the number of matches of type $\mathbf{y}_n^{\mathcal{M}}$ in the set of n-tuples $\mathbf{Z}$.

When used on a set of n-tuples, the exact matching functions simply counts the number of matches in a set of observations which belong to the given matching class. The relaxed matching function allows the observations to belong to a matching class or any predecessor of that class.

Definition 16. The *relaxed matching function* $R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x})$, where $\mathbf{x}$ is an n-tuple of observations and $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ is defined as:

$$R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x}_n^{\mathcal{M}}) = \begin{cases} 1 & \text{if and only if} \quad \mathbf{y}_n^{\mathcal{M}} \succeq \mathbf{x}_n^{\mathcal{M}} \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{x}_n^{\mathcal{M}} \sim \mathbf{x}$.

As previously this definition can be extended to a set of n-tuples as shown below.

Definition 17. The *relaxed matching function* $R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{Z})$, (where $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$, all $\mathbf{z}_i$ are n-tuples of observations and $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$) is defined as:

$$R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{Z}) = \sum_{\mathbf{z} \in \mathbf{Z}} R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{z}).$$

In other words, the relaxed matching function counts the number of n-tuples equivalent to a class $\mathbf{y}_n^{\mathcal{M}}$ or any successor class.

3.5 Some Proofs Relating To Exact and Relaxed Matches

It should be clear that the aim of the original problem (to find the number of genuine matches in a data set) is the problem of evaluating $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ where $\mathcal{S}$ is the set of all possible n-tuples of observations from Defintion 3. The problem is complicated by the fact that the observations $\mathcal{S}$ are not available and only the censored observations $C(\mathcal{S})$ are available to work with. All proofs in this section are omitted for reasons of space and can be found in (Clegg 2003b) which is available online.

Lemma 3. Given an n-tuple of observations $\mathbf{x} = (x_1, \ldots, x_n)$ and a matching class $\mathbf{y}_n^{\mathcal{M}} = (y_1, \ldots, y_n) \in \mathcal{M}_n$ these can be reordered without changing the values of the exact and relaxed matching functions. Swapping the elements i and j in both, giving the n-tuple $\mathbf{x}' = (x'_1, \ldots, x'_n)$ and the matching class $\mathbf{y}'^{\mathcal{M}}_n \in \mathcal{M}_n$ such that $\mathbf{y}'^{\mathcal{M}}_n \sim (y'_1, \ldots, y'_n)$ (where $x'_i = x_j$, $x'_j = x_i$ and $x'_k = x_k$ for all $k \neq i, j$ and, in addition, $y'_i = y_j$, $y'_j = y_i$ and $y'_k = y_k$ for all $k \neq i, j$) [2] does not change the value of the exact or relaxed matching functions. That is:

$$X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x}) = X(\mathbf{y}'^{\mathcal{M}}_n, \mathbf{x}'),$$

and

$$R(\mathbf{y}_n^{\mathcal{M}}, \mathbf{x}) = R(\mathbf{y}'^{\mathcal{M}}_n, \mathbf{x}').$$

It should be noted that this lemma also applies to the relaxed and exact matching functions operating on sets of n-tuples when each n-tuple in the set is reordered in the manner described in the lemma.

Lemma 4. Given a set of n-tuples of observations $\mathbf{Z} = \{\mathbf{z}_1, \ldots \mathbf{z}_m\}$ and a matching class $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$ then:

$$X(\mathbf{x}_n^{\mathcal{M}}, \mathbf{Z}) = R(\mathbf{x}_n^{\mathcal{M}}, \mathbf{Z}) - \sum_{\mathbf{y}_n^{\mathcal{M}}} X(\mathbf{y}_n^{\mathcal{M}}, \mathbf{Z}),$$

where the sum is over those elements $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{y}_n^{\mathcal{M}} \prec \mathbf{x}_n^{\mathcal{M}}$.

