Bayesian Analysis (0000) **00**, Number 0, pp. 1

Efficient and Scalable Bipartite Matching with Fast Beta Linkage (fabl)

Brian Kundinger*, Jerome Reiter* and Rebecca C. Steorts†

Abstract. Recently, researchers have developed Bayesian versions of the Fellegi Sunter model for record linkage. These have the crucial advantage of quantifying uncertainty from imperfect linkages. However, current implementations of Bayesian Fellegi Sunter models are computationally intensive, making them challenging to use on larger-scale record linkage tasks. We propose and investigate a variation on Bayesian Fellegi Sunter models that we call fast beta linkage, or `fabl`. Specifically, in `fabl` we propose independent prior distributions over the matching space, allowing us to use hashing techniques that reduce computational overhead. This also allows us to complete pairwise record comparisons over large datasets through parallel computing and reduce memory costs through a new technique called storage efficient indexing. Through simulations and two case studies, we show that `fabl` has markedly increased speed with minimal loss of accuracy.

Keywords: bipartite record linkage, Bayesian methods, hashing techniques, parallel/distributed computing, Markov chain Monte Carlo.

1 Introduction

In many data analysis tasks, analysts seek to identify duplicate records across two databases. This is an increasingly important task in "data cleaning," and is used for inferential and predictive analyses in fields such as statistics, computer science, machine learning, political science, economics, precision medicine, official statistics, and others (Christen, 2012; Gutman et al., 2013; Dalzell and Reiter, 2018; Tang et al., 2020). In this paper, we consider bipartite record linkage, which merges two databases together that contain duplications across but not within the respective databases.

Many statistical record linkage methods are extensions of the seminal work of Fellegi and Sunter (1969) and Newcombe et al. (1959). Specifically, Fellegi and Sunter created comparison vectors for each pair of records in the data and independently classified those pairs as a match or a non-match using a likelihood ratio test. Recent work in the statistical literature has extended this approach for a wide variety of applications (Winkler and Thibaudeau, 1991; Fair, 2004; Wagner et al., 2014; Gill and Goldacre, 2003; Enamorado et al., 2019).

The independent pairwise matching assumption from Fellegi and Sunter is popular mainly for its mathematical simplicity, but is often unreasonable in practice. In many

*Department of Statistical Science, Duke University, P.O. Box 90251, Durham, NC 27708, USA brian.kundinger@duke.edu, jreiter@duke.edu

†Departments of Statistical Science and Computer Science, Duke University, P.O. Box 90251, Durham, NC 27708, USA beka@stat.duke.edu

 DOI: 0000

situations, we know that there are no duplications within a database, meaning that one record from one database should be linked with at most one record from the other. Here, additional declared matches are known by assumption to be false. Many extensions to Fellegi and Sunter (1969) resolve these false matches as a post-processing step (Jaro, 1989), but this model misspecification can still lead to poor results (Sadinle, 2017).

Alternatively, one can embed one-to-one matching requirements into the model specification itself (Gutman et al., 2013; Tancredi and Liseo, 2011), at an additional computational cost. Larsen (2005) employed a Metropolis-Hastings algorithm to only allow sampling matches that respected one-to-one assumptions, but such algorithms exhibit slow mixing due to the combinatorial nature of the constrained matching space. Fortunato (2010) used simulated annealing to target the space of matches permitted under the one-to-one constraint, but the method was so computationally intensive it could only be used to link databases of less than 100 records. Sadinle (2017) proposed the *Beta Record Linkage* model (`BRL`), using a prior over the space of bipartite matchings to strictly enforce one-to-one requirements throughout his Gibbs sampler. Additionally, he introduced a class of loss functions that allows for a flexible estimation of the linkage structure, such that the modeller can weight the relative importance of false positives and false negatives, and identify records pairings to be be decided through clerical review. Although it was shown to work on larger tasks than previous one-to-one methods, `BRL` becomes slow when working with larger datasets.

In this paper, we propose fast beta linkage (`fabl`), which extends the `BRL` model for increased efficiency and scalability. As suggested by Heck Wortman (2019), we relax the one-to-one matching requirement of `BRL` and propose independent priors over the matching space, creating a "many-to-one" model for record linkage. This allows us to (1) employ hashing techniques that hasten calculations and reduce computational costs, (2) compute the pairwise record comparisons over large datasets via parallel computing, and (3) reduce memory costs through what we call storage efficient indexing. We argue that even in cases where a bipartite matching is desired, `fabl` (with a simple post-processing procedure) provides accurate estimation of the linkage structure and other parameters, more information through which to asses model misspecification, and greatly enhanced speed. Open source software to use `fabl` in `R` is available through Github.

