A Novel Methodology for Improving Applications of Modern Predictive Modeling Techniques to Linked Data Sets Subject to Mismatch Error

1st Emanuel Ben-David
Center for Statistical Research & Methodology
U.S. Census Bureau
Washington, District of Columbia, USA
e.h.bendavid@gmail.com

2nd Brady T. West
Institute for Social Research
University of Michigan
Ann Arbor, Michigan, USA
bwest@umich.edu

3rd Martin Slawski
Department of Statistics
George Mason University
Fairfax, Virginia, USA
mslawsk3@gmu.edu

2023 Big Data Meets Survey Science (BigSurv) | 979-8-3503-4527-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/BigSurv59479.2023.10486610

***Abstract*—In recent years, the rise of social media platforms such as Twitter/X has provided social scientists with a wealth of user-content data. Combining social media and survey data has the potential to produce a comprehensive source of information for social research. These data are often collected from multiple sources and combined by probabilistic record linkage. For the analysis of these linked data files, advanced machine learning techniques, such as random forests, boosting, and related ensemble methods, have become essential tools for survey methodologists and data scientists. There is, however, a potential pitfall in the widespread application of these techniques to linked data sets that needs more attention. Linkage errors such as mismatch and missed-match errors can distort the true relationships between variables and adversely alter the performance metrics routinely output by predictive modeling techniques, such as variable importance, confusion matrices, RMSE, etc. Thus, the actual predictive performance of these machine-learning techniques may not be realized. In this paper, we describe a methodology designed to adjust modern predictive modeling techniques for the presence of mismatch errors in linked data sets. The proposed approach, based on mixture modeling, is general enough to accommodate various predictive modeling techniques in a unified fashion. We evaluate the performance of our proposed methodology with simulations implemented in R. We conclude with recommendations for future work.**

***Index Terms*—record linkage, data integration, social media, Twitter/X, ensemble methods, bagging trees, random forests, mismatch error, mixture model, secondary analysis**

I. Introduction

A. Background and Motivation

In recent years, the rise of social media platforms such as Twitter/X has provided social scientists with a wealth of user-content data [1]–[4]. The research methodology of combining social media and survey data has the potential to produce a richer and more comprehensive source of information for social research [5], [6]. These data are often collected from multiple sources and combined using probabilistic record linkage (RL). Advanced machine learning techniques, such as random forests, boosting, and related ensemble methods, are then applied to these linked data sets to analyze relationships between variables of interest. These techniques have become essential tools for survey methodologists and data scientists conducting both methodological and substantive research [7], [8]. There is, however, a potential pitfall in the widespread application of these techniques to linked data sets that needs more attention. Linkage errors such as mismatch errors and missed-match errors can distort the true relationships between variables in these linked data sets and adversely alter the performance metrics used to evaluate the predictive power of these techniques, such as variable importance, confusion matrices, RMSE, etc.; see [9], [10] and the references within. Thus, the actual predictive performance of these machine learning techniques may not be fully realized due to the errors in RL.

RL is an essential task when combining data from heterogeneous data sources [11]–[13]. The fundamental RL task is to identify the records in different data sets that belong to the same entity. Examples include the linkage of data from survey respondents and corresponding administrative records for those respondents; insurance claims from patients and corresponding hospital records from those patients; data from historical censuses; social media and survey data; and many others. In most applications, due to the lack of common and unique identifiers, RL relies on a probabilistic matching process. The probabilistic process of identifying matching is not exact and subject to error. For instance, privacy considerations often prevent the use of unique identifiers for RL. Missing data and quality issues (e.g., formatting or spelling variations) can also induce substantial uncertainty, with one record yielding many candidate matches in the other data set. A linkage error could be either a false match (henceforth *mismatch*) or a false non-match (*missed match*). Both types of errors can negatively affect downstream statistical analyses (*post-linkage analysis*) and predictive modeling performed on the linked data set.

Missed matches can result in a non-representative sample and subsequently selection bias in estimates similar to that

The authors gratefully acknowledge funding for this research from NSF-MMS (NSF Grant 2120318, PI: Martin Slawski).

produced by nonresponse in survey data [14]. Mismatches, on the other hand, can cause data contamination and typically distorted relationships when analyzing associations between variables, e.g., in regression analysis. This is a well-studied problem pioneered by [15] with important follow-up work by [16], [17] and [18]. More recently, various approaches have been proposed to account for mismatches in post-linkage data analysis. This body of work can be roughly divided according to whether it addresses *primary analysis* or *secondary analysis*. The former refers to scenarios in which the same individual performs RL and downstream analysis, or the data analyst has at least significant insights into the details of the underlying RL [19]–[24].

In the secondary analysis setting, by contrast, the data analyst has no access to the original data sets being linked and only has access to the linked data set, which may also include some ancillary information about the mismatches. For instance, the data analyst may be given scores reflecting the likelihood of every linked record being a correct match, as in the recent study by [25]. Our focus in this paper is on the secondary analysis setting, and empirically evaluating potential tools that data analysts applying machine learning techniques to linked data sets can use to avoid the attenuation in predictive performance engendered by errors in record linkage.

B. Research Contributions

In this paper, we describe a methodology designed to adjust modern predictive modeling techniques, in particular, bagging trees and random forests, for the presence of mismatch errors in a secondary analysis setting. We propose a mixture model to account for data contamination resulting from incorrect links. The proposed approach is general enough to accommodate various predictive modeling techniques in a unified fashion. We will evaluate the performance of our proposed methodology via an empirical simulation study.

Our simulation study is motivated by research investigations where only limited information is available for linking microdata from survey respondents with data from some other source, such as social media data, administrative records, or historical data, and the secondary data analyst only has the linked data set with which to work. In such applications, the available identifying information for a given respondent may not be sufficiently unique for correct RL, resulting in many candidate matches in one or both data sets. For example, suppose that we wish to link responses from an economic survey, including a dependent variable of interest *Y* (e.g., current salary) with administrative records on employment, including a vector of predictor variables of interest *X* (e.g., establishment size and type of industry), but only socio-demographic information and job classifications are available for RL in the administrative data. In this setting, probabilistic record linkage may be employed, resulting in uncertainty about the quality of the matches and possible mismatch error.