It is worth noting a trivial corollory of this:

Corollary 2. For all n-tuples $\mathbf{x}$,

$$R(\mathcal{M}_n(\mathcal{T}), \mathbf{x}) = X(\mathcal{M}_n(\mathcal{T}), \mathbf{x})$$

To procede with the theory two definitions are necessary which will be used in the next lemma.

Definition 18. Define $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i)$ as the tuple of indices within $\mathbf{x}_n^{\mathcal{M}}$ which have the value i ordered in increasing value. That is:

$$\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i) = (s(1), s(2), \ldots, s(m))$$

where the $s(j)$ are those elements of $\mathbf{x}_n^{\mathcal{M}}$ such that $x_{s(j)} = i$ ordered such that $s(j) < s(k)$ if $j < k$ and, obviously, m is the number of such elements.

An example of this definition in use may help. If $\mathbf{x}_n^{\mathcal{M}} = (1, 2, 1, 1, 3, 2)$ then $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, 1) = (1, 3, 4)$ (since the first, third and fourth elements of $\mathbf{x}_n^{\mathcal{M}}$ are equal to one). $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, 2) = (2, 6)$ and $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, 3) = (5)$.

[2] Note that the n-tuple $(y'_1, \ldots, y'_n)$ is not necessarily a member of $\mathcal{M}_n$ hence the $\sim$ sign not the $=$ sign.

Definition 19. Define $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ as a set of tuples of observations, derived from $\mathcal{S}$ (the set of all possible n-tuples of observations made over the n sites in $\mathbf{S} = (\mathbf{S}_1, \dots \mathbf{S}_n)$ as given in Definition 3) which is given by the Cartesian product;

$$\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i) = \prod_{\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i)} \mathbf{S}_i,$$

where $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i) = (s(1), s(2), \dots)$ is as given in Definition 18.

In other words, $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ is the set of all possible tuples of observations in those sites picked out by $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i)$. The tuple $\mathbf{X}(\mathbf{x}_n^{\mathcal{M}}, i)$ picks out a selection of sites which have a given index for a given matching class $\mathbf{x}_n^{\mathcal{M}}$ and the set of observations $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ is the set of all possible tuples of observations made at those sites.

Lemma 5. Given a set $\mathcal{S}$ of all possible n-tuples from a set of observations made over n sites (as defined in Definition 3), the number of relaxed matches of class $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$ in $\mathcal{S}$ is given by:

$$R(\mathbf{x}_n^{\mathcal{M}}, \mathcal{S}) = \prod_{i=1}^{h} X(\mathcal{M}_{m(i)}(\mathcal{T}), \mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)),$$

where $h = H(\mathbf{x}_n^{\mathcal{M}})$, $m(i)$ = $\#\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ and $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ is given by Definition 19.

It is worth noting a trivial corollary of this.

Corollary 3.

$$R(\mathcal{M}_n(\mathcal{F}), \mathcal{S}) = \prod_{i=1}^{n} \#\mathbf{S}_i,$$

or, in other words, the number of relaxed matches against the false matching class in a set of observations is simply the number of observations.

Definition 20. For a given censoring function $C(\mathbf{x})$ the probability $p(n)$ is defined for $n \geq 1$ as:

$$p(n) = P(C(\mathbf{x}) \sim \mathcal{M}_n(\mathcal{T}) | \mathbf{x} \sim \mathcal{M}_n(\mathcal{F})),$$

for an n-tuple of observations $\mathbf{x} = (x_1, \dots, x_n)$ where $\mathbf{x}$ is chosen in such a way that $x_i = f(\omega_{j(i)})$ and the $j(i)$ are chosen from the same distribution as the genuine observations in the real data $\mathbf{S}$.

Note that it is implicit in this is the assumption that the distribution of the individuals is not dependent on the site chosen. (Or at least that the probability $p(n)$ does not depend on the particular sites chosen from a subset of sites). This is a reasonable assumption for the particular problem chosen (that of vehicle licence plates). Note also that by this definition then $p(1) = 1$ — this follows from the fact that $\mathcal{M}_1(\mathcal{T}) = \mathcal{M}_1(\mathcal{F})$.

