A Robust Classifier under Missing-Not-at-Random Sample Selection Bias

Huy Mai, Wen Huang, Wei Du, Xintao Wu
Department of Electrical Engineering and Computer Science
University of Arkansas
Fayetteville, Arkansas, USA
fhuymai, wenhuang, wd005, xintaowug@uark.edu

2023 IEEE International Conference on Big Data (BigData) | 979-8-3503-2445-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/BigData59044.2023.10386877

Abstract—The shift between the training and testing distributions is commonly due to sample selection bias, a type of bias caused by non-random sampling of examples to be included in the training set. Although there are many approaches proposed to learn a classifier under sample selection bias, few address the case where a subset of labels in the training set are missing-not-at-random (MNAR) as a result of the selection process. In statistics, Greene's method formulates this type of sample selection with logistic regression as the prediction model. However, we find that simply integrating this method into a robust classification framework is not effective for this bias setting. In this paper, we propose BiasCorr, an algorithm that improves on Greene's method by modifying the original training set in order for a classifier to learn under MNAR sample selection bias. We provide theoretical guarantee for the improvement of BiasCorr over Greene's method by analyzing its bias. Experimental results on real-world datasets demonstrate that BiasCorr produces robust classifiers and can be extended to outperform state-of-the-art classifiers that have been proposed to train under sample selection bias.

Index Terms—Robust classifier, missing-not-at-random, sample selection bias

I. Introduction

Dataset shift [16] describes the phenomenon in which the training and testing sets come from different distributions. One scenario that can cause dataset shift is sample selection bias, where an example is non-uniformly chosen from the population to be part of the training process. This type of bias can ultimately cause a set of training examples to be partially observed, where any of the covariates or label of an example is missing, or even completely unobserved. As a result, the performance of classifiers that are trained using a dataset subject to sample selection bias will be degraded. Most works have proposed solutions to problems dealing with missing-at-random (MAR) bias [3], [13], [23], where the selection of samples is assumed to be independent from the label given the observed variables in the training set. However, these proposed solutions cannot properly account for the missing-not-at-random (MNAR) setting, where the selection of samples is assumed to not be independent from the label given the observed variables in the training set.

In this paper, we focus on MNAR sample selection bias on the label. One classic method proposed to account for MNAR sample selection bias on the label is the Nobel Prize winning Heckman's two-step method [9]. Heckman's method models the prediction and selection of samples as linear equations, where their relationship lies in the correlation between the noise terms. The method constructs an unbiased model by first estimating inverse Mills ratio (IMR) using the selection features and then incorporating it as a new noise term in the prediction equation. Due to its short computation time and effectiveness on bias correction, Heckman's method has been a popular choice for solving linear regression under MNAR sample selection bias. However, applying Heckman's method in the classification context is difficult. This is because the assumptions made for the use of the IMR may not be present in classifiers, causing them to perform inconsistently [18].

The joint likelihood approach [15] addresses this challenge by fitting the selection and prediction simultaneously using full information maximum likelihood (FIML) estimators. In this work, we specifically examine the task of estimating this likelihood using Greene's method [8]. As a general framework for non-linear regression models under MNAR sample selection bias, Greene's method was one of the first methods to approximate the joint likelihood in order to reduce computational complexity. We find that although the method provides a first-order optimization process to produce an optimal solution, the minimization of its loss function over the biased training set does not take samples with missing labels into account. Thus, Greene's formulation alone cannot be used as an objective function when attempting to learn a classifier robust to MNAR sample selection bias.

A. Problem Definition

Formally, let X be the feature space and Y be the binary target attribute. We first consider the training set $D_{tr} = \{t_i\}_{i=1}^{n}$ of n samples that are originally sampled from the population to be modeled yet biased under MNAR sample selection bias. Each sample t_i is defined as:

$$t_i = \begin{cases} (x_i, y_i, s_i = 1) & 1 \le i \le m \\ (x_i, s_i = 0) & m+1 \le i \le n \end{cases} \quad (1)$$

where the binary variable s_i indicates whether or not y_i is observed for a training sample. Let D_s denote the set containing the first m training samples where each sample is

fully observed and D_u be the set that contains the remaining $n - m$ training samples with unobserved labels.

