A SET THEORETIC FRAMEWORK FOR ENUMERATING MATCHES IN SURVEYS AND ITS APPLICATION TO REDUCING INACCURACIES IN VEHICLE ROADSIDE SURVEYS

Richard G. Clegg,
Department of Mathematics, University of York,
York, YO10 5DD, United Kingdom
Email: richard@manor.york.ac.uk

1 INTRODUCTION

This paper describes a method for enumerating the ways in which combinations of vehicles can be observed at different survey points. This framework is quite general and can be applied to a variety of problems where matches are to be found in data surveyed at a number of locations (or at a single location over a number of days). As an example, the framework is applied to the problem of false matches in licence plate survey data.

In this paper, a method for representing the possible types of match is outlined using set theory. A method for quickly determining \( M_n \), the set of all types of match over \( n \) survey sites is described and it is shown that the number of types of match can be calculated using Stirling numbers. The method is applied to the problem of correcting survey data for false matches using a simple probabilistic method and an outline is given for a possible extension of a statistical method for the same problem.

2 THE FALSE MATCH PROBLEM IN LICENCE PLATE DATA

It is often the case that on-street traffic surveys collect partial vehicle licence plate information \(^1\). This information can then be used to reconstruct travel times and to infer route information about drivers. In partial plate data, however, problems can occur from false matches — that is to say matches where the section of the licence plate collected matches but the full plate (had it been recorded) would not have matched. Because of the combinatorial nature of the problem, the possible number of false matches could be quite large. With two surveys of a thousand vehicles each, even if the probability of a false match is only one in ten-thousand the expected number of false matches is one hundred. This figure could easily overwhelm the number of true matches.

At multiple sites an apparent match at four survey points may be: a true match (the same vehicle seen at all four points); a different vehicle at each of the four points which (by coincidence) have the same partial plate; a vehicle at survey point one and two which has the same partial plate as a second vehicle at survey points three and four; or any of fifteen total possibilities. The problem becomes more difficult as the number of sites increases.

\(^1\)The reason for collecting partial rather than full licence plate information is that the recording and transcription of the data is often done manually and time constraints would preclude recording a full plate.
A number of researchers have approached the false matching problem for licence plates. [Hauer, 1979] provides an early approach for two sites. [Maher, 1985] describes several methods including the possibility of two point matches between pairs selected from a number of survey sites. [Watling and Maher, 1992] describes a further refinement adding journey time information. However, all of these methods concentrate on matches between pairs of sites. [Liu, 2002] gives a method for three sites and this paper provides suggestions for extending the method to $n$ sites.

3 EQUIVALENCE CLASSES FOR TYPES OF MATCH

Before examining the types of match we must first specify exactly what we mean when we say that two matches are of the same type. Assume that we have $n$ survey sites. At each site $i$ we have a set of unique observations $L_i$ (assume that the complete observation of the entire licence plate is available to use). We will refer to an $n$-tuple of observations with the notation $x_{S_n} = (x_1, x_2, \ldots, x_n)$ and the set of all such $n$-tuples as $S_n$.

We can say that $S_n$ is the cartesian product (or product set) of the sets of observations from each site. That is $S_n = L_1 \times L_2 \times \ldots \times L_n = \prod_{i=1}^{n} L_i$

Take the following observation $n$-tuples made at three sites:

$x_{S_3} = (A123XYZ, B256ABC, B256ABC)$
$y_{S_3} = (A123XYZ, A123XYZ, B256ABC)$
$z_{S_3} = (C789ABC, A543OPQ, A543OPQ)$

it is clear that in some sense $x_{S_3}$ and $y_{S_3}$ are not the same type of match whereas $x_{S_3}$ and $z_{S_3}$ are the same type of match. We would therefore like to express the notion that two observations are of the same type if a match between two sites in the first $n$-tuple is also a match between the two sites in the second $n$-tuple and if two observations in the first $n$-tuple do not do match then they also do not match in the second observations. Formally we express this notion using the concept of an equivalence class (see, amongst others [Halmos, 1970]).

