A transparent and transportable methodology for evaluating Data Linkage software

Anna Ferrante *,1, James Boyd

Centre for Data Linkage, Curtin University of Technology, Western Australia, Australia

ARTICLE INFO

Article history:
Received 10 June 2011
Accepted 22 October 2011
Available online 30 October 2011

Keywords:
Data matching
Medical record linkage
Software evaluation
Linkage quality

ABSTRACT

There has been substantial growth in Data Linkage (DL) activities in recent years. This reflects growth in both the demand for, and the supply of, linked or linkable data. Increased utilisation of DL "services" has brought with it increased need for impartial information about the suitability and performance capabilities of DL software programs and packages.

Although evaluations of DL software exist; most have been restricted to the comparison of two or three packages. Evaluations of a large number of packages are rare because of the time and resource burden placed on the evaluators and the need for a suitable "gold standard" evaluation dataset.

In this paper we present an evaluation methodology that overcomes a number of these difficulties. Our approach involves the generation and use of representative synthetic data; the execution of a series of linkages using a pre-defined linkage strategy; and the use of standard linkage quality metrics to assess performance. The methodology is both transparent and transportable, producing genuinely comparable results. The methodology was used by the Centre for Data Linkage (CDL) at Curtin University in an evaluation of ten DL software packages. It is also being used to evaluate larger linkage systems (not just packages). The methodology provides a unique opportunity to benchmark the quality of linkages in different operational environments.

© 2011 Elsevier Inc. All rights reserved.

1. Introduction

1.1. Data Linkage-based research

Data Linkage (DL) methods are being used increasingly in health and human services research sector. Typically, these methods bring together administrative data from disparate sources and link them through various approaches (e.g., probabilistic, deterministic and/or fuzzy logic methods). The end product is a linked dataset which is used to study individuals and their health outcomes. A critical feature of many Australian linked datasets is that, once assembled, they are stripped of name-identifying information so that researchers work only with de-identified data.

There are a number of advantages in using linked data of this kind. Most importantly, they allow study of large, whole-population samples and extensive longitudinal research; they are relatively time- and cost-efficient; and have reduced methodological problems relating to loss-to-follow-up, recall, selection, response and reporting bias [1]. However, DL methods also have disadvantages. Most significantly, they use administrative data which were not collected for the purposes of research but rather for delivering government services and monitoring performance and expenditure.

Notwithstanding, DL methods have facilitated an array of health and health related research such as studies of the prevalence and incidence of chronic diseases, studies of the risk factors associated with such illnesses, assessments of health service utilisation, and evaluations of the impact of clinical treatments and health service provision on health outcomes [2–4]. This research has, in turn, led to improvements in patient care, reforms in health policy and law, improvements in the cost-efficiency of research, as well as preservation of privacy, community development, and commercial and competitive benefits [5,6].

1.2. DL infrastructure

Infrastructure enabling routine, population-based DL activity exists in only a handful of countries. In Australia, the Western Australian data linkage infrastructure (WADLS) was started in 1995 by the University of Western Australia’s School of Population Health, working closely with the Western Australian Department of Health. The infrastructure enables probabilistic person links to be created and maintained between the state’s population-based data collections. High quality, linkable, anonymised datasets are...
provided to specified users for approved research projects [2]. The Centre for Health Record Linkage (CHeReL) was established more recently in New South Wales, another Australian state. Both of these systems rival similar international operations such as the Oxford Record Linkage System and the Scotland Medical Record Linkage System in the UK, and the Manitoba Centre for Health Policy and the British Columbia Linked Health Database in Canada. A common element of these operations is that they are “production enterprises”, meaning that they are continuously engaged in routine linkage of large, population-level administrative datasets to service a broad research base.