We consider the following definition to formally describe the MNAR sample selection bias scenario on D_{tr}:

Definition 1 (MNAR Sample Selection): Missing-not-at-random occurs for a sample t_i if s_i is not independent of y_i given x_i, i.e. $P(s_i|x_i; y_i) = P(s_i|x_i)$. This means that s_i may depend on x_i and y_i. For Greene's method, the selection mechanism is expressed in terms of a set of selection features to model the missingness of y_i for a training sample. These selection features are observed for all training samples. Thus the following assumptions are additionally made in this work:

(i) Given a set of selection features $x_i^{(s)} \subseteq x_i$, $P(s_i|x_i; y_i)$ is approximated by computing $P(s_i|x_i^{(s)})$.
(ii) The set of selection features includes every prediction feature, i.e. $x_i^{(s)} \supseteq x_i^{(p)}$.

Problem Statement. Given a set of prediction features $x_i^{(p)} \subseteq x_i$, we seek to train a binary classifier $h(x_i^{(p)}; \theta)$ with parameters that learns to minimize a loss function over the biased training set D_{tr}.

B. Contributions

We summarize the core contributions of our work as follows. First, we propose BiasCorr to address training a robust classifier where some labels in the training set are MNAR due to sample selection bias. BiasCorr extends Greene's method to improve the performance of the robust classifier by assigning a soft selection value and pseudolabel to each unlabeled training sample. Second, using the soft selection value, we derive a condition based on the proportion of unlabeled samples in the training data to theoretically guarantee that BiasCorr is less biased than Greene's method whenever the condition is satisfied. Third, we extend BiasCorr to train a robust classifier given a training set of labeled samples that come from a biased source distribution and a testing set of unlabeled samples that come from an unbiased target distribution. Fourth, we provide empirical results on real-world datasets to confirm that BiasCorr trains classifiers that are robust against MNAR sample selection bias and can be extended to outperform state-of-the-art classifiers trained under sample selection bias.

II. Related Work

Heckman's method and its variants have been widely used for different applications to handle MNAR sample selection bias (see a comprehensive survey [21]). Despite its popularity, Heckman's method has some key limitations when applied to non-linear regression models. First, for non-linear models, this noise term of the prediction equation does not contain the IMR [18]. Second, the IMR may be incorrectly specified given the collinearity between the coefficients of the selection and prediction equations [17]. In the area of fair machine learning, [5] formulated a fair regression model under the assumption that a subset of training outcomes are MNAR. The model adopts Heckman's method as part of its framework to account for sample selection bias. Unlike these approaches, where the dependent variable is assumed to be continuous, our approach handles sample selection bias where the dependent variable is categorical. As closed-form solutions do not exist for likelihood equations maximized for logistic regression models, we depend on iterative optimization techniques in order to learn a classifier under MNAR sample selection bias.

Most research works in the area of learning under sample selection bias fall in the category of MAR bias. Approaches proposed in these works often incorporate ideas of importance weighting [2], [23] and minimax estimation [10], [13]. These approaches generally assume a labeled training set of biased samples and an unlabeled testing set of unbiased samples [2]. As we address MNAR bias, we differ from these assumptions. In our study, we assume that the testing set cannot be accessed during training and that the training set contains a mixture of labeled and unlabeled examples given that the labels are non-randomly selected.

Our problem setting is related to other machine learn-ing tasks. In recommender learning, [22] proposed the joint learning of imputation and prediction models to estimate the performance of rating prediction given MNAR ratings. While the approach in [22] also uses a separate propensity estimation model to predict label observation, it considers matrix factorization as the prediction model, which is not for binary classification on tabular data. In semi-supervised learning [20], where a training sample is treated differently based on whether the sample has a label or not, [11] employed class-aware propensity score and imputation strategies using pseudolabels to develop a semi-supervised learning model that is doubly robust against MNAR data. This approach computes the probability of label missingness for a training sample in terms of a class prior. On the other hand, our approach does not require a class prior to compute the probability of label missingness for a training sample.