**Definition 3.1.** $x_{S_n} \sim y_{S_n}$ iff $x_i = x_j \Leftrightarrow y_i = y_j \ \forall i, j \in \mathbb{N} : 1 \leq i, j \leq n$ \footnote{Note that for simplicity, such limits on indices will be omitted in future definitions where they are obvious.}

It can be trivially proved that this is an equivalence relation (reflexive, symmetric and transitive). Thus, from our example, above we can now say: $x_{S_3} \sim z_{S_3}$ but $x_{S_3} \not{\sim} y_{S_3}$.

If we can create a set containing exactly one representative from each of these equivalence classes then this set will have one representative for each type of match. Such a set is known as a transversal. Let $M_n$ be a transversal of the equivalence relation defined in 3.1. Let $x_{M_n} = (x_1, x_2, \ldots, x_n)$ be an $n$-tuple which is a member of $M_n$. If we can construct such a set $M_n$, then we have a set of all the different types of matches which can occur over $n$ survey sites.

**Definition 3.2.** $x_{M_n} \in M_n$ iff:

$$x_i = \begin{cases} 1 & i = 1 \\ x_j \text{ or } 1 + \max(x_j) & j < i \end{cases}$$

This is most easily understood as the following procedure: Label the first vehicle with a 1. Label subsequent vehicles either with the number of a previous vehicle which they match or with the next available integer. It is not difficult to prove that:

433
\[ \forall x_{S_n} \exists y_{M_n} \in M_n : x_{S_n} \sim y_{M_n} \]

(that is to say any n-tuple of observations is equivalent to a member of \( M_n \)) and also that:

\[ \forall x_{M_n}, y_{M_n} \in M_n \quad x_{M_n} \sim y_{M_n} \Rightarrow x_{M_n} = y_{M_n} \]

(in other words \( M_n \) contains no duplicates — if two n-tuples are members of \( M_n \) and are equivalent then they must be the same member), thus \( M_n \) is a transversal and has exactly one representative of each type of match. It is worth emphasising that these types of matches would apply to any observations where we can define an equality relation between observations — the framework would be just as applicable to discrete sets of colours or types of animal as it would to vehicle number plates.

To give an example, we can now express our three earlier n-tuples in terms of equivalent members of this matching class.

\[ x_{S_3} = (A123XYZ, B256ABC, B256ABC) = (1, 2, 2) \]
\[ y_{S_3} = (A123XYZ, A123XYZ, B256ABC) = (1, 1, 2) \]
\[ z_{S_3} = (C789ABC, A5430PQ, A5430PQ) = (1, 2, 2) \]

The height of a type of match \( x_{M_n} \) is \( H(x_{M_n}) = \max(x_i) \). It should be clear that the height of a type of match is the number of different observations which are in the n-tuple (the number of unique vehicles observed).

4 CONSTRUCTING THE SET OF EVERY TYPE OF MATCH

The set \( M_{n+1} \) can be easily constructed from the set \( M_n \) using definition 3.2. If \( x_{M_n} \in M_n \) then we can construct \( y_1, y_2, \ldots \in M_{n+1} \) from \( x_{M_n} \) by adding an \( n + 1 \)th element to the n-tuple. From 3.2 we can see that \( y_{n+1} \) (the \( n + 1 \)th element of \( y_1 \)) can take any integer value from 1 to \( H(x_{M_n}) + 1 \) (which is equal to \( \max(x_i) + 1 \).

To construct \( M_{n+1} \) from \( M_n \): a) Take each element of \( M_n \) in turn. b) To each n-tuple \( x_{M_n} \) construct a vector by adding the integers from 1 to \( H(x_{M_n}) \) as the \( n + 1 \)th element of the n-tuple. c) These vectors (\( n+1 \)-tuples) together form the set \( M_{n+1} \). Therefore, given that \( M_1 = (1) \) we can easily construct computationally \( M_n \). This is process is illustrated in diagram 1:

It can be simply shown that there is a one-to-one correspondance between the members of \( M_n \) and the partitions of the set of the first \( n \) natural numbers. Therefore we can count the members of \( M_n \) using Stirling numbers (see [Biggs, 1961] for more information on Stirling numbers).
Let \( S(n, k) \) be the number of members of \( M_n \) with height \( k \) (where \( 1 \leq k \leq n \)). Clearly \( S(n, 1) = 1 \) — the only member with height 1 is \((1,1,\ldots,1)\). Also \( S(n, n) = 1 \) — the only member with height \( n \) is \((1,2,\ldots,n)\).