With our simulation study, we demonstrate that our new methodology, which we have implemented in R [26], can recover predictive performance results from applications of modern predictive modeling techniques to such linked data sets that would have been seen *prior* to the introduction of mismatch errors in the RL process.

II. Methodology

Before we formally describe our proposed methodology, we provide some conceptual insight. We focus on bagging, an abbreviation for bootstrap aggregating [27], and random forests [28]. Bagging trees start by generating some decision trees, say T, each trained by bootstrap sampling with replacement from the observed data. In the setting of complex probability sampling, the bootstrap sampling needs to account for the features of the complex sample design, such as weighting, stratification, and cluster sampling [29], [30]. Thus, some observations may appear more than once, while others may not be present in the sample. The predictions are then aggregated by averaging the predictions of the T decision trees. Random forests are an extension of bagging trees in which, during the construction of a decision tree, an algorithm first randomly selects a subset of predictors at each step of determining splits.

There are two ways that variable weights for different cases (e.g., frequency weights, analytic weights, survey weights, etc.) can be incorporated into these ensemble methods. One option is to incorporate the weights in bootstrap sampling. This option is available in the R package "`ranger`" for a fast implementation of the random forest by specifying case weights [31]:

```
ranger(formula, data, case.weights, ...)
```

Weights can also be incorporated into the construction of the decision tree, which is available in the R package "`rpart`" for Recursive Partitioning and Regression Trees, by specifying weights [32]:

```
rpart(formula, data, weights,...)
```

As explained in the package manual, the sum of the case weights for those observations reaching a node is used to decide how to split the tree at that node. In the weighting-reweighting adjustment method proposed in subsection II-C, we use these options by specifying weights that reflect uncertainty about the matches in a linked data set and, accordingly, gauging their influence on training the ensemble method.

A. Notation and setup

We assume that there are n observations $(y_i, \mathbf{x}_i)$, $i = 1, ..., n$, generated from the joint distribution $f(y, \mathbf{x})$, with y_i representing a dependent variable of interest and $\mathbf{x}_i$ representing a vector of predictor variables. The y_i and $\mathbf{x}_i$ are collected from two separate data sources, with y_i arising from data set A and $\mathbf{x}_i$ arising from data set B. These variables are then paired in $\mathcal{D}$, obtained by RL of the data sets A and B. Due to mismatch errors, observations in the merged data set are $(\tilde{y}_i, \mathbf{x}_i)$, in which with a probability, say α, $\tilde{y}_i \neq y_i$. This means that the observations in the merged data are from the joint density of $\tilde{y}$ and $\mathbf{x}$.

Let $\mu_{y|\mathbf{x}}$ denote the regression function $E[y \mid \mathbf{x}]$. In terms of mean squared error, $\mu_{y|\mathbf{x}}$ is the best prediction of y given $\mathbf{x}$. The main objective in applications of machine learning methods is to accurately estimate $\mu_{y|\mathbf{x}}$ from the random sample $(y_1, \mathbf{x}_1), \ldots (y_1, \mathbf{x}_n)$ generated from $f(y, \mathbf{x})$. However, since the observations in the linked data set are from the joint density of $\tilde{y}$ and $\mathbf{x}$, the estimated regression function is an estimate $\hat{\mu}_{\tilde{y}|\mathbf{x}}$ of $\mu_{\tilde{y}|\mathbf{x}}$. Here we propose a general method for using $\hat{\mu}_{\tilde{y}|\mathbf{x}}$ to obtain an improved estimate $\hat{\mu}_{y|\mathbf{x}}$.

Following a mixture modeling approach and assuming that no selection bias is introduced by any missed matches [33], the conditional density of $\tilde{y}_i \mid (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is equal to $(1-\alpha) f_{y_i|\mathbf{x}_i}(\tilde{y}_i \mid \mathbf{x}_i) + \alpha f_y(\tilde{y}_i)$, where $f_{y_i|\mathbf{x}_i}$ denotes the conditional density of $y_i \mid \mathbf{x}_i$ and f_y denotes the marginal density of y. A slightly more generalized form of this mixture model is $f_{\tilde{y}_i|\mathbf{x}_1,\ldots,\mathbf{x}_n}(\tilde{y}_i \mid \mathbf{x}_1, \ldots, \mathbf{x}_n)$, equal to

$$(1-\alpha_i) f_{y_i|\mathbf{x}_i}(\tilde{y}_i \mid \mathbf{x}_i) + \alpha_i f_y(\tilde{y}_i), \tag{1}$$

where the mixing coefficient α_i is interpreted as the probability that $\tilde{y}_i \neq y_i$. This mixture density implies that

$$\mu_{y_i|\mathbf{x}_i} = \frac{1}{1-\alpha_i}\mu_{\tilde{y}_i|\mathbf{x}_i} - \frac{\alpha_i}{1-\alpha_i}\mu_y. \tag{2}$$

By interpreting $1-\alpha_i$ as the probability of a correct match, the posterior probability of $\tilde{y}_i = y_i \mid \mathcal{D}$ is

$$\frac{(1-\alpha_i) f_y(\tilde{y}_i)}{(1-\alpha_i) f_{y|\mathbf{x}}(\tilde{y}_i \mid \mathbf{x}_i) + \alpha_i f_y(\tilde{y}_i)}. \tag{3}$$

Equations (2) and (3) suggest two different adjustment methods for estimating $\mu_{y|\mathbf{x}}$. Although the proposed adjustments generally apply to any ensemble method, we focus on the adjustments applied to bagging trees and random forests in the sequel.

Finally, we note that our proposed methods do not require that the preceding RL step generates one-to-one matching. Because we are excluding the more general setting where there are missed matches, in addition to mismatches, our use of one-to-one matching for the evaluation ensures that the effect of linkage error on the post-linkage analysis is due to mismatches.