III. Greene's Method Revisited

A. Sample Selection Model

For any $(x_i; y_i) \in X \times Y$, the selection equation of the ith sample is $z_i = \gamma_i x_i^{(s)} + u_i^{(s)}$, where γ_i is the set of regression coefficients for selection, $x_i^{(s)}$ is the set of features for sample selection, and $u_i^{(s)} \sim N(0; 1)$ is the noise term for the selection equation. The selection value of the ith sample s_i is defined as:

$$s_i = \begin{cases} 1 & z_i > 0 \\ 0 & z_i \leq 0 \end{cases} \tag{2}$$

The prediction equation $f(y_i|x_i^{(p)}; \theta_i)$ of the ith sample is based on logistic regression with

$$f(y_i = 1|x_i^{(p)}; \theta_i) = \frac{\exp(x_i^{(p)} \theta_i)}{1 + \exp(x_i^{(p)} \theta_i)} \tag{3}$$

where θ_i is the set of regression coefficients for prediction, $x_i^{(p)}$ is the set of features for prediction, and ϵ_i is the noise term for the prediction equation, with σ_i as the standard deviation of

the term and $_i \sim N(0;1)$ as a random variable. We express as $u_i^{(p)}$, where $u_i^{(p)} \sim N(0; ^2)$. In our work, we let $h(x_i^{(p)};) = f(y_i = 1 | x_i^{(p)};)$.

The noise terms $u_i^{(s)}$ and $u_i^{(p)}$ are assumed to be bivariate normal, i.e. $u_i^{(s)} = + v_i$, where is the correlation coefficient between $u_i^{(s)}$ and $u_i^{(p)}$ and $v_i \sim N(0;1)$ is a random variable independent to $_i$.

B. Loss Function

From the above sample selection model, the loss function

$$L = \frac{1}{n} \sum_{i=1}^{n} \log f(y_i; s_i | x_i^{(p)}; x_i^{(s)}) \quad (4)$$

over D_{tr} is then derived. The joint density function $f(y_i; s_i | x_i^{(p)}; x_i^{(s)})$ is expressed as

$$f(y_i; s_i | x_i^{(p)}; x_i^{(s)}) = \int_{-1}^{1} f(y_i; s_i | x_i^{(p)}; x_i^{(s)}; _i) \, f(_i) \, d_i \quad (5)$$

Both y_i and s_i are independent when conditioned on $_i$. Thus,

$$f(y_i; s_i = 1 | x_i^{(p)}; x^{(s)}; _i) = f(y_i | x^{(p)}; _i) P(s_i = 1 | x_i^{(s)}; _i) \quad (6)$$

For $s_i = 0$, where y_i is missing,

$$f(y_i; s_i = 0 | x_i^{(p)}; x_i^{(s)}; _i) = P(s_i = 0 | x_i^{(s)}; _i) \quad (7)$$

Because $u_i^{(s)}$ and $u_i^{(p)}$ are bivariate normal, we have

$$P(s_i | x_i^{(s)}; _i) = (2s_i - 1) \left(\frac{x_i^{(s)} + _i}{\sqrt{1 - ^2}} \right) \quad (8)$$

where $()$ is the standard normal cumulative distribution function. Since $_i \sim N(0;1)$, $f(_i)$ is $(_i)$, where $()$ is the standard normal density function. Thus, Eq. (5) is rewritten as

$$f(y_i; s_i | x_i^{(p)}; x_i^{(s)}) = \int_{-1}^{1} [(1 - s_i) + s_i f(y_i | x_i^{(p)}; _i)] \, P(s_i | x_i^{(s)}; _i) \, (_i) \, d_i \quad (9)$$

using Eq. (6), Eq. (7), and Eq. (8). Thus the negative log-likelihood function L over n training data samples is

$$L = \frac{1}{n} \sum_{i=1}^{n} \log \int_{-1}^{1} [(1 - s_i) + s_i f(y_i | x_i^{(p)}; _i)] \, P(s_i | x_i^{(s)}; _i) (_i) \, d_i \quad (10)$$

L needs to be minimized with respect to ; ; ; and . Given that the computation of Eq. (10) is intractable, the simulation approach from [19] is used to minimize an approximate form of L, denoted

$$\hat{L} = \frac{1}{n} \sum_{i=1}^{n} \hat{l}_i \quad (11)$$

where

$$\hat{l}_i = \log \frac{1}{R} \sum_{r=1}^{R} [(1 - s_i) + s_i f(y_i | x_i^{(p)}; _{ir})] \, P(s_i | x_i^{(s)}; _{ir}) \quad (12)$$

This approach involves taking R random draws $_{ir}$ from the standard normal population for each t_i. As long as R is greater than $\sqrt{n}$, then asymptotically $\hat{L} = L$. A proof of this claim is provided in [7].