We can also show that \( S(n, k) = S(n - 1, k - 1) + kS(n - 1, k) \). A sketch of such a proof is that using our constructive method above, every member of \( M_{n-1} \) with height \( k - 1 \) will construct one member of \( M_n \) with height \( k \) and every member with height \( k \) will construct \( k \) members with height \( k \) (by adding the final elements from \( 1\ldots k \)). From this recursive formula we can calculate the number of elements in \( M_n \) of a given height and, by summing, the number of elements in \( M_n \).

5 INTRODUCING FALSE MATCHING INTO THE FRAMEWORK

We introduce the complications caused by false matching using the concept of a partial ordering of a set (see [Halmos, 1970]).

**Definition 5.1.** \( x_{M_n} \preceq y_{M_n} \) 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. These properties are easily proved. Note that this definition is the same as that for the equivalence relation except that the implication only goes one way.

This partial ordering is useful because it relates to the false matching problem. The previous discussion has all assumed that our observations are unique. When we consider the censored data that we get by only observing part of the licence plate then false matches can occur. Let \( y_{S_n} = C(x_{S_n}) \) be the \( n \)-tuple of censored data derived from an uncensored \( n \)-tuple \( x_{S_n} \).

**Theorem 5.1.** If \( y_{M_n} \sim x_{S_n} \) and \( z_{M_n} \sim C(x_{S_n}) \) then \( y_{M_n} \preceq z_{M_n} \).

That is to say, the match type of the censored data is a successor of the match type of the uncensored (complete) data.

This is proved by assuming that if two observations are equal in the complete data then they must also be equal in the censored (partial plate) data. However, two observations which are unequal in the complete data may become equal in the partial data.

We can visualise our transversal \( M_n \) and its partial ordering using a Hasse diagram. A Hasse diagram is a depiction of a finite partially ordered set where the elements are represented by points in a plane and a directed arrow from element \( x \) to element \( y \) indicates that \( x \prec y \). \( (x \) immediately precedes \( y \) — that is \( \exists z : x < z < y \)). The diagram of \( M_n \) has discrete levels defined by \( H(x_{M_n}) \) and will have singular upper and lower levels defined by \( x_{M_n} = (1,2,...,n) \) and \( x_{M_n} = (1,1,...,1) \) respectively. The Hasse diagram for \( M_4 \) is shown in figure 2.

6 SOLVING THE FALSE MATCH PROBLEM

If we want to be able to count the number of vehicles which are seen at all sites then this is the problem of enumerating the number of \( n \)-tuples of observations which are in matching class \((1,1,...,1)\) (this will be referred to as \( M_n(T) \) — a true match across all sites). A coincidental match of partial plate observations can only move observations down arrows on the Hasse diagram.

Let \( p(n) \) be the probability that \( n \) distinct vehicles chosen at random have the same partial licence plate (define \( p(1) = 1 \)). We can then say that if a set of observations is in matching
class $x_{M_n}$ then the probability of this vehicle being observed in matching class $M_n(T)$ is $p(H(x_{M_n}))$ since $H(x_{M_n})$ is, as previously stated, the number of vehicles involved in a match of that type. Therefore, if we know how many n-tuples are in the higher matching classes we can get an estimator for the number of false matches which will be added to $M_n(T)$. These can be estimated by a recursive procedure which considers subsets of sites. Results will be presented showing this procedure correcting false matches on simulated and real data.

[Liu, 2002] describes a statistical model for correcting false matches at three sites. A major difficulty with extending the method to surveys at $n$ sites is that the parameters to be fitted to the model are dependent on the variety of routes which vehicles can take to the survey sites. The framework described in this model could be used to extend this model to $n$ sites by assigning parameters for each of the match types in $M_n$.

References


