

Probabilistic Matching: Causal Inference under Measurement Errors

Fani Tsapeli

School of Computer Science
University of Birmingham
Birmingham, UK

Email: t.tsapeli@cs.bham.ac.uk

Peter Tino

School of Computer Science
University of Birmingham
Birmingham, UK

Email: P.Tino@cs.bham.ac.uk

Mirco Musolesi

Department of Geography
University College London
London, UK

Email: m.musolesi@ucl.ac.uk

Abstract—The abundance of data produced daily from large variety of sources has boosted the need of novel approaches on causal inference analysis from observational data. Observational data often contain noisy or missing entries. Moreover, causal inference studies may require unobserved high-level information which needs to be inferred from other observed attributes. In such cases, inaccuracies of the applied inference methods will result in noisy outputs. In this study, we propose a novel approach for causal inference when one or more key variables are noisy. Our method utilizes the knowledge about the uncertainty of the real values of key variables in order to reduce the bias induced by noisy measurements. We evaluate our approach in comparison with existing methods both on simulated and real scenarios and we demonstrate that our method reduces the bias and avoids false causal inference conclusions in most cases.

I. INTRODUCTION

Nowadays, there is an increasing data availability. Smartphones and wearables sensors, social media, web browsing information and sales recordings are only few of the newly available information sources. Discovering dependencies and patterns within such datasets could provide useful insights to businesses or social scientists studying human behavior.

An important part of such studies involves the discovery of correlation or causation links among factors of interest. For example, several studies examine the impact of sentiment and opinions expressed through social media to traded assets prices [1], [2], [3]. Others study the impact of location, activity and communication patterns on people mood [4], stress level [5] or eating and sleeping disorders [5] by processing smartphones sensor data. In such studies, the variables of interest are usually not directly measured. Instead, they are inferred from raw data. For example, in [1], [3] a sentiment index is inferred from Twitter data by applying text processing and classification techniques. However, such procedures result in inaccurate estimation of the variables of interest. Moreover, several studies have shown that social media data in some countries have undergone censorship [6]. In such cases, sentiment or opinion tracking could be biased. Also, in [5], location context (e.g. home, work, entertainment place etc.) is used to understand the causal link between location and stress level. However, the real location context is not known; instead it is inferred from smartphone sensors and consequently it could be inaccurate.

Causal discovery when key variables are unobserved or inaccurately measured is a particularly challenging task. Latent variable models have been used to handle such cases [7], [8]. Such models include one or more unobserved or latent variables. Scientists usually attempt to estimate the values of a latent variable from other observed variables by fitting the data in a structural equation model [7]. However, the selection of a proper model is a complex task which may result in misspecification and overfitting.

Matching design [9] is an alternative causal inference approach which does not require fitting data to a structural equation model. The main idea is that the average impact of a treatment variable (or predictor) to an outcome or effect variable can be approximated by comparing the outcome variable values of units with similar characteristics and different treatment values. In order to minimize any bias due to differences on units characteristics (*confounding bias*), matching methods attempt to find optimal pairs of units. When the values of these characteristics are inaccurately measured or inferred from other variables, matching may fail to sufficiently eliminate any confounding bias which may result in misleading conclusions about the causal relationship of the examined variables.

In this work, we propose *probabilistic matching*, a matching method that takes into account the uncertainty about the real values of a noisy variable and attempts to find optimal pairs of units in order to maximize the probability that the matched units have similar characteristics. Our method is based on the assumption that a probability distribution describing the real values of each unobserved variable is known or can be approximated. Although this assumption may restrict the applicability of the proposed method, it is realistic in many scenarios. For example, when an inference procedure is applied in order to learn the values of an unobserved variable L from some observed attributes C , a probability distribution $Pr(L|C)$ can be approximated, as we discuss later in Section IV.

We evaluate the proposed matching framework on two different simulation studies in comparison with a conventional matching method. We demonstrate that probabilistic matching reduces significantly the confounding bias and results in more accurate causal conclusions. We also evaluate our method on a real dataset. In particular, we use the social media dataset described in [10] in order to test whether

text messages containing URLs tend to be reposted more often. This dataset includes a rich variety of features extracted from Weibo microblogging service for 111 users along with a manually assigned binary label for each user indicating whether he/she has been characterized as *spammer* or not. In our scenario, we assume that the *spammer* label is an unobserved confounding variable and we apply a spammer detection method [11] in order to infer a label for each user from other observed attributes. We map the classification outputs to probability distributions describing the probability of a user to be a *spammer* and we use these probability distributions to our matching framework. We demonstrate that the number of URLs in text messages indeed influences the number of reposts. We repeat our causality study by applying a conventional matching method in two scenarios: 1) the ground truth binary *spammer* identifier is known and 2) only the noisy *spammer* identifier inferred from the data is known. The results of the first scenario serve as the ground-truth. We demonstrate that our results come in agreement with the conclusions of the first scenario, while the examined conventional matching method fails to detect the causal link.