Lemma 6. For a given censoring function $C(\mathbf{x})$ and some n-tuple of observations $\mathbf{x}$ then:

$$P(C(\mathbf{x}) \sim \mathcal{M}_n(\mathcal{T})) = p(h),$$

where $h = H(\mathbf{y}_n^{\mathcal{M}})$ and $\mathbf{y}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{y}_n^{\mathcal{M}} \sim \mathbf{x}$ and $\mathbf{x}$ is randomly chosen in the same manner as in Definition 20. That is, the probability, that after censoring, a set of observations appears to be a true match is $p(h)$.

Lemma 7. For a set of n-tuples of observations $\mathbf{Z}$ with a censoring function $C(\mathbf{Z})$ then an unbiased estimator for the number of true matches in the set of observations can be given by:

$$X(\widehat{\mathcal{M}_n(\mathcal{T})}, \mathbf{Z}) = X(\mathcal{M}_n(\mathcal{T}), C(\mathbf{Z})) - \sum_{\mathbf{x}_n^{\mathcal{M}}} X(\mathbf{x}_n^{\mathcal{M}}, \mathbf{Z}) p(h),$$

where $h = H(\mathbf{x}_n^{\mathcal{M}})$ and the sum is over $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n$ such that $\mathbf{x}_n^{\mathcal{M}} \neq \mathcal{M}_n(\mathcal{T})$.

4 An Algorithm For Estimating False Matches

It is not immediately obvious, but from the above Lemmas 4, 5 and 7 a procedure can be created to estimate $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ — the number of true matches in a set of observations over the set of sites $\mathbf{S}$. This was the original aim of the false match problem in licence plate data.

Lemma 5 allows estimation of $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ from $X(\mathcal{M}_n(\mathcal{T}), C(\mathcal{S}))$ (which can be measured directly since it is measured on the censored data) and $X(\mathbf{x}_n^{\mathcal{M}}, \mathcal{S})$ if it is known for all $\mathbf{x}_n^{\mathcal{M}} \in \mathcal{M}_n : \mathbf{x}_n^{\mathcal{M}} \prec \mathcal{M}_n(\mathcal{T})$. Thus the number of true matches can be estimated from the number of exact matches in all other matching classes.

From Lemma 4 these matches can be calculated exactly if the number of relaxed matches $R(\mathbf{x}_n^{\mathcal{M}}, \mathcal{S})$ is known and also the number of exact matches in all successor matching classes is known.

From Lemma 5 the number of relaxed matches of a particular type can be calculated if the number of exact true matches in a subset of sites is known. The value of $R(\mathcal{M}_n(\mathcal{F}), \mathcal{S})$ is given by corollary 3. From corollary 2, then $R(\mathcal{M}_n(\mathcal{T}), \mathcal{S}) = X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$, which is the quantity desired. For all other values of $\mathbf{x}_n^{\mathcal{M}}$, Lemma 5 allows the calculation of $R(\mathbf{x}_n^{\mathcal{M}}, \mathcal{S})$ in terms of $X(\mathcal{M}_m(\mathcal{T}), \mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i))$ where $m < n$ and $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ is, from Defintion 19, a set of tuples defined over some subset of the original sites. Thus the problem has been reduced to a sub problem of calculating the number of true matches in a subset of sites. This procedure can be followed recursively until the number of sites is 1 when the problem becomes trivial — with one site, $X(\mathcal{M}_1(\mathcal{T}), \mathcal{S}) = X(\mathcal{M}_1(\mathcal{F}), \mathcal{S}) = \#\mathcal{S}$.

Therefore, if $p(n)$ can be estimated, the problem of estimating $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ is solved by the procedure defined in Table 2.