B. The optimal-α adjustment method

In a semi-parametric setting, suppose $f_{y_i|\mathbf{x}_i}$ is parameterized by $\mu_{y_i|\mathbf{x}_i}$ and $\sigma^2_{y_i|\mathbf{x}_i}$, and f_y is parameterized by μ_y and σ^2_y. We assume that $\sigma^2_{y_i|\mathbf{x}_i}$ is the same for each i, i.e., homoscedastic variances. The pseudo-likelihood function for the mixture density in (1), denoted by $\mathcal{L}\left((\mu_{y_i|\mathbf{x}_i}, \alpha_i)_{i=1}^n, \mu_y, \sigma^2_{y|\mathbf{x}}, \sigma^2_y\right)$, can be written as:

$$\prod_{i=1}^{n} \left((1-\alpha_i) f_{y_i|\mathbf{x}_i}(\tilde{y}_i \mid \mathbf{x}_i, \mu_{y_i|\mathbf{x}_i}, \sigma^2_{y|\mathbf{x}}) + \alpha_i f_y(\tilde{y}_i \mid \mu_y, \sigma^2_y) \right). \tag{4}$$

Note that the main parameters of interest are $\mu_{y_i|\mathbf{x}_i}$. Among these parameters, μ_y and σ^2_y can be directly estimated as the sample mean and variance of $\tilde{y}_1, \ldots, \tilde{y}_n$. An estimate of $\sigma^2_{y|\mathbf{x}}$ can be obtained from an ordinary least squares regression of $\tilde{y}$ on $\mathbf{x}$. Plugging in the estimates $\hat{\mu}_{\tilde{y}_i|\mathbf{x}_i}$ and $\hat{\mu}_y$ in (2) yields

$$\hat{\mu}_{y_i|\mathbf{x}_i} = \frac{1}{1-\alpha_i}\hat{\mu}_{\tilde{y}_i|\mathbf{x}_i} - \frac{\alpha_i}{1-\alpha_i}\hat{\mu}_y. \tag{5}$$

By replacing these estimates in the right-hand side of (4), we maximize the pseudo-likelihood as a function of $(\alpha_1, \ldots, \alpha_n)^\top$, i.e.,

$$(\hat{\alpha}_1^{opt}, \ldots, \hat{\alpha}_n^{opt})^\top = \text{argmax}_{0 \leq \alpha_i \leq 1} \mathcal{L}(\alpha_1, \ldots, \alpha_n).$$

More precisely,

$$\hat{\alpha}_i^{opt} = \text{argmax}_{0 \leq \alpha_i \leq 1} (1-\alpha_i) f_{y_i|\mathbf{x}_i}(\tilde{y}_i \mid \mathbf{x}_i, \frac{1}{1-\alpha_i}\hat{\mu}_{\tilde{y}_i|\mathbf{x}_i} - \frac{\alpha_i}{1-\alpha_i}\hat{\mu}_y, \hat{\sigma}^2_{y|\mathbf{x}}) + \alpha_i f_y(\tilde{y}_i \mid \hat{\mu}_y, \hat{\sigma}^2_y). \tag{6}$$

Since estimation here involves many parameters (one per observation), improvements can be achieved via regularization, the use of a suitable prior distribution for "borrowing strength", or an additional layer of modeling. For example, the α_i parameters can be modeled conditional on information regarding the correctness of links from a preceding (external) record linkage step (e.g., predicted matching probabilities). We also note that to proceed with (6), we need to specify a semi-parametric density for the conditional density $f_{y|\mathbf{x}}$. In Algorithm 1, we use bagging to aggregate the optimal estimates of $\alpha_1, \ldots, \alpha_n$ obtained from bootstrap samples. The optimal α_i's obtained in Algorithm 1 now can be used to estimate the $\mu_{y_i|\mathbf{x}_i}$'s. First, we apply the analyst's favorite predictive model to the $\mathcal{D}$ data. This gives us estimates $\hat{\mu}_{\tilde{y}_1|\mathbf{x}_1}, \ldots, \hat{\mu}_{\tilde{y}_n|\mathbf{x}_n}$. Using (5), the optimal α adjustment method yields:

$$\hat{\mu}_{y_i|\mathbf{x}_i} = \frac{1}{1-\hat{\alpha}_i^{opt}}\hat{\mu}_{\tilde{y}_i|\mathbf{x}_i} - \frac{\hat{\alpha}_i^{opt}}{1-\hat{\alpha}_i^{opt}}\hat{\mu}_y.$$

Algorithm 1: Obtaining an optimal α

Input : regression `formula`, `data`, `ntrees` (number of trees)
Output: $\alpha = (\hat{\alpha}_1^{opt}, \ldots, \hat{\alpha}_n^{opt})$

initiate estimates of μ_y, σ^2_y and $\sigma^2_{y|\mathbf{x}}$
/* We estimate μ_y and σ^2_y as the sample mean and variance of $\tilde{y}_i$'s, and $\sigma^2_{y|\mathbf{x}}$ from the ordinary least squares regression of $\tilde{y} \sim \mathbf{x}$. */
for $t \leftarrow 1$ **to** `ntrees` **do**
    `subdata` <- subsample with replacement from data
    obtain estimates $\hat{\mu}^{(t)}_{\tilde{y}_i|\mathbf{x}_i}$ from a decision tree applied to `formula` and `subdata`
    for $i \leftarrow 1$ **to** n **do**
        calculate $\hat{\alpha}_i^{opt,(t)}$ using (6)
    end for
end for
return $(\hat{\alpha}_i^{opt})_{i=1}^n = \left(\sum_{t=1}^{ntrees} \hat{\alpha}_i^{opt,(t)} / ntrees\right)_{i=1}^n$

If the desired setting is $\alpha_1, \ldots, \alpha_n = \alpha$, we use $\hat{\alpha}^{opt} = \sum_{i=1}^{n} \hat{\alpha}_i^{opt}/n$ in the equation above. We refer to this adjustment method as the mean-optimal α adjustment. The mean-optimal α can be efficiently approximated by a small sample of $\hat{\alpha}_i^{opt}$'s, which results in a computationally much less expensive adjustment method than that of the optimal-α method, as the latter requires computing $\hat{\alpha}_i^{opt}$ for all observations. In terms of mean squared error (MSE) and mean squared prediction error (MSPE), we did not find any significant difference between the optimal α and the mean-optimal version of the adjustment methods in our simulation study, presented in Section III.