II. BACKGROUND

In this section we provide some background knowledge on causal inference based on the matching design framework. For more clarity, we summarize the notation used in the paper in Table I. According to Rubin’s framework [12], the causal impact of a binary treatment on a unit u can be assessed by comparing the outcome $Y_1(u)$, if the unit has received the treatment, with the outcome $Y_0(u)$, if the unit has not received the treatment. The *fundamental problem of causal inference* is that it is not possible to observe both $Y_1(u)$ and $Y_0(u)$ for the same unit u . If treatment is randomly assigned, the average treatment effect (ATE) can be estimated as $E\{Y_1\} - E\{Y_0\}$, where $E\{Y_1\}$ and $E\{Y_0\}$ are expectations w.r.t. uniform distribution over treated and untreated units, respectively. However, if the treatment assignment is conditional on certain characteristics of the units, the causal inference could be biased. The characteristics of units are called *confounding variables*. Matching methods attempt to eliminate this bias by comparing the outcome values of units with similar observed characteristics. In particular, if U is a set of treated units and V is a set of control units (i.e. units which have not received the treatment), matching methods match each treated unit $u \in U$ with the “most similar” control unit $v \in V$. If G is the set of matched pairs of units, the average treatment effect is estimated as $E_{(u,v) \in G}\{Y_1(u) - Y_0(v)\}$, where the expectation is with respect to uniform probability distribution over G . The (dis)similarity between units is measured as a distance between their confounding variable values (for some metric).

A. Matching Methods

Several methods for creating pairs of units $(u, v) \in G$ have been proposed. The matching methods involve two steps: 1) the matching and 2) the balance check. In the matching step, a method that creates pairs of treated and control units $(u, v) \in$

| Symbol | Description |
|-----------------------------|---|
| N | Number of units |
| P | Number of confounding variables |
| Y | Outcome variable, described with a $1 \times N$ vector |
| y_u | The outcome value of unit u |
| X | Treatment variable, described with a $1 \times N$ vector |
| x_u | The treatment value of unit u |
| Z | $P \times N$ matrix of confounding variables |
| z_u | the u^{th} column of Z , denoting a $P \times 1$ vector of values of unit u for the P confounders |
| z^p | the p^{th} line of Z , denoting a $1 \times N$ vector of values of the N units for the p^{th} confounder |
| z_u^p | element in column u and line p of Z , denoting the value of unit u for the p confounder |
| G | Set of matched treated and control units |
| G_U | Set of matched treated units |
| G_V | Set of matched control units |
| \tilde{L} | Variable with measurement errors, described with a $1 \times N$ vector |
| L_u | Random variable with $Pr(L_u \tilde{L} = \tilde{l}_u)$ |
| X_u | stochastic variable describing the treatment of u |
| \mathbf{X} | $1 \times N$ vector of stochastic variables describing the treatments of the N units |
| \mathbf{Z} | $P \times N$ matrix of stochastic confounders |
| Z_u | u^{th} column of \mathbf{Z} , denoting a $P \times 1$ vector of random variables for unit u |
| Z^p | p^{th} line of \mathbf{Z} , denoting a $1 \times N$ vector of random variables for the p confounder |
| Z_u^p | element in column u and line p of \mathbf{Z} , denoting a random variable for the p^{th} confounder of unit u |
| $\mathcal{D}(z_u, z_v)$ | Distance between vectors z_u, z_v |
| $\mathcal{D}(Z_u^p, Z_v^p)$ | Distance between random variables Z_u^p, Z_v^p |
| $\mathbf{D}(Z_u, Z_v)$ | Distance between random variables vectors Z_u, Z_v |
| \mathbb{D}_{z_u, z_v} | $P \times 1$ vector of distances between the P random variables Z_u^p, Z_v^p |
| $\Delta(u, v)$ | Distance between units u, v |

TABLE I: Notation.

G based on closeness of their confounding variables is applied using some notion of a distance between confounding variable values. Afterwards, in the balance check step, the remaining confounding bias due to imperfectly matched units needs to be estimated. If the resulted groups of matched treated and control units are not adequately balanced, the matching method needs to be revised. The balance can be examined by checking the standardized mean difference between the treated and control units, by applying a t-test or a Kolmogorov-Smirnov test, or by examining the quantiles of the matched units. This checking has to be done for each confounding variable.

The simplest matching method is the *Nearest Neighbor Matching* which matches each treated unit to the control unit with the lowest distance between the corresponding confounding variable values. Another popular matching method is *genetic matching* [13]. Let us denote by N the number of units in the study, P the number of confounding variables and Z a $P \times N$ matrix of confounding variables with z_u the u^{th} column of Z , i.e. $P \times 1$ column vector of values of unit u for the P confounders. Genetic matching uses as distance

metric between confounder vectors the following weighted Mahalanobis distance:

$$d_{u,v,W} = \sqrt{(z_u - z_v)^T \cdot \mathbf{W} \cdot (z_u - z_v)} \quad (1)$$

where $\mathbf{W} = (S^{-\frac{1}{2}})^T \cdot W \cdot S^{-\frac{1}{2}}$, with \mathbf{W} a $P \times P$ diagonal positive definite weight matrix and $S^{-\frac{1}{2}}$ is the Cholesky decomposition of the sample covariance matrix of $Z = [z_1, \dots, z_N]$. The diagonal elements of W are selected by applying an evolutionary search algorithm that attempts to find the optimal weights to minimize a loss function. Several loss functions can be used. A commonly used loss is the minimum p-value of a t-test or a Kolmogorov-Smirnov distributional test on the matched pairs of treated and control units resulting from applying a given W in the distance calculations between confounders. The loss is calculated for each confounding variable. Thus, if p_p is the p-value of the p^{th} confounding variable, the objective is to find a matrix W that minimizes the $\min_p p_p$. Other loss functions are based on comparisons of the quantiles of confounding variables for the matched treated and control units. In detail, denote by G_U and G_V the sets of matched treated and control units, respectively, i.e. for each pair $(u, v) \in G$, $u \in G_U$ and $v \in G_V$. For p^{th} confounding variable, we think of the corresponding values for matched treated units $\{z_u^p : u \in G_U\}$ as realizations of a random variable A^p . Analogously, the values $\{z_v^p : v \in G_V\}$ of matched control units will be considered realizations of a random variable B^p . Given a set of K quantiles $a^p(k)$ and $b^p(k)$ of A^p and B^p , respectively, we calculate a set of quantile differences $\Delta^p = \{|a^p(k) - b^p(k)|\}_{k=1}^K$. Then, one of the following loss functions can be applied: 1) $\text{mean}_p \text{mean} \Delta^p$, 2) $\text{max}_p \Delta^p$, 3) $\text{median}_p \Delta^p$, 4) $\text{mean}_p \text{max} \Delta^p$, 5) $\text{max}_p \text{max} \Delta^p$, 6) $\text{median}_p \text{max} \Delta^p$, 7) $\text{mean}_p \text{median} \Delta^p$, 8) $\text{max}_p \text{median} \Delta^p$ and 9) $\text{median}_p \text{median} \Delta^p$.