1. Calculate from the data, $X(\mathcal{M}_n(\mathcal{T}), C(\mathcal{S}))$ for all n sites — this is simply a matter of totalling the *true matches* in the censored data.
2. Use a computer to expand Lemmas 4, 5 and 7 to give an expression which estimates $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ as shown above.
3. Again using a computer, gather all the terms which are $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ on the left hand side — these terms will all be functions of $p(k)$ where $1 < k \leq n$.
4. Steps 1 to 3 produce an equation for $X(\mathcal{M}_n(\mathcal{T}), \mathcal{S})$ in terms of $p(k)$, $R(\mathcal{M}_n(\mathcal{F}), \mathcal{S})$ (given by Corollary 3) and $X(\mathcal{M}_m(\mathcal{T}), \mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i))$ where $m < n$ and $\mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$ is the set of tuples of observations over some subset of sites.
5. For each of the terms $X(\mathcal{M}_m(\mathcal{T}), \mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i))$ then if $m = 1$ the answer is trivial. If $m > 1$ then use this whole procedure from step 1 with $n = m$ and $\mathcal{S} = \mathcal{S}(\mathbf{x}_n^{\mathcal{M}}, i)$.

Table 2: Algorithm for correcting false matches.

4.1 Simulation Results

The procedure developed in the previous section has been implemented in C++ and tried both on real data (from roadside surveys) and on simulated data. The simulated data is also presented as if it were a roadside survey. Results on the real data are not presented here since it is impossible to know the correct answer for this data.

Table 4.1 shows simulation results for between two and six observation sites. The table is to be interpreted as follows. Num. Veh. refers to the total number of observations at each of the sites (in these simulations, there are the same number of vehicles in each data set). The five columns of the form $1 – n$ refer to the number of vehicles which genuinely went from site one to site n visiting all sites in between. If this column is blank it means that there was no site n. For example, if $1 – 2 = 100$, $1 – 3 = 200$ and $1 – 4$ is blank. This means that 100 vehicles travelled between site one and site two, 200 vehicles travelled between sites one, two and three and there were only three sites. Note that these are cumulative so that if $1 – 2 = 20$ and $1 – 3 = 10$ this means that 30 vehicles in total went from site one to site two and 10 of them continued to site three. Thus the first experiment is two sites, 1000 vehicles at each for which there were ten vehicles which were genuinely seen at both sites. Note that in every experiment, the number of different vehicle types was set at $10,000$ with a flat distribution (equal numbers of vehicles seen at each site). It should be clear that the desired answer from the correction process is the rightmost figure in these columns.

Each experiment is repeated twenty times with simulated data being generated anew each time. The correction process has no random element and will always give the same result for the same data. The mean raw number of matches is given — this is the total number of n-vector which were seen to have the same value for each observation at every site (averaged over the twenty simulation runs). Note that, because of the combinatorial nature of the procedure, this could, in principle, be much larger than the number of vehicles in any of the data sets (since it counts any n-vector). The sample standard deviation (σ) is given for the raw matches. The mean estimated correct number of matches is then given (again averaged over the twenty simulations). The sample standard deviation σ is then given for the ten corrected matches. It is clear that the most important test is that the mean corrected number of matches is as near to correct as possible. However, it should also be kept in mind that in reality, a researcher could only run the matching procedure once on any given set of data — so it is also important that σ is as low as possible. A significant improvement to the method would be to estimate the variance as well as producing an estimate then the researcher could have some idea as to the likely accuracy of the corrected results. It should also be noted that in every experiment, the chances of any given two vehicles