1.3. Expanding capabilities

The demand for DL services is expanding. In Western Australia, the number of DL-based studies supported by the WADLS grew from approximately 87 projects between 1995 and 1999 to over 308 by 2003–2007 [7]. This kind of research is set to grow further through investment by Australian governments in the Public Health Research Network (PHRN). Funded through the National Collaborative Research Infrastructure Strategy [8], the PHRN has been allocated over $50 million to establish DL infrastructure nationwide. This infrastructure includes six State-based nodes (each responsible for conducting data linkage at State/Territory level), a national Centre for Data Linkage (CDL) and a Program Office. The overall vision is to ‘improve the population health through seamless supply of linked, de-identified data for approved research’ [9].

1.4. Need for information about data linkage software

With increased utilisation of data linkage “services” comes the need for information about the suitability and reliability of data linkage software products. The number of software packages available to undertake DL activities has increased substantially in recent years. There is, however, little information available to guide the selection of software. Empirical evidence of the linkage performance of proprietary data linkage programs is scant [10]. Although evaluations of data linkage software exist, most have been restricted to the comparison of a limited set of packages – typically, two or three products (see, for example, Herzog et al. [11] and Campbell et al. [10]). Evaluations of a large number of packages are rare because of the time and resource burden placed on the evaluators and, additionally, because of the need for a suitable “gold standard” evaluation dataset [12].

1.5. Evaluation datasets

Publicly available, real world datasets for data linkage which can be used as test decks for comparison and evaluation are rare [13]. “Gold standard” evaluation datasets are both difficult to source (as they tend to be based on previously linked datasets where the quality of the linkages is known to be high) and virtually impossible to share (since disclosure of personally identifying information would breach privacy laws). As a consequence, linkage software evaluations tend to be intensive in-house operations that cannot be easily replicated or shared.

1.6. Purpose of this paper

In this paper we present an evaluation methodology that overcomes some of the difficulties in undertaking an evaluation of data linkage software and systems. Our approach involves the creation and use of synthetic but representative datasets; the execution of a series of linkages with a pre-defined linkage strategy; and the use of standard metrics to assess performance. The methodology is both transparent and transportable. The evaluation data and method can be applied to any linkage package or system, be undertaken by any reviewing group, and can be used to produce linkage quality results that are genuinely comparable. The methodology was used by the CDL at Curtin University in an evaluation of ten data linkage software packages [14]. Some of the findings from that evaluation are presented.

2. Methodology

In this section we outline the various components of our methodology. These comprise: (i) the creation and use of synthetic datasets, (ii) the specification of a linkage plan with a pre-defined linkage strategy, and (iii) the use of standard linkage quality metrics to assess performance.

2.1. Creation and use of synthetic datasets

Since “gold standard” datasets are both difficult to source and virtually impossible to share, we opted to create and use synthetic datasets. Such datasets can be created using purpose-built data generation programs. For our purposes, we selected the probabilistic data generation programme that was developed and implemented as part of the open-source FEBRL data linkage system [15]. The generator was originally developed in 2005 [13] and is based on ideas by Hernandez and Stolfo [16]. It is argued to be an improvement on other generators such as the UIS Database Generator [17] and the generator by Bertolazzi and colleagues [18].

The FEBRL data generator [13] creates data sets that can contain names and addresses, dates, telephone and identifiers (e.g., social security number). As a first step, the generator creates a user-specified number of original records. These are created randomly, based on frequency lookup tables. Duplicate records are created in a second step, based on the original records. Duplicate records are created by randomly selecting an original record, then randomly choosing the number of duplicates to be created from it, and then randomly introducing errors according to user-specified parameters (probabilities). An additional probability distribution specifies how likely data items or attributes are selected for introducing errors (it is possible for data items to have no errors at all).

As part of our methodology, we generated datasets that were suitably representative (i.e. based on real world frequency and error distributions) and of sufficient size to enable realistic testing of the run-time performance and linkage quality of each package.