B. Matching with Continuous Treatments

Although matching frameworks have been proposed mainly for bivariate treatment variables, some recent studies also consider continuous treatments [14], [15]. In such cases units cannot be split into treatment and control groups. Instead, each unit can be matched to any other unit. The goal of matching is to create pairs of units with similar values on their confounding variables but different treatment values. In [14] the distance between units u, v is estimated as follows:

$$\Delta(u, v) = \frac{\mathcal{D}(z_u, z_v) + \epsilon}{(x_u - x_v)^2} \quad (2)$$

where $\mathcal{D}(z_u, z_v)$ is the distance between the vectors of confounding variables values of units u and v (this can be the euclidean distance, the Mahalanobis distance or any other distance metric), $\epsilon > 0$ a small constant and x_u, x_v are the treatment values of u and v , respectively. With respect to unit v , unit u will be considered as treated if $x_u > x_v$. The average treatment effect is estimated as follows:

$$E_{(u,v) \in G} \left\{ \frac{y_u - y_v}{x_u - x_v} \right\} \quad (3)$$

C. Genetic Matching with Continuous Treatments

To the best of our knowledge, Genetic Matching, so far has been used only for binary treatments. However, it can be extended to continuous treatments by modifying Eq. (2) as follows:

$$\Delta(u, v) = \frac{d_{u,v,W} + \epsilon}{|x_u - x_v|} \quad (4)$$

The loss function also needs to be modified in order to penalize any matrix W which results in matched units with similar treatments. We think of the absolute differences on the p^{th} confounding variable values of the matched treated and control units $\{|z_u^p - z_v^p| : (u, v) \in G\}$ as realizations of a random variable A^p . Then, we define a set of K quantiles $\Delta^p = \{q^p(k)\}_{k=1}^K$. The loss function can be selected based on this quantiles set as described in Section II-A.

III. PROBABILISTIC MATCHING

We further extend the framework for matching with continuous treatments to cases where treatment and/or one or more confounding variables may have noisy or censored measurements. We assume that for each unobserved variable L there is an observed noisy version \tilde{L} . For example, \tilde{L} could be a location label inferred from smartphone sensor data (and consequently subject to inaccuracies) and L the real unknown location label. We also assume that for each observation \tilde{l}_u of \tilde{L} the corresponding random variable L_u has known probability distribution $Pr(L_u | \tilde{L} = \tilde{l}_u)$. In the following, we will consider the general case where all the key variables are noisy with the understanding that in the case of no noise the corresponding distribution reduces to the delta function:

$$Pr(L_u | \tilde{L} = \tilde{l}_u) = \begin{cases} 1 & , L_u = \tilde{l}_u \\ 0 & , L_u \neq \tilde{l}_u \end{cases} \quad (5)$$

Denote by X_u the random variable describing the treatment of unit u and with \mathbf{X} a $1 \times N$ random vector of treatment variables of all units. We also denote by Z_u^p the random variable describing the p^{th} confounding variable of unit u and with \mathbf{Z} a $P \times N$ matrix of random variables Z_u^p . As before, Z^p will denote the p^{th} row of \mathbf{Z} and Z_u its u^{th} column. Our objective is to find pairs of units with minimum distance Δ as given in Eq. (2). However, if the treatment and/or any of the confounding variables are noisy, the real distance cannot be calculated. Consequently, the applied matching method may result in poor matches. We attempt to improve the matching by including our knowledge about the uncertainty of the variables into the matching process. Suppose we have a notion of a distance $D(X_u, X_v)$ between random variables X_u, X_v and a distance $\mathbf{D}(Z_u, Z_v)$ between random vectors Z_u and Z_v . We need to find pairs of units u, v that minimize

$$\Delta(u, v) = \frac{\mathbf{D}(Z_u, Z_v)}{D(X_u, X_v)} \quad (6)$$

We need to define a suitable distance metric D for our random variables. Commonly used distance metrics for distributions such as f-divergence metrics (e.g. Kullback-Leibler divergence) are not suitable in our case, since our objective is to estimate the probability that the values of two random variables X_u, X_v are close (i.e. $Pr(|X_u - X_v| < \epsilon)$, where ϵ a small positive constant). Since our distance metric needs to measure also the proximity between the values of two random variables, we suggest a metric that is based on comparison of the quantiles of the examined variables. Let us denote by $q_{X_u}(k)$ the k^{th} quantile of variable X_u , $k = 1, 2, \dots, K$. Then, we define $D(X_u, X_v)$ as follows:

$$D(X_u, X_v) = \frac{1}{K} \cdot \sqrt{\sum_k (q_{X_u}(k) - q_{X_v}(k))^2} \quad (7)$$

If X is not noisy, the quantile values will be the same and $D(X_u, X_v)$ reduces to the Euclidean distance of x_u and x_v .