No. Veh.	1 – 2	1 – 3	1 – 4	1 – 5	1 – 6	Av. Raw Matches	σ Raw Matches	Av. Cor. Matches	σ Cor. Matches
1000	10					111.4	8.5	11.4	8.5
2000	10					411.8	19.5	11.8	19.5
1000	100					199.2	12.0	99.2	12.0
1000	200					302.3	7.7	202.3	7.7
1000	500					596.6	12.3	496.7	12.3
1000	0	10				21.9	4.6	9.3	3.3
1000	500	10				73.8	7.5	10.2	6.2
1000	100	100				152.1	8.5	101.9	7.5
1000	500	250				388.3	22.7	253.2	20.1
1000	0	500				667.2	24.9	506.0	22.3
1000	0	0	100			154.6	26.6	104.0	22.6
1000	100	100	100			164.4	11.4	97.7	9.3
500	100	100	100			140.7	19.3	105.8	17.4
1000	500	250	100			207.8	29.7	106.1	23.7
500	10	10	10	10		14.2	2.2	10.5	1.8
1000	10	10	10	10		17.4	4.1	9.4	2.8
500	50	50	50	50		71.3	14.3	47.8	12.3
500	100	100	100	100		151.9	26.9	92.0	22.3
1000	0	0	0	100		177.6	29.9	103.4	22.6
1000	100	100	100	100		222.2	61.5	111.0	46.7
1000	0	0	0	0	10	21.2	13.4	12.3	9.9
500	0	0	0	0	100	152.6	45.5	92.2	37.3
1000	0	0	0	0	100	214.6	58.0	103.5	40.2
1000	100	100	100	100	100	289.8	88.4	101.3	55.0

Table 3: Simulation results — all performed over twenty runs with 10,000 distinct vehicle types.

being a false match is 1 in $10,000$ with a flat distribution (so the chance of three distinct vehicles having the same partial plate is the square of this). In fact this is an extremely pessimistic assumption since four digits of a licence plate would be the least that a partial plate survey was likely to capture (in the UK, one letter and three digits is the most common). A significant weakness of the method is that it requires a good estimate for $p(n)$. (In fact, it is mainly significant for lower values of n with $p(2)$ being the most important).

The first five rows are all results on just two test sites. This procedure is not the ideal one to use for estimates on matches between just two sites and the work of other authors in the field should be used in such a circumstance. However, these results are included here for completeness. In the two site case, the average corrected matches is simple obtained by subtracting $\frac{n^2}{10000}$ from the raw matches (where n is the number of vehicles at each site) — to take an example, in the first experiment, the average number of raw matches over the ten runs is 111.4. The average number of corrected matches is 100 less than this (11.4). This is close to the correct answer of 10. However, it should be noticed that the σ is high in comparison to the actual answer. In this case, the σ is 8.5 which is of the same order of magnitude as the answer. This is to be expected since we are looking for only 10 true matches in over 110 observed matches. If we increase the number of vehicles to 2000 then, as would be expected, the number of false matches goes up (to approximately 400) and the σ also rises (to almost 20).

The next five rows of results are all over three sites. In the first of these, 10 vehicles travel between all three and all other matches are coincidence. 1000 vehicles are observed at all sites. The mean corrected match across all sites 9.3 is close to the actual answer of 10 and the σ is lower than in the two site case. However, when the same experiment is run with 500 vehicles travelling from sites one to two in addition to 10 vehicles travelling from sites two to three, the σ increases markedly (it almost doubles). In all cases with three sites, the mean is a good estimate and the σ is generally low enough that a good estimate can be expected.

The next four rows of results are for experiments made over four sites. The first experiment has 100 vehicles which visit all four. The mean corrected match is 104 (very close) and the σ is only 22. It is hard to explain why this σ actually falls in the next experiment when more vehicles are genuinely seen in common between the other sites. This fall in σ is puzzling. In all cases the mean of the predictions is approximately correct (the worst performance being in the case of the fourth experiment when the mean was 106 not 100).

The next six rows of results are experiments made over five sites. Again, the mean corrected results are approximately correct. However, in the worst case, the mean is 11 too high and the σ in the result is 46.7 which is comparable to the level of the effect being observed. In this case approximately 120 false matches are being removed each time. However, previous experiments have been able to correct for a greater proportion of false matches with less σ in the result.

The final four rows of results are experiments over six sites. This was the largest number of sites for which it was practical to do runs of twenty or more simulations with the computer power available. Again, the mean corrected estimate of matches was nearly correct in all cases. The worst performance was an estimate of 92.2 (correct result 100). The σ was, however, relatively high. This was a surprise in some cases — particularly the first row of results where the mean number of false matches was only 21.2. In many senses, the worst results was the final one where a σ of 55.0 was given on an corrected prediction of only 101.3.