Finally, we note that if estimated probabilities of correct matches are available for each linked case from an external record linkage process, we could forego the computation of $\hat{\alpha}_i^{opt}$ and simply use these estimated probabilities in the adjustment described above. That is,

$$\hat{\mu}_{y_i|\mathbf{x}_i} = \frac{1}{w_i}\hat{\mu}_{\tilde{y}_i|\mathbf{x}_i} - \frac{1-w_i}{w_i}\hat{\mu}_y,$$

where w_i is the estimated probability of a correct match.

C. The weighting-reweighting adjustment method

As mentioned at the beginning of this section, uncertainty about the matches can be incorporated into random forests via either bootstrap sampling or the construction of decision trees. The main idea in either method is that observations with greater certainty of being true matches get larger contributions in training the predictive model. If these weights are known (e.g., estimated probabilities of correct matches are provided from an external probabilistic record linkage process), we propose to initially weight each observation $(\tilde{y}_i, \mathbf{x}_i)$ based on its (estimated) probability of being a match. Otherwise, we give each observation the same weight, which is essentially the specification of a "non-informative prior" for this method. The availability of external information about probabilities of correct matches has the potential to lead to faster convergence of the algorithm described below and more accurate predictions. After estimating $\mu_{y_i|\mathbf{x}}$ from our predictive model, we update each weight as the posterior probability of being a match using (3). Then, we refit the model with the new weights to update the predictions. We repeat this procedure until the likelihood function indicates that the new weights achieve no significant improvement.

In Algorithm 2, we apply the bagging method to decision trees such that their estimates of $\mu_{y_i|\mathbf{x}}$ are adjusted according to the weighting-reweighting method. Note that depending on whether the weights are incorporated via the case weights option in bootstrap sampling or via the weights option in decision trees, Algorithm 2 results in two different weighting-reweighting adjustment methods. We denote the former as Adj-rf and the latter as Adj-trees. We compare these two versions of the weighting-reweighting adjustment methods in Section III.

Algorithm 2: The weighting-reweighting adjustment method

Input : regression `formula`, `data`, `ntrees`, `num.iter` (number of iterations), `weights`, and the mismatch probabilities $\alpha_1, \ldots, \alpha_n$

Output: $(\hat{\mu}_{y_1|\mathbf{x}_1}, \ldots, \hat{\mu}_{y_n|\mathbf{x}_n})$

```
. /* We choose weights = 1 and
    alpha_i, ..., alpha_n = .5 for default values. */
initiate estimates of mu_y, sigma_y^2 and sigma_{y|x}^2
/* We estimate mu_y and sigma_y^2 as the sample
    mean and variance of tilde{y}_i's, and sigma_{y|x}^2
    from the ordinary least squares
    regression of tilde{y} ~ x. */
for t <- 1 to ntrees do
    subdata <- subsample with replacement from
     data
    obtain estimates mu^(t)_{y_i|x_i} from a decision tree applied
     to formula, subdata and weights
    iter <- 1
    while iter < num.iter do
        update each alpha_i as the posterior probability of
            mismatch /* this is 1 -
            posterior probability of
            match in (3) using the
            previous value of alpha_i. */
        update weights_i = 1 - alpha_i, for each i
        update mu^(t)_{y_i|x_i} by rerunning the decision tree
         with the new weights
        evaluate L(alpha_1, ..., alpha_n) for the current and
         previous alpha_i's
        if the difference between the two likelihood
         values is not significant, e.g., 10e-8, then
            break
        end if
    end while
end for
return (mu^_{y_i|x_i})_{i=1}^n = ( sum_{t=1}^{ntrees} mu^(t)_{y_i|x_i} / ntrees )_{i=1}^n
```

D. Combining proposed adjustment methods

With the optimal-α adjustment method and the weighting-reweighting method described in II-B and II-C, a natural extension is to combine these two methods by applying the optimal-α adjustment to the $\hat{\mu}_{y|\mathbf{x}}$ estimated from the weighting-reweighting adjustment method Adj-rf or Adj-trees.

III. Simulations and Evaluation Results

In this section, we compare and evaluate the performance of the proposed adjustment methods with two simulation studies. We use the MSE to measure the fit of a model and the MSPE to measure the quality of the model predictions. The MSE is calculated as the mean of squared residuals, i.e., $(1/n)\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$, where n denotes the size of the data and $\hat{y}_i$ denoted the fitted y- value for the i^{th} observation. The MSPE is calculated similarly but by averaging over the test data set, which is obtained by randomly splitting the data to 30% for the test data and the rest for the training data.

mismatch rate	0%	10%	15%	20%
bagging trees	22.0	28.7	41.7	50.2
random forest	**18.4**	58.6	90.2	112
Adj-rf	18.5	25.1	**33.6**	40
Adj-trees	22.0	28.8	41.6	50
Optimal-α-bagging	21.9	**24.0**	**33.6**	37.0
Optimal-α-rf	19.1	60.0	92.6	115
Optimal-α-Adj-rf	19.1	27.7	37.5	45.9
Optimal-α-Adj-trees	21.9	24.1	**33.6**	**36.9**

mismatch rate	25%	30%	35%	40%
bagging trees	61.7	78.9	91.7	109
random forest	132	156	176	202
Adj-rf	55.5	77.9	90.8	108
Adj-trees	61.8	79.0	91.6	109
Optimal-α-bagging	**45.3**	**60.2**	69.2	78.6
Optimal-α-rf	135	161	184	210
Optimal-α-Adj-rf	62.4	86.8	102	125
Optimal-α-Adj-tree	**45.3**	**60.2**	**69.1**	**78.4**