A. Probabilistic Genetic Matching

Although several distance metrics can be used as the distance between random vectors $\mathbf{D}(Z_u, Z_v)$, in this work, we propose Probabilistic Genetic Matching (*ProbGenMatch*), a modified version of the Genetic Matching distance metric. Denote by $\mathbb{D}_{Z_u, Z_v} = [D(Z_u^1, Z_v^1), D(Z_u^2, Z_v^2), \dots, D(Z_u^P, Z_v^P)]^T$ the $P \times 1$ vector of distances $D(Z_u^p, Z_v^p)$ between the P random variables Z_u^p, Z_v^p , $p = 1, 2, \dots, P$ (see Eq. (7)). Then, we calculate $\mathbf{D}(Z_u, Z_v)$ by modifying the Genetic Matching distance of Eq. (1) as follows:

$$\mathbf{D}(Z_u, Z_v) = \sqrt{\mathbb{D}_{Z_u, Z_v}^T \cdot (S^{-\frac{1}{2}})^T \cdot W \cdot S^{-\frac{1}{2}} \cdot \mathbb{D}_{Z_u, Z_v}} \quad (8)$$

The loss function used to select the optimal weight matrix W also needs to be modified. We use the quantiles-based loss functions described in Section II-A. In particular, for each pair of units $(u, v) \in G$ we define a random variable:

$$\mathcal{A}_{u,v}^p = \frac{|Z_u^p - Z_v^p|}{|X_u - X_v|} \quad (9)$$

We denote by $\alpha_{u,v}^p(k)$ the k^{th} quantile of $\mathcal{A}_{u,v}^p$. We also define the average k -th quantile for the p^{th} confounding variable, $\bar{\alpha}^p(k) = \frac{1}{|G|} \cdot \sum_{(u,v) \in G} \alpha_{u,v}^p(k)$. Finally, we collect the average quantiles in the set $\Delta^p = \{\bar{\alpha}^p(k)\}_{k=1}^K$ to be used in a quantile-based loss functions described in Section II-A.

B. Implementation

ProbGenMatch has been implemented as an R package and it is based on the *Matching* R package, an open source software which implements several matching methods. *ProbGenMatch* takes as input the probability distributions for all the confounding variables and treatment variable for all the units of the study (along with other optional parameters) and returns the matched pairs according to the previously described framework. If all the variables of the study are observed without any measurement errors, then *ProbGenMatch* is equivalent to the continuous Genetic Matching approach that we

presented in Section II-B. If also the treatment variable is binary, *ProbGenMatch* is equivalent to the Genetic Matching framework [13].

Caliper Distance. Matching with caliper distance has been previously proposed as a way to impose restrictions on the maximum allowed dissimilarity between the matched units [16]. A caliper distance is simply a threshold that defines the maximum allowed difference of two units on their confounding variable values. In our implementation we also support matching with caliper distance as an optional parameter. For stochastic confounding variables, a probability threshold T_{prob} should be provided along with the caliper distance. This probability threshold allows the matching of two units only if the probability to have a larger difference than the caliper distance on their confounding variable values is smaller than T_{prob} . Our implementation also allows users to specify a threshold on the minimum difference between the treatment values of two matched units. For stochastic treatment variables, a probability threshold should be provided along with the minimum treatment difference threshold.

IV. EVALUATION

We evaluate the proposed probabilistic matching framework on two synthetic and one real dataset. All the examined scenarios include one unobserved variable L along with an observed noisy version \tilde{L} . We use as baselines for our evaluation:

- 1) the traditional Genetic Matching (*GenMatch*) approach which treats \tilde{L} as the *true* variable.
- 2) the *optimal* Genetic Matching (*OptGenMatch*), where we assume that L is observed without any noise. The performance of Genetic Matching under this optimal scenario serves as an upper bound to the performance of our method. The results obtained by *OptGenMatch* will be considered as ground-truth.

We use the synthetic datasets to evaluate the performance of the proposed framework on different noise levels. We also examine the sensitivity of our approach to the selected caliper distance. Finally, we apply our method on the social media dataset described in [10] in order to test whether text messages containing URLs tend to be reposted more often.

Evaluation Criteria. The objective of a matching method is to match units with small dissimilarity on their confounding variables values and large difference on their treatment values. We evaluate the performance of *ProbGenMatch* on these metrics in comparison with *GenMatch* and *OptGenMatch*. Additionally, we investigate whether the amount of the remaining bias due to purely matched pairs is sufficiently large to influence the validity of the causal inference by comparing our conclusions with the corresponding results of *OptGenMatch*.

A. Synthetic Dataset

We consider a binary variable L describing the class of objects represented by M -dimensional vectors of real numbers. We consider two types of vectors. The first type corresponds to positive examples (i.e. vectors that belong to class $L = 1$). The data in each of the M dimensions of the first vector type

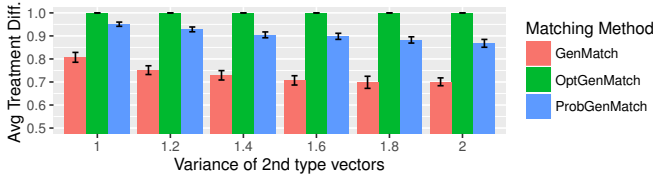


Fig. 1: Average treatment difference between the matched units.

are generated by a Gaussian process with mean value 1 and standard deviation σ_1 . The second type of vectors corresponds to negative examples (i.e. $L = 0$) and their values in each dimension are generated by a random Gaussian process with mean -1 and standard deviation σ_2 . We train a Support Vector Machine classifier on this synthetic dataset and afterwards we use the classifier on unseen synthetic data (generated with the same procedure) in order to learn a label \tilde{L} for each vector. Afterwards, we map the SVM outputs into probabilities by applying the process described in [17]. For each vector v , the probability distribution of random variable L_v corresponds to the output of this mapping procedure.