The time taken to do one run over six sites with one thousand pieces of data on each site was thirty seconds on a Celeron 366 computer running Debian Linux. It is practical (if time consuming) to do experiments on seven sites, even using such comparatively obsolete equipment. However, eight sites or more is probably too computationally expensive for the moment and this is a limitation of the method outlined.

4.2 Summary of Results

The results given here are certainly consistent with the idea that the method gives an unbiased estimator for the true number of matches. In some experiments, there were problems with the standard deviation being higher than would be desirable in real cases. It is important to bear in mind that these were relatively extreme tests of the method since $p(2)$ and $p(3)$ were relatively low and the number of samples given were quite high. Often the method was attempting to predict only ten true matches in a number of observed matches which might be several hundred.

References

Automobile Association: 2003, Automobile Association guide to licence plate formats: www.theaa.com/allaboutcars/drive_plates_history.html.

Clegg, R. G.: 2003a, A freely available data set for modelling day-to-day route choice. Presented at the 2003 Universities Transport Studies Groups. Available online at: http://gridlock.york.ac.uk/route/docs/utsg2003.doc.

Clegg, R. G.: 2003b, A set theoretic framework for multiple data sets with application to the problem of false matches. Available online at: http://gridlock.york.ac.uk/route/docs/opres03.pdf.

Dixon, M. P. and Rilett, L. R.: 2002, Real-time OD estimation using automatic vehicle identification and traffic count data, *Journal of Computer-Aided Civil and Infrastructure Engineering* **17**(1), 7–21.

Halmos, P. R.: 1970, *Naive Set Theory*, Springer-Verlag.

Hauer, E.: 1979, Correction of licence plate surveys for spurious matches, *Transpn. Res. A* **13A**, 71–78.

Jan, O., Horowitz, A. J. and Peng, Z. R.: 2000, Using global positioning system data to understand variations in path choice, *Transpn. Res. Rec.* **1725**, 37–44.

Liu, R.: 2002, Analysis of the uncertainties in day-to-day dynamic models from observed choice responses. Presented at 13th Mini-EURO Conference and 9th Meeting of the EURO Working Group on Transportation, Bari, Italy.
Available online at: http://gridlock.york.ac.uk/route/docs/rliubari.doc.

Maher, M.: 1985, The analysis of partial registration-plate data, *Traf. Eng. & Cont.* **26**(10), 495–497.

Quiroga, C., Henk, R. and Jacobson, M.: 2000, Innovative data collection techniques for roadside origin-destination surveys, *Transpn. Res. Rec.* **1719**, 140–146.

Schaefer, M. C.: 1988, License plate matching surveys: Practical issues and statistical considerations, *Inst. of Trans. Engineers Journal* **July**, 37–42.

Slavik, M. M.: 1985, Errors in origin-destination surveys done by number-plate techniques, *Transpn. Res. Rec.* **1050**, 46–52.

van Lint, J. H. and Wilson, R. M.: 2001, *A Course in Combinatorics: Second Edition*, Cambridge University Press.

Watling, D. P.: 1994, Maximum likelihood estimation of an origin-destination matrix from a partial registration plate survey, *Transpn. Res. B* **28B(3)**, 289–314.

Watling, D. P. and Maher, M. J.: 1988, A graphical procedure for analysing partial registration-plate data, *Transpn. Res. B* **October**.

Watling, D. P. and Maher, M. J.: 1992, A statistical procedure for estimating a mean origin-destination matrix from a partial registration plate survey, *Transpn. Res. B* **26B(3)**, 171–193.

Williams, P. G., Kirby, H. R., Montgomery, F. O. and Boyle, R. D.: 1988, Evaluation of video-recognition equipment for number-plate matching, *Proc. of PTRC Annual Meeting*, Vol. P306, pp. 229–239.