TABLE I
EACH CELL VALUE IS THE MSE OF THE PREDICTION METHOD SPECIFIED BY THE ROW NAME APPLIED TO THE LINKED DATA WITH A MISMATCH RATE SPECIFIED BY THE COLUMN NAME. THE MISMATCH RATES VARY BETWEEN 0% - 40%.

mismatch rate	0%	10%	15%	20%
bagging trees	35.1	40.7	48.3	60
random forest	**18.4**	61.3	81	108
Adj-rf	**18.4**	26.5	33.7	43.1
Adj-trees	23.9	29.4	37.4	49
Optimal-α-bagging	35.7	37.8	41.6	48.8
Optimal-α-rf	19.1	63.2	83	111
Optimal-α-Adj-rf	19.1	29.7	38.2	50.2
Optimal-α-Adj-trees	24	**25**	**29**	**35.4**

mismatch rate	25%	30%	35%	40%
bagging trees	74.5	87.4	105	124
random forest	133	156	181	218
Adj-rf	55.4	68.6	88.2	113
Adj-trees	63	76.2	92.8	116
Optimal-α-bagging	58	65.9	78.7	92.9
Optimal-α-rf	137	162	188	229
Optimal-α-Adj-rf	65.1	81	103	134
Optimal-α-Adj-trees	**44.2**	**52.3**	**64**	**81.2**

TABLE II
EACH CELL VALUE IS THE MSPE OF THE PREDICTION METHOD SPECIFIED BY THE ROW NAME APPLIED TO THE LINKED DATA WITH A MISMATCH RATE SPECIFIED BY THE COLUMN NAME. THE MISMATCH RATES VARY BETWEEN 0% - 40%

Optimal-α	Mean-optimal α	bagging trees
15.455	1.911	0.614
random forest	Adj-rf	Adj-trees
0.041	0.26	7.569

TABLE III
THE RUNNING TIME OF EACH METHOD IN SEC

A. *Simulation with a single predictor variable*

In the first simulation study, the simulated data set contains $n = 1000$ observations and two variables: a single predictor x and a response variable y generated via the equation $y_i = g(x_i)+4N(0,1)$, where $g(\cdot)$ is a non-linear function of x, and $N(0,1)$ is Gaussian noise.

The experiment begins by creating a new data set resembling a linked data set, in which k percent of the pairs (y_i, x_i) are mismatched. The mismatches are created by randomly permuting k percent of the indices of y. The prediction and adjustment methods are trained by this data set, and the MSE and MSPE are calculated. To see the effects of different mismatch rates on these methods, the MSE and MSPE are computed for $k = 0, 10, 15, 20, 25, 30, 35, 40$. Henceforth, when no mismatches are in the data, i.e., $k = 0$, we refer to the data as the *exact* data. Each run of the experiment thus results in two tables, one for MSE and another for MSPE, each with 8 columns representing the mismatch rate and 8 methods applied. We run each experiment 1000 times and report the averages in Table I for the MSE and Table II for MSPE over the 1000 replications.

These tables show that the best adjustment is achieved by combining the optimal-α adjustment with the Adj-trees method. Among the methods with a single adjustment, Optimal-α-bagging, i.e., Optimal-α adjustment with $\hat{\mu}_{y|\mathbf{x}}$ estimated from bagging trees, has the best performance.

Table III shows the running time of each method in seconds in every single implementation of the method on the data. Optimal-α and Adj-trees are computationally expensive; therefore, when the computational cost is a concern, mean-optimal α combined with bagging trees, random forests, or Adj-rf may be preferable.

Next, we use some visualization tools to compare the bagging trees, random forests, and the proposed adjustment methods further. Fig. 1, based on Table II, visualizes the comparison between bagging trees, random forests, and the optimal-α trees and Adj-rf in terms of relative prediction errors. The relative prediction errors are computed relative to the smallest prediction error in column one of Table II, i.e., 18.4, which is the prediction error of random forest with the exact data. Fig. 1 confirms that the optimal-α-Adj-trees have the lowest prediction error.

Instead of comparing the alternative methods based on their MSE and MSPE, we now compare these based on $\hat{\mu}_{y|x}$, their estimates of the regression function. In the plots, $\hat{\mu}_{y|x}$ is depicted by drawing the scatter or line plot of $(\hat{y}, x)$, where

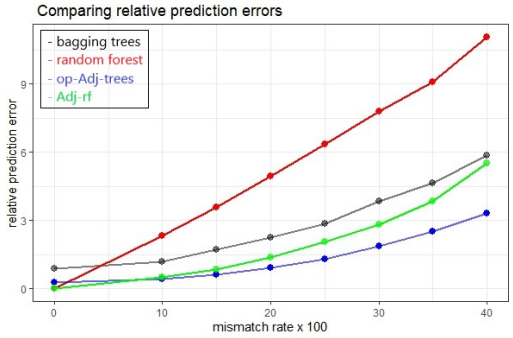

Fig. 1. Relative prediction error plot of bagging trees, random forests, optimal-α trees and Adj-rf.

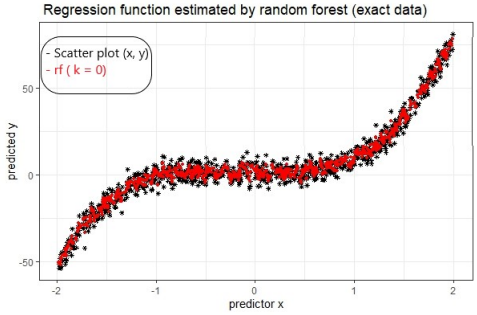

Fig. 2. The scatter plot of (x, y) in black and the scatter plot of $(x, \hat{y})$ in red , where $\hat{y}$ is the fitted y-value obtained from the random forest trained by the exact data. The scatter plot $(x, \hat{y})$ closely matches that of (x, y).