In our test case, we consider two-dimensional vectors (i.e. $M = 2$) and we set $\sigma_1 = 1$. We test the performance of our matching framework with different noise levels on the observed variable \tilde{L} by increasing σ_2 from 1 to 2 with step 0.2. By increasing the variance of the second vector type, we make our vectors less separable and consequently, the resulted classes \tilde{L} are less accurate.

1) *Unobserved Treatment Variable*: In the first case, we consider L as the treatment variable. We generate two confounding variables $Z^1 = \alpha_1 \cdot L + e_1$ and $Z^2 = \alpha_2 \cdot L + e_2$, where e_1 and e_2 random Gaussian noise with mean 0 and variance 1 and 2 respectively for Z^1 and Z^2 and α_1, α_2 are model coefficients. In the following results, we do not use a caliper distance for the confounding variables. We set the minimum allowed distance between the treatments of matched units equal to 0.1 and the maximum allowed probability that the matched units have a treatment difference larger than 0.1 equal to 0.25. We repeat our study for 10 randomly selected sets of model coefficients (α s). All model coefficients are randomly generated from a uniform distribution on $[0, 1]$. For each one of the 10 sets of model coefficients we repeat each study for 100 different noise realizations. In Fig. 1 we present the average treatment difference between the matched units for the three examined matching algorithms along with the 95% confidence intervals. The *OptGenMatch* method always avoids matching units with the same treatment value. Thus, given that in this scenario we consider binary treatments, the average treatment difference is always equal to 1. According to our results, the performance of both *GenMatch* and *ProbGenMatch* declines for higher noise levels (i.e. larger σ_2). However, *ProbGenMatch* significantly outperforms *GenMatch* by avoiding matching units with the same treatment for more than 88% of the matched pairs for all examined noise levels.

When the resulted group of matched units contains pairs

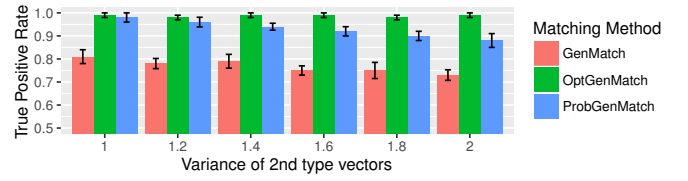


Fig. 2: Percentage of true positive causality conclusions.

with the same treatment level, the impact of the examined treatment on the outcome variable cannot be reliably assessed by comparing the matched units on their outcome values. We demonstrate this by generating the following outcome variable:

$$Y = \beta_0 \cdot L + \beta_1 \cdot Z^1 + \beta_2 \cdot Z^2 + n_u + e_n \quad (10)$$

where $\beta_0, \beta_1, \beta_2$ are model coefficients, n_u is a uniform random variable on $[0, 4]$ and e_n is gaussian noise with mean 0 and variance 1. All β coefficients are randomly generated from a uniform distribution on $(0, 1]$. For non-zero β_0 , the treatment variable L has a causal impact on Y . We apply a Wilcoxon non-parametric test in order to examine whether the average treatment effect, (Eq. (3)) is significantly different than zero. When the performance of *OptGenMatch* is examined, we use as treatment (i.e. the variable X of Eq. (3)) the binary variable L , while for *GenMatch* and *ProbGenMatch* we use the noisy variable \tilde{L} . We repeat our study for 10 different sets of model coefficients and 100 realizations of n_u, e_n , for all the groups of matched units resulted after applying the three examined methods, as it was previously described. In Fig. 2 we depict the average percentage of times that the null hypothesis of the Wilcoxon test (i.e. that the average treatment effect is equal to zero) was rejected with p-value equal to 0.05. *OptGenMatch* successfully detects the causal impact of L on Y in most cases, while *ProbGenMatch* significantly outperforms *GenMatch*.

2) *Latent Confounding Variable*: In the second case, L corresponds to a binary confounding variable. In detail, we consider a continuous treatment variable X which follows a uniform distribution on $[0, 1]$. Our binary confounding variable L follows a binomial distribution with success probabilities given by the vector of probabilities $P_S = 1/(1 + e^{-t})$, where $t = \alpha_0 + \alpha_1 \cdot X$. We also create a confounding variable $Z^1 = \alpha_1 \cdot X + e_1$. We evaluate the performance of the three examined matching approaches by generating different realizations of the model coefficients and noise e_1 , as it was previously described in Section IV-A1. We assess the remaining bias due to imperfect matches by calculating the standardized difference in means for each confounding variable [18]. In detail, for the binary confounding variable L , we consider the values $\{\frac{l_u}{x_u - x_v} : (u, v) \in G\}$ as realizations of a random variable C_U and the values $\{\frac{l_v}{x_u - x_v} : (u, v) \in G\}$ as realizations of a random variable C_V . Then the standardized difference in means for the confounding variable L is estimated as:

$$\frac{|\bar{C}_U - \bar{C}_V|}{\sqrt{(\sigma_U^2 + \sigma_V^2)/2}} \quad (11)$$