C. Optimization

Iterative first-order optimization techniques such as stochastic gradient descent can be used to solve Eq. (11) and obtain an estimate $\hat{}$ for the classifier h. We note that the gradient of Eq. (12) with respect to for the ith training sample is expressed as

$$r \hat{l}_i = \frac{1}{\hat{l}_i} \frac{1}{R} \sum_{r=1}^{R} s_i \, P(s_i | x_i^{(s)}; _{ir}) \, f(y_i | x_i^{(p)}; _{ir}) \, x_i^{(p)} \, \frac{@ f(y_i | x_i^{(p)}; _{ir})}{@} \quad (13)$$

We also apply the first-order optimization techniques to compute the other estimated parameters in Eq. (11), namely , $\hat{}$, and $\hat{}$.

IV. Robust Classification under MNAR Sample Selection Bias

Despite Greene's method incorporating a sample selection model towards fitting logistic regression, the task of training a robust classifier h over D_{tr} under MNAR sample selection bias cannot be accomplished using this method. We specifically note a key issue in the optimization process. For any sample in the training set such that $s_i = 0$, the value of Eq. (13) is 0, meaning that $r \hat{l}_i$ would account for only samples such that y_i is observed. Thus, using a first-order optimization technique to solve Eq. (11) does not result in an iterative solution such that the classifier $h(x_i^{(p)};)$ is robust against MNAR sample selection bias on the label.

However, learning a robust classifier under MNAR sample selection bias can still be achieved by making improvements to Greene's method. First, we can refine the selection value of each sample in D_u to have a soft value in order to include information regarding the losses of samples in D_u when optimizing the classifier. While making the refinement, we still assume that each sample in D_s is assigned $s_i = 1$. Second, we can impute the missing labels in D_u with pseudolabels to further improve Greene's method.

A. BiasCorr

To ensure that we learn classifiers that are robust to MNAR sample selection bias, we introduce BiasCorr, a framework that addresses the challenge of training a classifier using Greene's method. In BiasCorr, we ensure that the losses of samples with missing labels are included in the optimization process. Using this framework, we train $h(x_i^{(p)};)$ to minimize L^0, which is an enhanced version of Eq. (11), over a modified training set D^0_{tr}. We make these modifications while conforming to the original MNAR conditions on the label. Figure 1 gives an illustration of the process to obtain D^0_{tr}.

Using the same assumptions as Greene's method on the training set D_{tr}, BiasCorr assigns both an estimated soft selection value s and a pseudolabel $\tilde{y}$ to each sample in D_u, resulting in h training to minimize the equation

$$L^0 = \frac{1}{n} \sum_{i=1}^{n} \log \frac{1}{R} \sum_{r=1}^{R} [(1 - s_i^0) + s_i^0 f(y_i^0 | x_i^{(p)}; _{ir})] \, P(s_i | x_i^{(s)}; _{ir}) \quad (14)$$

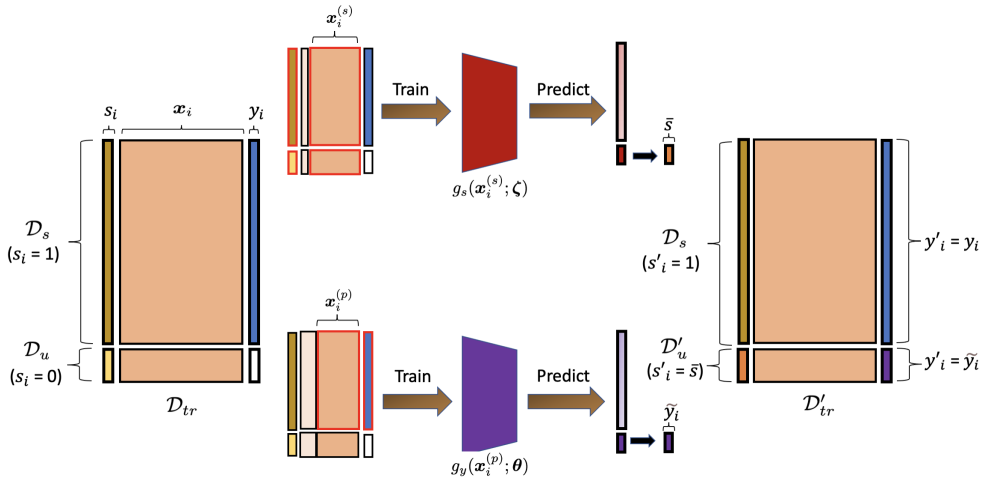


Fig. 1. Process of producing D_{tr}^{C} using BiasCorr. The boxes outlined in red indicate the parts of D_{tr} used to train g_s and g_y.