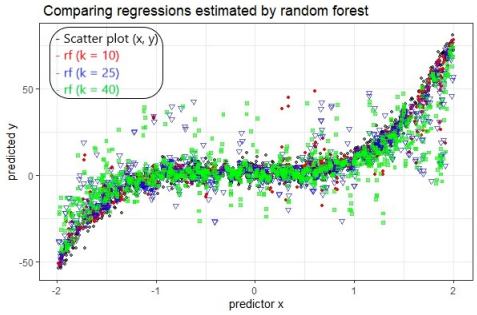

Fig. 3. The scatter plot of $(x, \hat{y})$ is drawn when, respectively, $10\%, 20\%$ and 40% of the observations are mismatched. As the mismatch rate increases, the associated scatter plot rapidly deviates from the scatter plot of (x, y) from the exact data.

$\hat{y}$ is the average, over 1000 replications, of the fitted y-value obtained from the associated method. Since we are averaging, the plot depicts the expectation of $\hat{\mu}_{y|x}$.

Fig. 2 shows that the random forest trained by the exact data very well estimates the regression function. However, as the mismatch rate increases, Fig. 3 shows that the $\hat{\mu}_{y|x}$ estimated from the random forest rapidly deviates from the scatter plot of (x, y) based on the exact data.

In Fig. 4, we compare the regression functions estimated from the random forest, Adj-rf and optimal-α-trees trained by the data when 20% of the observations are mismatched. It is somewhat surprising that Adj-rf more closely matches the regression function estimated from the random forest trained by the exact data. As we mentioned earlier, since the estimated functions are averaged over 1000 replications, this shows that the regression function estimated from Adj-rf is less biased than that of optimal-α-trees but has a larger variance.

Finally, in Fig. 5, we compare the regression functions estimated from the optimal-α-trees when the mismatches in the training data are respectively 10%, 25% and 40%. Fig. 5 shows that the optimal-α-trees method, as opposed to the random forest, is more robust to mismatches in the data.

B. Simulation with multiple predictor variables

In the next simulation study, the simulated data set also has $n = 1000$ observations with 10 predictors $x_1, \ldots, x_{10}$. The

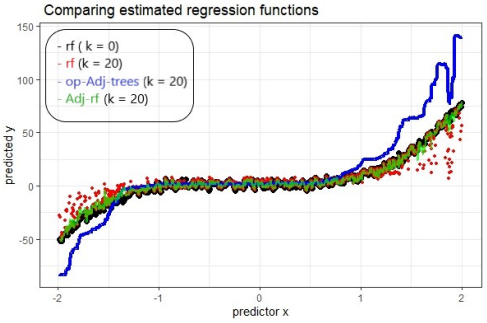

Fig. 4. The plots show the regression functions estimated from the random forest, Adj-rf and optimal-α-trees trained by the data when 20% of the observations are mismatched. The comparison is based on how closely they match the regression function estimated from the random forest trained by the exact data.

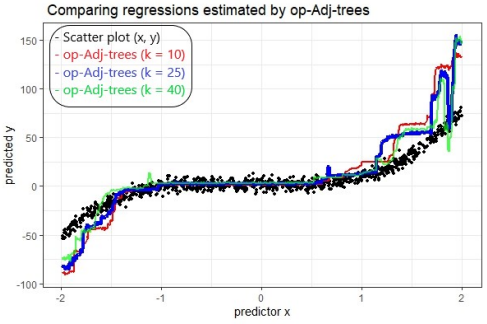

Fig. 5. The regression functions estimated from the optimal-α Adj-trees when the mismatches in the training data are respectively 10%, 25% and 40%. The scatter plot of (x, y) from the exact data is drawn for comparison.

response variable y is generated via equation $y_i = h(\mathbf{x}_i) + \sqrt{2}/2N(0, 1)$, where $h(\cdot)$ is a non-linear real-valued function of $\mathbf{x} = (x_1, \ldots, x_{10})^\top$. The experiment is performed similarly to the one described in III-A with 1000 replications. Table IV presents the MSE and Table V the MSPE over the 1000 replications.

These tables show that the best adjustment based on MSPE is achieved by combining the optimal-α adjustment with Adj-rf, although Optimal-α-bagging and Optimal-α-Adj-trees have lower MSE. These tables also show, as opposed to the previous experiment in III-A with a single predictor, that the random forest approach is robust to contamination caused by mismatches. This could also be due to more minor noise in generating y.

The running times of these methods shown in Table VI are consistent with those in Section III. Although the number of observations in both simulation studies is 1000, the increase in running time in this experiment is due to the larger number of predictors.

Fig. 6 shows the relative prediction error plot of bagging trees, random forests, optimal-α-trees, and optimal-α-Adj-rf, based on Table V. The plot shows that optimal-α-Adj-rf has a smaller MSPE than other methods. In this simulation, the random forest is more robust to mismatches in the data.

mismatch rate	0%	10%	15%	20%
bagging trees	0.75	0.79	0.82	0.87
random forest	0.84	0.91	0.94	0.99
Adj-rf	0.84	0.86	0.89	0.92
Adj-trees	0.75	0.79	0.82	0.87
Optimal-α-bagging	**0.7**	**0.71**	**0.73**	**0.77**
Optimal-α-rf	0.79	0.83	0.86	0.91
Optimal-α-Adj-rf	0.8	0.82	0.84	0.86
Optimal-α-Adj-trees	**0.7**	**0.71**	**0.73**	**0.77**
mismatch rate	25%	30%	35%	40%
bagging trees	0.92	0.96	1	1.04
random forest	1.04	1.08	1.11	1.14
Adj-rf	0.94	0.97	1	1.03
Adj-trees	0.92	0.96	1	1.04
Optimal-α-bagging	**0.82**	**0.86**	**0.9**	**0.94**
Optimal-α-rf	0.95	0.99	1.02	1.05
Optimal-α-Adj-rf	0.87	0.9	0.93	0.96
Optimal-α-Adj-trees	**0.82**	**0.86**	**0.9**	**0.94**

TABLE IV
EACH CELL VALUE IS THE MSE OF THE PREDICTION METHOD SPECIFIED BY THE ROW NAME APPLIED TO THE LINKED DATA WITH A MISMATCH RATE SPECIFIED BY THE COLUMN NAME. THE MISMATCH RATES VARY BETWEEN 0% - 40%.