As per the FEBRL generator approach, generation of synthetic data was broken into two stages: (i) creation and use of a large, representative version of the population; and (ii) generation of duplicate records with errors (in our case, synthetic morbidity and mortality records) based on this population (see Fig. 1). Final datasets comprised:

- A population file, containing 4 million records (1 record per person). The file was based on frequency distributions obtained from the Western Australian electoral roll. Note that in Australia, voting at national and state level is compulsory. Hence, electoral rolls are highly representative of the adult population. To avoid the potential of identifying individuals from the electoral data, the frequency list was truncated so that frequency counts below five were excluded.
- A morbidity file simulating hospital admissions (and re-admissions) for a random sample of persons selected from the population file (each person included in the morbidity file could have up to six admissions). The full morbidity file contained 400,000 simulated hospital admissions.
Our methodology specified that two blocking strategies be used. Block 1 comprised Soundex of the NYSIIS code of the surname plus first initial of first name. The phonetic encoding of names using both NYSIIS and Soundex phonetic codes follows the convention set by the Oxford Names Compression Algorithm (ONCA) used at OX-Link [25]. Block 2 comprised all elements of date of birth (day, month, year). Records were, therefore, not compared if they disagreed on one or more of the first set of blocking items and also disagreed on one or more of the second set of blocking items. Under this strategy it is possible that two records belonging to the same person will disagree on both blocks. Thus, under these circumstances, a small proportion of true links will be lost through blocking.

2.2.2. Comparison specification
Our methodology specified that pairs of records be compared as follows:

Surname – Approximate string comparison (Jaro-Winkler method) or truncated string comparison (1st eight letters); Also, phonetic (NYSIIIS) comparison.
First name – Truncated string comparison (1st four letters). First initial – Exact comparison.
Date of birth – Date comparison, allowing some difference in month and day.
Sex – Exact comparison.
Postcode – Exact comparison.

2.2.3. Weight
Our methodology did not define or specify weights to be used in linkage, as there are considerable variations in the implementation of weighting by the various software packages. It was decided that control of this step be left to the software package and/or user.

2.2.4. Setting thresholds & classification of pairs
Our methodology did not specify threshold values (this was again left to the control of software package and/or user). However, our methodology specified that ‘possible’ matches were prohibited. In other words, our methodology specified that upper and lower thresholds should be set at the same value and that large scale clerical review of potential or possible matches should not be undertaken. In this way, the evaluation methodology would test the linkage capabilities of the software only.

2.3. Specification of file linkages
The next component of our methodology consisted of specifying the types of linkages to be undertaken (i.e. de-duplication and/or file-to-file linkage) using the linkage strategy described above. A set of linkages were proposed:

- A de-duplication (or internal linkage) of Morbidity_10percent, i.e. identify all possible duplicate records within the 10% sample morbidity file (40,000 records).
- A de-duplication (or internal linkage) of Morbidity_25percent, i.e. identify all possible duplicate records within the 25% sample morbidity file (100,000 records).
- A de-duplication (or internal linkage) of Morbidity_full, i.e. identify all possible duplicate records within the full morbidity file (400,000 records).
- A file-to-file linkage of Morbidity_full to Population_file, i.e. attempt to link the full morbidity file (Morbidity_full; 400,000 records) to the population file (4 million records).

The linkages were designed to be progressively more complex and to place an increasingly larger load on computer resources.

2.4. Specification of run-time performance statistics and linkage quality metrics
The next component of our methodology consisted of specifying run-time statistics and linkage quality metrics. Run-time statistics were specified as the number of hours, minutes and seconds required to complete each of the four linkages specified above. Linkage quality metrics were drawn from the range of quality measures.
used in record linkage. (For a good description of these, see both Bishop and Khoo [26] and Christen and Goiser [27].)

In assessing the quality of a linkage, primary interest is in knowing how many true matched and non-matched records are identified or returned. True matches and true non-matches are not usually known prior to a linkage. However, as the datasets used in our methodology were synthetically generated, it was possible to flag which morbidity records were sourced from, or belonged to, specific population records. In this way it was possible to know all true matches and non-matches a priori.

In terms of quality metrics, our preference was to use three standard metrics – precision, recall and \( f \)-measure.