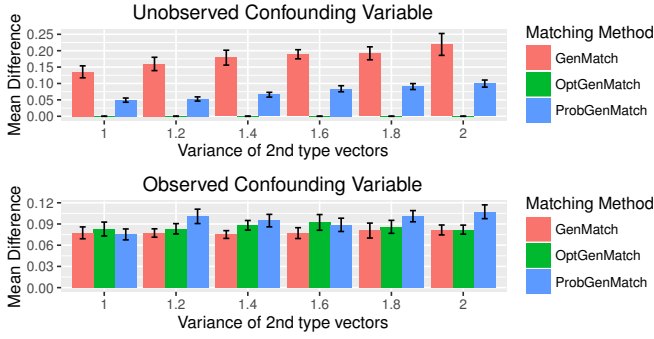


Fig. 3: Remaining bias for the two confounding variables.

where \bar{C}_U , \bar{C}_V the mean values of C_U , C_V respectively and σ_U^2 , σ_V^2 their variances. The same process is followed for the estimation of the standardized difference in means for Z^1 .

In Fig. 3 we present the standardized difference in means for the two confounding variables (i.e. the binary variable L on the top and the continuous Z^1 on the bottom). *OptGenMatch* always matches units with the same value on L and therefore, there is zero bias. The proposed *ProbGenMatch* method achieves also low bias, smaller than 0.1 for all the noise levels and significantly outperforms *GenMatch*. Finally, all methods achieve similar performance on the continuous confounding variable Z^1 , which is considered to be observed without any noise, although the performance of *ProbGenMatch* is slightly worse for large noise levels.

Failing to sufficiently eliminate the bias induced by confounding variables may result in false positive causality conclusions. We demonstrate this by considering again the outcome variable of Eq. (10). This time we set $\beta_1 = 0$, thus, there is no causal impact of Z^1 on Y . In Fig. 4 we present the rate of the false positive causality conclusions (i.e. the average percentage of times that the null hypothesis of the Wilcoxon test was not rejected) along with the 95% confidence interval. *ProbGenMatch* achieves up to 8% lower false positive rate than *GenMatch*.

B. Location-based Synthetic Dataset

In this scenario, our latent variable L represents the daily time that the participants of a study spend in entertainment venues such as pubs, restaurants, bars etc. We assume a study based on smartphones sensor data, where participants do not report their location; instead location, along with the underlying context (i.e. work, home, restaurant etc.) is inferred from other raw sensor data. Several methods for automatic location label inference have been proposed [19], [20]. However, the real location context cannot be inferred accurately. Location context could be very important for studies examining the impact of social behavior or daily activities (e.g. exercising socializing etc.) on well-being indicators such as stress level [5] or eating disorders [21].

We synthetically generate a location dataset based on the description of the real dataset presented in [19]. In [19], authors gather several sensor data along with ground truth

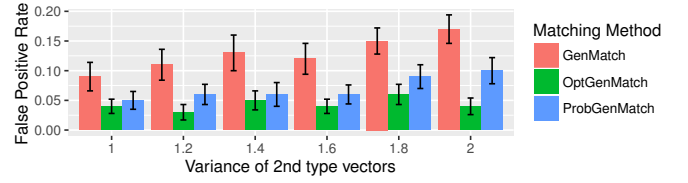


Fig. 4: Percentage of false positive causality conclusions.

labels for the locations of 36 participants and they apply a method for automatic location label inference. In order to generate our dataset, we define a variable P denoting a location label. As described in [19], we consider 7 location labels: *home*, *work*, *college*, *entertainment*, *food*, *shops* and *other*. $P_u(t)$ denotes the location of a participant at day u and time t and it is sampled on an hourly basis. We use the mobility patterns and location statistics described in [19] in order to randomly generate the variable $P_u(t)$. We also define a variable $E_u(t)$ as follows:

$$E_u(t) = \begin{cases} 1 & , P_u(t) = \textit{entertainment} \\ 0 & , \textit{otherwise} \end{cases} \quad (12)$$

Finally, we create a variable L , with values $l_u = \sum E_u(t)$ for each day u . However, in a real study, where participants would probably be unwilling to continuously provide labels for their location data, L would be a latent variable. We generate the discrete variable $\tilde{P}(t)$ denoting the inferred location label based on the method described in [19] by utilizing the confusion matrix (Table 3 of [19]) which presents the performance of the proposed location inference method. According to this matrix, only 41% of the places with resulted label *entertainment* are correctly labeled while the rest 59% of the places actually correspond to *college* (4%), *work* (4%), *shops* (4%), *food* (33%) and *others* (9%). We create a noisy variable $\tilde{P}(t)$ by randomly inserting bias on $P(t)$ based on these results. Then we define $\tilde{E}_u(t)$ as:

$$\tilde{E}_u(t) = \begin{cases} 1 & , \tilde{P}_u(t) = \textit{entertainment} \\ 0 & , \textit{otherwise} \end{cases} \quad (13)$$

We also create \tilde{L}_u with values $\tilde{l}_u = \sum \tilde{E}_u(t)$ for each day u . Finally, based on the performance of the location inference method, we create a random variable L_u with probability distribution $Pr(L_u | \tilde{P}_u(1), \tilde{P}_u(2), \dots, \tilde{P}_u(24))$. We normalize L , \tilde{L} to $[0, 1]$.