	0%	10%	15%	20%
bagging trees	1.25	1.28	1.31	1.33
random forest	0.88	0.95	0.98	1.02
Adj-rf	0.88	0.92	0.94	0.97
Adj-trees	0.97	1.01	1.03	1.05
Optimal-α-bagging	1.33	1.35	1.37	1.39
Optimal-α-rf	**0.83**	0.88	0.92	0.95
Optimal-α-Adj-rf	0.84	**0.87**	**0.89**	**0.91**
Optimal-α-Adj-trees	0.96	0.97	0.98	1
mismatch rate	25%	30%	35%	40%
bagging trees	1.36	1.39	1.43	1.47
random forest	1.06	1.11	1.16	1.21
Adj-rf	1	1.03	1.07	1.11
Adj-trees	1.08	1.12	1.15	1.2
Optimal-α-bagging	1.4	1.44	1.47	1.51
Optimal-α-rf	0.99	1.03	1.08	1.13
Optimal-α-Adj-rf	**0.93**	**0.96**	**1**	**1.04**
Optimal-α-Adj-trees	1.02	1.05	1.08	1.13

TABLE V
EACH CELL VALUE IS THE MSPE OF THE PREDICTION METHOD SPECIFIED BY THE ROW NAME APPLIED TO THE LINKED DATA WITH A MISMATCH RATE SPECIFIED BY THE COLUMN NAME. THE MISMATCH RATES VARY BETWEEN 0% - 40%

Optimal-α	Mean-optimal-α	bagging trees
26.544	2.886	3.029
random forest	Adj-rf	Adj-trees
0.065	0.239	121.836

TABLE VI
THE RUNNING TIME OF EACH METHOD IN SECONDS.

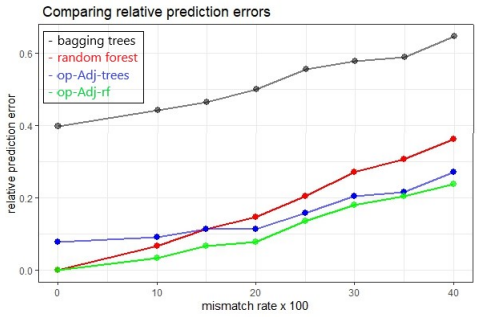

Fig. 6. Relative prediction error plot of bagging trees, random forest, optimal-α trees, and optimal-α-Adj-rf.

C. Discussion

The simulation studies in Sections III-A and III-B show that random forests and bagging trees can perform well on the exact data; however, their performance can rapidly deteriorate as mismatch rates increase. In the presence of mismatches in the data, our proposed methods are generally effective in adjusting the outputs of random forests and bagging trees and improving their performance in terms of reduction in MSE, MSPE, or bias in the estimated regression function. Combining the optimal-α method with Adj-rf or Adj-trees can be more effective than a single adjustment method. Both Optimal-α-Adj-rf and Adj-trees are computationally expensive, however, and may not be scalable for large data sets. In the case of the optimal-α method, the mean-optimal α method is a viable replacement. The mean-optimal α can be implemented much faster, and in both of our simulations, the mean-optimal α method performs almost identically to the optimal-α method. However, this may be because the mismatches in the data are created by a permutation selected at random; thus, each observation has the same probability of being correctly matched.

IV. CONCLUSIONS

The rise of social media platforms such as Twitter/X has provided social scientists with an excellent opportunity to create rich, comprehensive data sets better tailored for social research by linking social media data to survey data. In survey methodology and data science research, ensemble methods such as bagging trees, random forests, and gradient boosting are often gold-standard techniques available for predictive modeling. However, the simulation studies presented in this paper show that in the presence of mismatches arising from imperfect record linkage, the actual predictive performance of these machine learning techniques may not be realized. Our simulation studies also indicate that the proposed optimal-α and weighting-reweighting methods can efficiently improve the predictive performance of these techniques.

While we focused on bagging trees and random forests in this empirical evaluation, the methodology described in this paper can be extended to other popular machine learning approaches, such as gradient boosting and neural networks. This is one promising direction for future research and evaluation. Several other extensions of this research are also possible.

These include generalizing the adjustment methods to classification problems, allowing for more flexible distributions in modeling the conditional density of $y \mid \mathbf{x}$, and designing a more specific adjustment method for gradient boosting.

Concerning the simulation study, a recommendation for future research is a more thorough comparison between the optimal-α and mean-optimal-α methods by creating a more complex pattern for matches and mismatches, where the observations can have different probabilities of being correctly matched. In this paper, all observations had the same probability of being correctly matched. In practice, some subgroups of observations may be more difficult than others to correctly link to other data sets. For example, certain types of cases may have a higher likelihood of having missing data on key linking variables, increasing the probabilities of a mismatch for those cases (if other less-informative variables need to be used in the RL algorithm as a result). Future empirical work could consider evaluations of scenarios where the probabilities of a correct match vary across different subgroups of cases.

Finally, we also focused on *mismatches* in this study, as opposed to *missed matches*. The latter type of error in record linkage is more likely to introduce selection bias in estimates of the relationships between variables based on the linked data file, depending on the extent to which the linked records differ from the missed matches in terms of the relationships of interest. In this paper, we did not consider the possible biasing effects of missed matches on estimates of the parameters in the underlying mixture model. Future studies should focus on the development of adjustment methods that recognize whether the mechanism underlying the probability of a missed match is *ignorable* (in that missed matches can be effectively predicted as a function of other observed variables), or *non-ignorable*, in that it is a function of variables that were not observed in the two data sets being linked. Future extensions of this work should focus on adjustment methodologies that can efficiently accommodate both mismatches and missed matches.