**Precision** refers to the proportion of returned matches that are true matches. It is sometimes referred to as **positive predictive value** and is measured as:

\[
\text{Precision} = \frac{\text{Number of true positives}}{\text{Number of true positives} + \text{Number of false positives}}.
\]

where a true positive is a pair of correctly matched records, and a false positive is one that is incorrectly or falsely matched. False positives are pairs of records that have been falsely linked (i.e. brought together through linkage but actually belong to different people).

**Recall** is the proportion of all true matches that have been correctly identified. Recall is also known as sensitivity and is measured as:

\[
\text{Recall} = \frac{\text{Number of true positives}}{\text{Number of true positives} + \text{Number of false negatives}}.
\]

where a true positive is a pair of correctly matched records, and a false negative is a missed match, i.e. a pair of records that should have been linked because they belong to the same person but were not. False negatives or Type II errors are difficult to estimate in real world situations.

**\( f \)-Measure** is the harmonic mean of precision and recall and is calculated as:

\[
\text{\( f \)-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.
\]

\( f \)-Measure has a high value when both precision and recall have high values; however, there is an underlying trade-off between precision and recall (when one is high, the other is invariably lower). The \( f \)-measure is thus seen as a way of finding the best compromise between the two metrics.

### 3. Application of the methodology: results from the Centre for Data Linkage (CDL) software evaluation

To demonstrate the utility of the methodology, we present the results from a recent evaluation. The evaluation used the methodology to evaluate the performance of ten data linkage software packages. The evaluation was conducted in order to inform decision making on the most appropriate choice of software for production-level DL enterprises by the CDL and by other participant organisations of the Population Health Research Network [14].

To summarise, the evaluation shortlisted ten DL packages. These included Bigmatch, dIPowerStudio, FEBRL, FRL, HDI, LinkageWiz, LINKS, QualityStage, The Link King and a programme developed in-house based on the Scottish Record Linkage System. Most of the packages implemented probabilistic matching [20,28]; however, a small number used deterministic processes.
Deterministic matching systems use a rules-based approach to determine when two or more records match. The algorithms sometimes use fuzzy matching logic to decide whether records are matched. The algorithms account for common errors such as typographical errors, phonetic variations and transpositions. The packages were evaluated on the same configuration of hardware so that run-time performance could be fairly compared. A review of the functionality and features of each software package was also undertaken, using a template based on a checklist developed by Day [29] (described in detail in Chapter 18 of Herzog et al. [11]).

Each software package was used to undertake four linkages, using the linkage strategy and synthetic datasets (all described above). In the case of software packages using a deterministic matching protocol, the linkage strategy was adapted such that blocking strategies and field comparisons were converted to rule-based equivalents.

3.1. Sample of evaluation results – run-time performance

The run-times for each software package were recorded for the various types of linkages undertaken. Software packages were ranked on performance times (a sample is provided in Table 2). Note that for the purposes of demonstrating the methodology, we have anonymised the results for each package. A fully identified report of package performance is available on request. Speed of execution was classified into three categories: fast, moderate or slow, depending on the relative performance of each software package on the same hardware.

The speed classifications across the different types of linkage were then combined to produce a single, overall speed rating for each software package (as per Table 3).

3.2. Sample evaluation results – linkage quality

Linkage Quality (LQ) measures for each software package were recorded for each of the linkage runs. Software packages were subsequently ranked on LQ metrics. For each linkage, cut-off levels were set where the f-measure was maximised. Packages were graded on the basis of their relative maximum f-measure: packages with maximum f-measure scores \( \geq 0.90 \) were considered ‘very good’, those with 0.85 \( \leq \) max f-measure < 0.90 were considered relatively ‘good’, while those with 0.80 \( \leq \) max f-measure < 0.85 were rated as ‘fair’ (a sample of results is provided in Table 4).

Results showed that f-measures did not vary greatly across software packages or types of linkage. This suggested significant robustness in the matching methods implemented by most, if not all, of the packages included in the evaluation.

The LQ results for each package were then combined across the different types of linkage to produce a single, overall LQ ranking (as demonstrated in Table 5).