We use L as the unobserved treatment variable and we generate the confounding variables Z^1 , Z^2 as it is described in Section IV-A1. In this scenario, we examine the impact of the allowed minimum treatment difference on the three examined matching methods. In detail, let us denote with T_{min} the minimum allowed treatment distance. We vary T_{min} from 0.05 to 0.4 with 0.05 step. For *ProbGenMatch* we set the maximum allowed probability that the treatment difference is smaller than T_{min} equal to 0.25. In Fig. 5 we present the average treatment difference between the matched treated and control

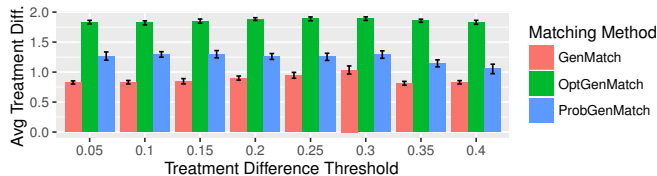


Fig. 5: Average Treatment Difference between the matched units.

groups achieved by the three examined matching algorithms. According to our results, there is not significant impact of the treatment difference threshold on the average treatment difference when the *OptGenMatch* method is applied. There is an improvement on the performance of *GenMatch* for the threshold values 0.2 to 0.3, however its performance is decreased for larger than 0.3 thresholds. Since the threshold is applied on the observed noisy variable \tilde{L} and not on L , large threshold values may prevent the matching of units which are actually good matches. *ProbGenMatch* is not strongly influenced by the treatment difference threshold, however, its performance is also decreasing for large threshold values.

Finally, we generate again an outcome variable as described in Eq. (10) in order to examine the influence that the resulted matching may have on a causality study. We examine the rate of true positive causality conclusions for the three examined methods by repeating the process described in Section IV-A1 and we present our results in Fig. 6. *ProbGenMatch* achieves a higher rate of true positive conclusions compared to *GenMatch*, however their difference is less significant compared to the binary treatments case examined in Section IV-A1. This is reasonable considering that for binary noisy treatments matching will result more often on pairs with the same treatment value; thus, the treatment effect will be weaker.

C. Social Media Dataset

In this section, we evaluate our method on a real dataset. We use the microblogPCU dataset, which is available in the UCI Machine Learning repository [10], in order to examine whether the number of URLs included in microblog messages influences the number of times that these messages are re-posted. MicroblogPCU dataset has been collected from the sina Weibo microblog and contains information about the profiles of 782 users, their social network and their microblog activity. It also contains ground-truth binary labels indicating whether a user is a *spammer* or not for 111 users.

We use the ratio of messages with URLs as the treatment variable of our study and the number of re-posts as the outcome variable. Spammers tend to use more URLs in their messages and spammers messages are re-posted less often. Thus the spammer binary indicator should be used as a confounding variable. We also use as confounding variables other indicators that correlate both with the treatment and the outcome variables. In detail, we found that the number of posts, the class level of the user account (this is an indicator

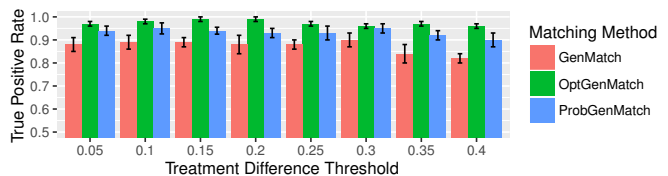


Fig. 6: Percentage of True Positive Causality Conclusions.

assigned by Weibo) and the number of followers correlate both with our treatment and outcome variables.

We assume that the binary spammer indicator is unknown and it needs to be inferred from the data. We apply the method described in [11] in order to classify the users to spammers and non-spammers. We extract attributes from the text content and users profiles as described in [11]. We use all the attributes of [11] apart from the number of times a user replied to a message or received a reply and whether a message is a *reply* message, since this information is not provided in this dataset. Also, instead of the user account age, we use the user account class. The interested reader should refer to [11] for a complete list of all the extracted features. Afterwards, we apply the chi-squared feature selection method in order to find the most important attributes. Six attributes were selected, namely: 1) the fraction of tweets with URLs, 2) the user account class, 3) the average number of URLs per message, 4) the number of followees, 5) the average number of hashtags per message and 6) the average number of re-posts. Following the procedure of [11], we use weka [22] to train a support vector machine classifier. We used only the data for the 111 users for which a ground-truth label is available. We used 50% of the dataset for training and the rest for testing. 76% of the spammers and 82% of the non-spammers are correctly classified. Our classifier is more successful on recognizing the spammers and less on recognizing the non-spammers compared to [11]. This difference can be attributed to the differences between the dataset characteristics of the two studies.

We define as L the ground-truth binary label indicating whether a user is spammer and \tilde{L} the inferred label based on the above-mentioned process. We also create a random variable L_u for each user u and we obtain a probability distribution for each L_u by mapping the SVM outputs into probabilities. We match the users based on their confounding variables values by applying the three examined approaches. The results obtained by *OptGenMatch* serve as the ground-truth. Finally, we use the Wilcoxon test to examine whether the mean value of the outcome variable for the treated units significantly differs from the mean outcome value of the control units. In Table II we present the mean difference (see eq. (11)) achieved for the binary confounding variable L with the 3 examined methods. We also present the p-values of the Wilcoxon test under the null hypothesis that the treatment variable has no effect. Both *OptGenMatch* and *ProbGenMatch* reject the null hypothesis with p-value smaller than 0.05. However, when the treatment and control pairs are created

| | Balance on L | Wilcoxon test p-value |
|---------------------|--------------|-----------------------|
| OptGenMatch | 0.014 | 0.005 |
| GenMatch | 0.36 | 0.15 |
| ProbGenMatch | 0.15 | 0.041 |

TABLE II: Causality Study Results

by applying the *GenMatch* method, the remaining bias on the binary indicator L is large and results in the false conclusion that there is no significant impact of the number of URL's included in text messages to the number of re-posts.