Algorithm 1 BiasCorr(g_s; g_y)

Input: Original training set $D_{tr} = \{(x_i, y_i, s_i = 1)\}_{i=1}^{m} \cup \{(x_i, s_i = 0)\}_{i=m+1}^{n}$, g_s, g_y
Output: Estimated classifier parameters $\hat{\theta}$

1: $D_s \leftarrow \{(x_i, y_i, s_i = 1)\}_{i=1}^{m}$
2: $D_u \leftarrow \{(x_i, s_i = 0)\}_{i=m+1}^{n}$
3: $D_u^C \leftarrow$;
4: Train classifier $g_s(x_i^{(s)}; \zeta)$ on D_{tr} to predict s_i
5: Train classifier $g_y(x_i^{(p)}; \theta)$ on D_s to predict y_i
6: **for** $t_i \in D_u$ **do**
7: $p_i^{(s)} \leftarrow g_s(x_i^{(s)}; \zeta)$
8: $\hat{y}_i \leftarrow \mathbb{1}[g_y(x_i^{(p)}; \theta) > 0.5]$
9: **end for**
10: $\bar{s} \leftarrow \frac{1}{n-m} \sum_{i=m+1}^{n} p_i^{(s)}$
11: **for** $i \in \{m+1, \ldots, n\}$ **do**
12: $D_u^0 \leftarrow D_u^0 \cup \{(x_i, y_i^0 = \hat{y}_i, s_i^0 = \bar{s})\}$
13: **end for**
14: $D_{tr}^0 \leftarrow D_s \cup D_u^0$
15: Train $h(x_i^{(p)}; \theta)$ to minimize $L^{0\wedge}$ using D_{rt}^0 and obtain $\hat{\theta}$
16: **return** $\hat{\theta}$

over $D_{tr}^0 = D_s \cup D_u^0$, where

$$s_i^0 = \begin{cases} 1 & t_i \in D_s \\ \bar{s} & t_i \in D_u \end{cases} \tag{15}$$

and

$$y_i^C = \begin{cases} y_i & t_i \in D_s \\ \hat{y}_i & t_i \in D_u \end{cases} \tag{16}$$

To estimate the soft selection value $\bar{s}$, we start by computing the probability $p_i^{(s)}$ of predicting $s_i = 1$ for all samples in D_u. This is based on our observation that the value of $P(s_i = 1 | x_i^{(s)}; \zeta)$ is not always equal to 0 for a tuple in D_u, where the ground truth selection value is $s_i = 0$. In our framework, we train a separate binary classifier $g_s(x_i^{(s)}; \zeta)$ on D_{tr} to predict s_i and obtain $p_i^{(s)}$ based on predictions using D_u. We then get a fixed soft selection value by taking the average value of $p_i^{(s)}$ for all samples in D_u.

The pseudocode for BiasCorr is provided in Algorithm 1. In line 4, we first train g_s on D_{tr} to predict the original ground-truth selection value s_i. In line 5, we train another binary classifier $g_y(x_i^{(p)}; \theta)$ with parameters θ on D_s to predict the ground-truth label y_i. To add samples to D_u^0, in line 7, we evaluate g_s and obtain the probability $p_i^{(s)}$ for each sample in D_u. In line 8, we use the prediction from the evaluation of g_y on each sample in D_u to obtain a pseudolabel $\hat{y}_i$. In line 10, we compute the average $\bar{s}$ of $p_i^{(s)}$ of each sample in D_u. In line 12, we add each tuple $(x_i, \hat{y}_i, \bar{s})$ to D_u^0, where each x_i is taken from D_u. In line 15, using D_{tr}^0 we obtain $\hat{\theta}$ after minimizing Eq. (14) such that $h(x_i^{(p)}; \hat{\theta})$ is robust against non-random sample selection bias on the label.