---

Table 2

<table>
<thead>
<tr>
<th>Software</th>
<th>Run-time</th>
<th>Rank</th>
<th>Relative speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of linkage: De-duplication of full morbidity file 400,000 records</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Package 1</td>
<td>&lt;5 min</td>
<td>1</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 2</td>
<td>&lt;5 min</td>
<td>2</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 5</td>
<td>&lt;5 min</td>
<td>3</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 4</td>
<td>&lt;1 h</td>
<td>4</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 3</td>
<td>&lt;1 h</td>
<td>5</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 8</td>
<td>&lt;1 h</td>
<td>6</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 6</td>
<td>&lt;1 h</td>
<td>7</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 9</td>
<td>&lt;3 h</td>
<td>8</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 10</td>
<td>&lt;3 h</td>
<td>9</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 7</td>
<td>&lt;3 h</td>
<td>10</td>
<td>Slow</td>
</tr>
<tr>
<td>Type of linkage: File-to-file, full morbidity-to-population 400,000 to 4 million records</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Package 1</td>
<td>&lt;1 h</td>
<td>1</td>
<td>Very Fast</td>
</tr>
<tr>
<td>Package 2</td>
<td>&lt;3 h</td>
<td>2</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 5</td>
<td>&lt;3 h</td>
<td>3</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 6</td>
<td>&lt;3 h</td>
<td>4</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 4</td>
<td>&lt;10 h</td>
<td>5</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 3</td>
<td>&lt;10 h</td>
<td>6</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 8</td>
<td>&lt;10 h</td>
<td>7</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 9</td>
<td>&lt;20 h</td>
<td>8</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 10</td>
<td>&lt;20 h</td>
<td>9</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 7</td>
<td>&lt;20 h</td>
<td>10</td>
<td>Slow</td>
</tr>
</tbody>
</table>

Table 3

<table>
<thead>
<tr>
<th>Software</th>
<th>Type of linkage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Overall speed</td>
</tr>
<tr>
<td></td>
<td>Small de-duplication (40,000 records)</td>
</tr>
<tr>
<td>Package 1</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 2</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 5</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 6</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 4</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 3</td>
<td>Fast</td>
</tr>
<tr>
<td>Package 8</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 7</td>
<td>Moderate</td>
</tr>
<tr>
<td>Package 9</td>
<td>Slow</td>
</tr>
<tr>
<td>Package 10</td>
<td>Slow</td>
</tr>
</tbody>
</table>

---

The template is available on request.
Finally, ratings for both speed and LQ were brought together into a total rating. Overall, two packages (Packages 5 and 1) performed better than the others. These were subsequently used in a Proof of Concept project to further test performance and functionality, using larger-sized, real world datasets.

4. Discussion

4.1. Strengths of the methodology

As evidenced above, the methodology has a number of strengths. Above all, it is a transparent methodology – using a pre-defined linkage strategy, a set of open and shareable datasets, and a set of well-defined, established performance metrics. The approach is also robust – adopting a systematic approach to testing (starting with a small-sized de-duplication and moving to larger file-to-file linkages; running on standard hardware configuration) and realistic strategies to perform an evaluation.

The most significant strength of the methodology is the use of representative but artificial data. This makes the entire approach highly portable – the method can be picked up and used at any time, by any reviewing group, be applied to any DL software package and returning results that are genuinely comparable.

The methodology can also be adapted and extended. With small modification, the methodology can be used to evaluate larger DL systems such as those implemented as part of large production DL infrastructure. This application of the methodology provides an opportunity to benchmark the linkage quality of different DL operations. The methodology may also have the potential to assess the inter-rater reliability of linkage officers. Assessment of both of these applications of the methodology is currently underway.