In what follows, Section 2 reviews the work of Fellegi and Sunter (1969) and Sadinle (2017). Section **??**, proposes the `fabl` model, derives the Gibbs sampler for posterior inference, and provides the loss function used to calculate the Bayes estimate for the bipartite matching. Section **??** introduces the hashing technique and storage efficient indexing used to increase the speed of calculatoins and the scale of linkage tasks amenable to `fabl`. Sections **??** and **??** demonstrate the speed and accuracy of `fabl` through simulation studies and case studies of homicides from the El Salvadoran Civil War and the National Long Term Care Study. Section **??** concludes with a discussion of future work.

2 Review of Prior Work

Consider two databases $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ with respective sizes n_1 and n_2. Without loss of generality, denote the files such that $n_1 \geq n_2$. In the context of bipartite matching, we assume that there are duplications across, but not within, each database. Under this framework, the set of matches across databases can be represented in two equivalent ways. First, we may use a matrix $\Delta \in \{0,1\}^{n_1 \times n_2}$, where

$$\Delta_{ij} = \begin{cases} 1 & \text{if records } i \in \boldsymbol{X}_1 \text{ and } j \in \boldsymbol{X}_2 \text{ refer to the same entity;} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Though intuitive, this sparse matrix representation can become costly for large linkage tasks. More compactly, bipartite matching also can be viewed as a labeling $\boldsymbol{Z} = (Z_1, \ldots, Z_{n_2})$ for the records in database $\boldsymbol{X}_2$ such that

$$Z_j = \begin{cases} i & \text{if records } i \in \boldsymbol{X}_1 \text{ and } j \in \boldsymbol{X}_2 \text{ refer to the same entity;} \\ n_1 + j & \text{if records } j \in \boldsymbol{X}_2 \text{ does not have a match in database } \boldsymbol{X}_1. \end{cases} \tag{2}$$

Depending on which representation is the most convenient, we can go back and forth between the two using $\Delta_{ij} = I(Z_j = i)$, where $I(\cdot) = 1$ when the expression inside the parentheses is true, and $I(\cdot) = 0$ otherwise.

Denote the set of matches by $\boldsymbol{M} = \{(i,j) : i \in \boldsymbol{X}_1, j \in \boldsymbol{X}_2, \Delta_{ij} = 1\}.$, and the set of non-matches by $\boldsymbol{U} = \{(i,j) : i \in \boldsymbol{X}_1, j \in \boldsymbol{X}_2, \Delta_{ij} = 0\}$. The record linkage task can be viewed as identifying the sets of $\boldsymbol{M}$ and $\boldsymbol{U}$. We refer to record pairs that are estimated as matches as "links", and record pairs that are estimated as non-matches as "non-links".

Intuitively, co-referent records (those that refer to the same entity) should be similar; records that are not co-referent should not be similar. Fellegi and Sunter (1969) proposed encoding this is using a comparison vector $\boldsymbol{\gamma}_{ij}$ computed for each record pair (i,j) in $\boldsymbol{X}_1 \times \boldsymbol{X}_2$. Denote the number of criteria for comparing records by F, such that $\boldsymbol{\gamma}_{ij} = (\gamma_{ij}^1, \gamma_{ij}^2, \ldots, \gamma_{ij}^f, \ldots, \gamma_{ij}^F)$. In most cases, $\boldsymbol{\gamma}_{ij}$ consists of one comparison for each feature shared between the two datasets.

The simplest way to compare two records is to check for agreement or disagreement, and this is commonly used for categorical variables. For more complex measurements, we can take into account partial agreement to more richly characterize the comparison; for numerical data, we can use absoluate difference, and for text data, we can use string distance metrics such as Levenstein or Jaro-Winkler distance (Cohen et al., 2003). We then can propose thresholds that allow us to represent comparisons through discrete levels of disagreement (Bilenko and Mooney, 2006; Elmagarmid et al., 2007). Let $\mathcal{S}_f(i,j)$ denote a general similarity measure for feature f of records i and j, where the range of $\mathcal{S}_f$ can be divided into $L_f + 1$ intervals denoted by $I_{f0}, I_{f1}, \ldots, I_{fL_f}$. Following convention, I_{f0} represents the highest level of agreement (inclusive of complete agreement) and I_{fL_f} represents the highest level of disagreement (including complete disagreement). Thus, we can construct comparison vectors in the following way: $\gamma_{ij}^f = \ell$ if $\mathcal{S}_f(i,j) \in I_{f\ell}$. The choice of $I_{f\ell}$ are application specific, which we discuss in our simulation and case studies.