V. RELATED WORK

Causal inference when important variables are missing is mainly based on structural equation models (SEMs) with latent variables [7], [8]. SEMs include two components: the structural model part describing the causal relationships between the predictors and the outcome variables and the measurement model describing the relationships of the latent variables with other observed variables. Model selection is based on theoretical assumptions about data structure. After the model is specified it should be assessed whether the data fit on it by using statistical tests or fit indexes. Graphical causal models with latent variables have also been used [23]. Our approach examines the problem of unobserved variables on a different causality framework (i.e. the matching framework), thus it should be considered complementary rather than competitive to the existing methods. SEMs are usually based on assumptions about the model-class of the data, while matching methods require fewer assumptions. Moreover, assessing the model bias with matching-based methods is straightforward by using balance diagnostics tests while, goodness-of-fit tests used with structural equation models cannot assess whether systematic differences between units have been eliminated [15]. Thus, causal inference based on matching could be preferable in some cases. To the best of our knowledge, our method is the first that considers the problem of unobserved factors on matching methods.

VI. CONCLUSIONS

We propose *probabilistic matching*, a novel approach for causal inference when one or more key variables are unobserved or noisy. Our method is based on the assumption that probability distributions describing the values of the unobserved variables are known or can be approximated. We define a distance metric, based on Genetic Matching distance, that measures the dissimilarity between units by examining the difference on the quantiles of the stochastic variables of the study. We evaluate our method both on simulated and real datasets. We demonstrate that when the treatment variable is noisy, traditional matching methods may result in matching pairs with the same or similar treatment values. This weakens the observed effect of the treatment on the outcome variable and consequently, may result in missing true causal links. Similarly, when there is noise on the confounding variables, existing matching methods fail to sufficiently reduce

the confounding bias, which often results in false causality conclusions. We show that our approach is able to find better matches and, consequently, achieves more accurate causal conclusions.

REFERENCES

- [1] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.
- [2] X. Zhang, H. Fuehres, and P. A. Gloor. Predicting stock market indicators through Twitter "I hope it is not as bad as I fear". In *Collaborative Innovation Networks*, pages 55–62. Elsevier, 2011.
- [3] F. Tsapeli, M. Musolesi, and P. Tino. Model-free causality detection: An application to social media and financial data. *arXiv preprint arXiv:1601.03610*, 2016.
- [4] N. Lathia et al. Smartphones for large-scale behavior change interventions. *IEEE Pervasive Computing*, 12(3):66–73, 2013.
- [5] F. Tsapeli and M. Musolesi. Investigating causality in human behavior from smartphone sensor data: a quasi-experimental approach. *EPJ Data Science*, 4(1):1, 2015.
- [6] D. Bamman, B. O'Connor, and N. Smith. Censorship and deletion practices in chinese social media. *First Monday*, 17(3), 2012.
- [7] C. Fornell and D. F. Larcker. Structural equation models with unobservable variables and measurement error: Algebra and statistics. *Journal of marketing research*, pages 382–388, 1981.
- [8] P. M. Bentler. Multivariate analysis with latent variables: Causal modeling. *Annual review of psychology*, 31(1):419–456, 1980.
- [9] William R. S., T. D. Cook, and D. T. Campbell. *Experimental and quasi-experimental designs for generalized causal inference*. Wadsworth Cengage learning, 2002.
- [10] UCI Machine Learning Repository: microblogPCU dataset. <http://aiweb.techfak.uni-bielefeld.de/content/bworld-robot-control-software/>, 2015.
- [11] F. Benevenuto et al. Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, volume 6, page 12, 2010.
- [12] DB. Rubin. Causal inference using potential outcomes. *Journal of the American Statistical Association*, 2011.
- [13] A. Diamond and J. S. Sekhon. Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. *Review of Economics and Statistics*, 95(3):932–945, 2013.
- [14] B. Lu et al. Matching with doses in an observational study of a media campaign against drug abuse. *Journal of the American Statistical Association*, 96(456):1245–1253, 2001.
- [15] K. Hirano and G. W. Imbens. The propensity score with continuous treatments. *Applied Bayesian modeling and causal inference from incomplete-data perspectives*, 226164:73–84, 2004.
- [16] P. C Austin. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate Behavioral Research*, 46(3):399–424, 2011.
- [17] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.
- [18] E. A. Stuart. Matching methods for causal inference: A review and a look forward. *Statistical Science: a Review Journal of the Institute of Mathematical Statistics*, 25(1):1, 2010.
- [19] Y. Chon et al. Automatically characterizing places with opportunistic crowdsensing using smartphones. In *ACM Conference on Ubiquitous Computing*, pages 481–490. ACM, 2012.
- [20] C. Zhou et al. Discovering personally meaningful places: An interactive clustering approach. *ACM Transactions on Information Systems (TOIS)*, 25(3):12, 2007.
- [21] A. Madan et al. Social sensing: obesity, unhealthy eating and exercise in face-to-face networks. In *Wireless Health*, pages 104–110. ACM, 2010.
- [22] G. Holmes, A. Donkin, and I. H. Witten. Weka: A machine learning workbench. In *Intelligent Information Systems.*, pages 357–361. IEEE, 1994.
- [23] S. Meganck, P. Leray, and B. Manderick. Causal graphical models with latent variables: Learning and inference. In *European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 5–16. Springer, 2007.