REFERENCES

[1] G. Bello-Orgaz, J. J. Jung, and D. Camacho, "Social big data: Recent achievements and new challenges," *Information Fusion*, vol. 28, pp. 45–59, 2016.

[2] N. A. Ghani, S. Hamid, I. A. Targio Hashem, and E. Ahmed, "Social media big data analytics: A survey," *Computers in Human Behavior*, vol. 101, pp. 417–428, 2019.

[3] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of Twitter data," in *Proceedings of the Workshop on Language in Social Media (LSM 2011)*. Portland, Oregon: Association for Computational Linguistics, Jun 2011, pp. 30–38.

[4] T. H. McCormick, H. Lee, N. Cesare, A. Shojaie, and E. S. Spiro, "Using twitter for demographic and social science research: Tools for data collection and processing," *Sociological Methods & Research*, vol. 46, no. 3, pp. 390–421, 2017.

[5] J. Murphy, M. Link, H. J. Childs, L. C. Tesfaye, E. Dean, M. Stern, J. Pasek, J. Cohen, M. Callegaro, and P. Harwood, "Social media in public opinion research: Executive summary of the aapor task force on emerging technologies in public opinion research," *The Public Opinion Quarterly*, vol. 78, no. 4, p. 7, 2014.

[6] A. B. Tarek, A. Wenz, L. Sloan, and C. Jessop, "Linking twitter and survey data: asymmetry in quantity and its impact," *EPJ Data Sci.*, vol. 10, no. 1, p. 32, 2021.

[7] G. Gautam and D. Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," in *2014 Seventh International Conference on Contemporary Computing (IC3)*, 2014, pp. 437–442.

[8] Y. Wan and Q. Gao, "An ensemble sentiment classification system of twitter data for airline services analysis," in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, 2015, pp. 1318–1325.

[9] Z. Wang, E. Ben-David, G. Diao, and M. Slawski, "Regression with linked datasets subject to linkage error," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 14, no. 4, p. e1570, 2022.

[10] R. Chambers, E. Fabrizi, M. Ranalli, N. Salvati, and S. Wang, "Robust regression using probabilistically linked data," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 15, no. 2, p. e1596, 2023.

[11] H. Newcombe and J. Kennedy, "Record linkage: making maximum use of the discriminating power of identifying information," *Communications of the ACM*, vol. 5, no. 11, pp. 563–566, 1962.

[12] O. Binette and R. Steorts, "(Almost) all of entity resolution," *Science Advances*, vol. 8, no. 12, p. eabi8021, 2022.

[13] P. Christen, *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer, 2012.

[14] R. Little and D. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019.

[15] J. Neter, S. Maynes, and R. Ramanathan, "The effect of mismatching on the measurement of response error," *Journal of the American Statistical Association*, vol. 60, pp. 1005–1027, 1965.

[16] F. Scheuren and W. Winkler, "Regression analysis of data files that are computer matched I," *Survey Methodology*, vol. 19, pp. 39–58, 1993.

[17] ——, "Regression analysis of data files that are computer matched II," *Survey Methodology*, vol. 23, pp. 157–165, 12 1997.

[18] P. Lahiri and M. D. Larsen, "Regression analysis with linked data," *Journal of the American Statistical Association*, vol. 100, no. 469, pp. 222–230, 2005.

[19] Y. Han and P. Lahiri, "Statistical analysis with linked data," *International Statistical Review*, vol. 87, pp. 139–157, 2019.

[20] M. Hof and A. Zwinderman, "A mixture model for the analysis of data derived from record linkage," *Statistics in Medicine*, vol. 34, pp. 74–92, 2015.

[21] R. Gutman, C. Afendulis, and A. Zaslavsky, "A Bayesian Procedure for File Linking to Analyze End-of-Life Medical Costs," *Journal of the American Statistical Association*, vol. 108, pp. 34–47, 2013.

[22] N. Dalzell and J. Reiter, "Regression Modeling and File Matching Using Possibly Erroneous Matching Variables," *Journal of Computational and Graphical Statistics*, vol. 27, pp. 728–738, 2018.

[23] A. Tancredi and B. Liseo, "Regression analysis with linked data: problems and possible solutions," *Statistica*, vol. 75, no. 1, pp. 19–35, 2015.

[24] R. C. Steorts, A. Tancredi, and B. Liseo, "Generalized Bayesian Record Linkage and Regression with Exact Error Propagation," in *International Conference on Privacy in Statistical Databases*, 2018, pp. 279–313.

[25] J. Abowd, J. Abramowitz, M. Levenstein, K. McCue, D. Patki, T. Raghunathan, A. M. Rodgers, M. Shapiro, and N. Wasi, "Optimal probabilistic record linkage: Best practice for linking employers in survey and administrative data," U.S. Census Bureau, Center for Economic Studies, Working Papers, 2019.

[26] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2023. [Online]. Available: https://www.R-project.org/

[27] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[28] ——, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[29] J. N. Rao and C. Wu, "Resampling inference with complex survey data," *Journal of the american statistical association*, vol. 83, no. 401, pp. 231–241, 1988.

[30] S. Kolenikov, "Resampling variance estimation for complex survey data," *The Stata Journal*, vol. 10, no. 2, pp. 165–199, 2010.

[31] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in c++ and r," *Journal of Statistical Software*, vol. 77, no. 1, pp. 1–17, 2017.

[32] T. Therneau and B. Atkinson, "rpart: Recursive Partitioning and Regression Trees for R 4.1.19," 2022, https://CRAN.R-project.org/package=rpart.

[33] M. Slawski, G. Diao, and E. Ben-David, "A pseudo-likelihood approach to linear regression with partially shuffled data," *Journal of Computational and Graphical Statistics*, vol. 30, no. 4, pp. 991–1003, 2021.