In practice, it is not feasible to make all-to-all record comparisons as the computational complexity is of order $O(n_1 \times n_2)$. The most common solution is to utilize blocking, which places similar records into partitions, or "blocks," to reduce this computational burden (Steorts et al., 2014; Murray, 2016). In deterministic blocking, the modeller chooses a field thought to be highly reliable, and only compares records that agree on that field. The record linkage method is then applied independently across all blocks, which can be done in parallel for additional speed gains. However, blocking on an unreliable field can lead to missed matches, making this form of blocking undesirable in some situations.

In this paper, we use hashing and new technique called storage efficient indexing to increase the scale of the linkage tasks we can undertake without blocking. This is useful when there is no reliable blocking field available, or one desires estimates of model parameters for the entire sample in question. In practice, `fabl` can be combined with blocking, but all derivations, simulations, and case studies are presented here without blocking.

References

Bilenko, M. and Mooney, R. (2006). "Riddle: Repository of Information on Duplicate Detection, Record Linkage, and Identity Uncertainty." Online; retrieved July 29, 2020. URL http://www.cs.utexas.edu/users/ml/riddle/

Christen, P. (2012). "A survey of indexing techniques for scalable record linkage and deduplication." *IEEE Transactions on Knowledge and Data Engineering*, 24(9): 1537–1555. 1

Cohen, W. W., Ravikumar, P., and Fienberg, S. E. (2003). "A Comparison of String Distance Metrics for Name-Matching Tasks." In *Proceedings of the 2003 International Conference on Information Integration on the Web*, 73–78. AAAI Press.

Dalzell, N. M. and Reiter, J. P. (2018). "Regression modeling and file matching using possibly erroneous matching variables." *Journal of Computational and Graphical Statistics*, 27(4): 728–738. 1

Elmagarmid, A. K., Ipeirotis, P. G., and Verykios, V. S. (2007). "Duplicate Record Detection: A Survey." *IEEE Transactions on Knowledge and Data Engineering*, 19(1): 1–16.

Enamorado, T., Fifield, B., and Imai, K. (2019). "Using a Probabilistic Model to Assist Merging of Large-Scale Administrative Records." *American Political Science Review*, 113: 353–371. 1

Fair, M. (2004). "Generalized record linkage system–Statistics Canada's record linkage software." *Austrian Journal of Statistics*, 33(1&2): 37–53. 1

Fellegi, I. P. and Sunter, A. B. (1969). "A Theory for Record Linkage." *Journal of the American Statistical Association*, 64(328): 1183–1210. 1, 2, 3

Fortunato, S. (2010). "Community detection in graphs." *Physics Reports*, 486(3): 75–174. 2

Gill, L. and Goldacre, M. (2003). "English national record linkage of hospital episode statistics and death registration records." *Report to the department of health*. 1

Gutman, R., Afendulis, C. C., and Zaslavsky, A. M. (2013). "A Bayesian procedure for file linking to analyze end-of-life medical costs." *Journal of the American Statistical Association*, 108(501): 34–47. 1, 2

Heck Wortman, J. (2019). "Record Linkage Methods with Applications to Causal Inference and Election Voting Data." 2

Jaro, M. A. (1989). "Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida." *Journal of the American Statistical Association*, 84(406): 414–420. 2

Larsen, M. (2005). "Hierarchical Bayesian Record Linkage Theory." 2

Murray, J. S. (2016). "Probabilistic Record Linkage and Deduplication after Indexing, Blocking, and Filtering." *Journal of Privacy and Confidentiality*, 7(1): 3–24.

Newcombe, H. B., Kennedy, J. M., Axford, S. J., and James, A. P. (1959). "Automatic Linkage of Vital Records." *Science*, 130(3381): 954–959. 1

Sadinle, M. (2017). "Bayesian Estimation of Bipartite Matchings for Record Linkage." *Journal of the American Statistical Association*, 112(518): 600–612. 2

Steorts, R. C., Ventura, S. L., Sadinle, M., and Fienberg, S. E. (2014). "A Comparison of Blocking Methods for Record Linkage." In *Privacy in Statistical Databases*, Lecture Notes in Computer Science, 253–268. Springer, Cham.

Tancredi, A. and Liseo, B. (2011). "A Hierarchical Bayesian Approach to Record Linkage and Size Population Problems." *Annals of Applied Statistics*, 5(2B): 1553–1585. 2

Tang, J., Reiter, J., and Steorts, R. C. (2020). "Bayesian Modeling for Simultaneous Regression and Record Linkage." In *Privacy in Statistical Databases*, Lecture Notes in Computer Science. Springer, Cham. 1

Wagner, D., Lane, M., et al. (2014). "The person identification validation system (PVS): applying the Center for Administrative Records Research and Applications'(CARRA) record linkage software." Technical report, Center for Economic Studies, US Census Bureau. 1

Winkler, W. E. and Thibaudeau, Y. (1991). *An Application of the Fellegi-Sunter Model of Record Linkage to the 1990 US Decennial Census*. 1