4.2. Limitations

The evaluation methodology is not without shortcomings, however. There are obvious limitations around the use of synthetic data. The methodology presented is sound for standardised comparison; however, the validity of the comparative results are difficult to gauge given the artificial nature of the data. One way to overcome this problem and make the methodology more robust might be to include performance of each of the software packages on a real world dataset that is of size amenable to manual (human) evaluation. This would indirectly allow evaluation of the synthetic datasets themselves, in terms of their suitability for checking the performance of DL software. However, as discussed...
earlier, there are challenges in this approach and suitable data cannot always be obtained. Ironically, the CDL software evaluation project originally sought to use real world data in the evaluation; however, a request to use previously linked data was refused by an ethics committee on the grounds that the benefit of the research (software evaluation) did not significantly outweigh the risks to privacy (through the release of named data). As a consequence, the evaluation was limited to the use of synthetic data only. Extending the evaluation to incorporate the use of real data has been included into the next phase of the project and results will be reported in the future.

Another limitation of the methodology lies in the approach used to create the synthetic datasets. The paper describes our effort to make the datasets as representative as possible – not only in terms of matching the characteristics of a real world population, but also in terms of matching the types and quantity of errors typically found in real world data. ASCertaining representative rates of different types of errors was a challenging but not arbitrary process and, in our case, involved abstracting errors manually from real data and applying these to the artificial data using features of the FEBRL data generator. The approach produced synthetic datasets with errors that are verifiably typical of those found in real administrative data.

It is difficult to compare the synthetic data with studies using different identifying variables and without some assessment of the quality of the underlying evaluation datasets. However, in a study of three linkage methods which provided similar IQ metrics, the estimated precision rates ranged from 0.95 to 0.97 [10] and recall (sensitivity) rates from 0.79 to 0.94 [10]. The reported precision rates in our full de-duplication linkage were similar (ranging from 0.84 to 0.97); however, the recall rates were lower.

The linkage strategy defined as part of the methodology may also be argued to be limited or limiting. For instance, during the CDL software evaluation, it was found that some packages ran poorly when implementing the defined strategy, yet operated at significantly greater speed when alternative (internally optimised) settings were used. Some other software packages were not able to adhere strictly to the defined blocking and comparison strategies (as these were hard-wired in the software and could not be altered by users). Therefore, it may be argued that the evaluation strategy is unnecessarily restrictive of the performance of some packages.

A further limitation of the methodology, which arose during the CDL software evaluation, concerns the setting of thresholds and the difficulty of making final cut-off decisions in a relatively artificial context. The methodology overcomes this issue by setting cut-offs at a level where the F-measure is maximised. This procedural method for setting thresholds is well-suited to the task of software evaluation; however, the approach differs from the methods more commonly used to determine cut-off points in day-to-day linkage activity. These methods often include a manual review of matches on or near the cut-off point and localised decision making around acceptable levels of false positives (Type I errors).

Another potential limitation of the linkage strategy is the reliance on a single cut-off and the absence of any clerical review of possible matches. While this strategy may reduce the overall quality of linkage, it was a strategy applied to all packages and so maintains consistency within the evaluation methodology.

Without doubt, there is scope to assess and potentially improve upon the evaluation methodology presented here. One way of doing this would be to apply an alternative linkage strategy to the same data, while keeping the software unchanged. Any change in performance could thus be attributed to a different linkage strategy. In fact, this approach has been incorporated into the next phase of the evaluation and results are expected to be reported in the near future.

5. Conclusion

The methodology presented here attempts to overcome some of the limitations that have been experienced in previous DL software evaluations. Application of the methodology should facilitate easier and more comparable evaluations in the future. This should assist in assessing the performance of linkage operations and in the decision making regarding choice of linkage software.

Acknowledgments

The CDL software evaluation described in the paper would not have been possible without the cooperation of the many software suppliers who generously made their packages available for review. We also wish to thank Maxine Croft of Maximal Computer Solutions who assisted in the evaluation and the Western Australian Data Linkage Board for the provision of frequency data from the electoral roll. Finally, we thank the anonymous reviewers who, through their constructive comments, have improved this paper greatly. This project is supported by the Australian Government National Collaborative Research Infrastructure Strategy’s Population Health Research Network.

References