The computational complexity of Algorithm 1 trivially depends on the complexity of training g_s, g_y, and h to convergence. Similar to the training of h using Eq. (11), the complexity of training h by minimizing Eq. (14) is $O(Tn)$, where T is the number of iterations for training h.

We further note that the types of models used to train g_s and g_y are listed as inputs to Algorithm 1. In our work, we experiment with training g_s using the probit and logistic regression models. Compared to logistic regression models, which are based on the sigmoid function, probit models use the normal cumulative distribution function to model binary classification. For g_y, we consider logistic regression and multi-layer perceptron.

B. Bias Analysis Regarding Loss Function

In this section, we analyze the bias of the loss function estimator for both Greene's method and BiasCorr. We compare the two biases and show that our BiasCorr algorithm further reduces the bias for classification performance estimation given that the ratio of the unlabeled training set is larger than a

threshold. We first define the optimized negative log-likelihood loss function where the training data D_{tr} is fully observed:

$$L = \frac{1}{|D_{tr}|} \sum_{i \in D_{tr}} \log P(y_i|x_i) = \frac{1}{n} \sum_{i=1}^{n} \log f(y_i|x_i^{(p)}) \quad (17)$$

where $f(y_i|x_i^{(p)})$ takes the form of logistic regression. The bias of an arbitrary loss function estimator $\hat{L}$ is defined as:

$$Bias(\hat{L}) = L - E_{D_{tr}}[\hat{L}] \quad (18)$$

Given $|D_{tr}| = n$ and $|D_s| = m$, we further define the missingness ratio of the unlabeled training samples as $= |D_u|/|D_{tr}| = 1 - \frac{m}{n}$. We also denote $p(s_i)$ as the ground truth selection probability for each tuple t_i based on its selection features $x_i^{(s)}$ and the expectation of the estimated selection model $P(s_i|x_i^{(s)}; \hat{\gamma})$ and prediction model $f(y_i|x_i^{(p)}; \hat{\theta})$ over R random draws on the error terms as $\hat{p}(s_i)$ and $\hat{f}(y_i|x_i^{(p)})$ respectively. We next formally derive the bias of the loss function estimators from Greene's method and BiasCorr in the following two lemmas:

Lemma 1 (Bias of Greene's method estimator): Given the estimated selection model $\hat{p}(s_i)$ and outcome model $\hat{f}(y_i|x_i^{(p)})$, the bias of the loss function estimator for Greene's method shown in Eq. (11) is:

$$Bias(\hat{L}) = \frac{1}{n} \sum_{i=1}^{n} \log \frac{f(y_i|x_i^{(p)})}{\hat{p}(s_i) + p(s_i)\hat{p}(s_i)(\hat{f}(y_i|x_i^{(p)}) - 1)} \quad (19)$$

Lemma 2 (Bias of BiasCorr estimator): Given the definitions of s_i^0, s and y_i^0 in Section IV-A, the bias of the BiasCorr loss function estimator shown in Eq. (14) is:

$$Bias(\hat{L}^0) = \frac{1}{n} \sum_{i=1}^{n} \log \frac{f(y_i|x_i^{(p)})}{\hat{p}(s_i^0) + (p(s_i) + \epsilon)\hat{p}(s_i^0)(\hat{f}(y_i^0|x_i^{(p)}) - 1)} \quad (20)$$

Note that both $Bias(\hat{L})$ and $Bias(\hat{L}^0)$ are non-zero even if the estimated selection and outcome models are accurate, that is, $\hat{p}(s_i) = p(s_i)$ and $\hat{f}(y_i|x_i^{(p)}) = f(y_i|x_i^{(p)})$. According to the design of the log-likelihood loss function in Eq. (10), Greene's method estimates the likelihood function $f(y_i, s_i|x_i^{(p)}; x_i^{(s)})$ by computing $\hat{f}(y_i|x_i^{(p)})\hat{p}(s_i)$ for samples in D_s and $\hat{p}(s_i)$ for samples in D_u. Due to the fundamental difference between selection and prediction models, it is very challenging to derive an unbiased estimator for the loss function based on Greene's method. However, by applying the modification from BiasCorr, we are able to further reduce the bias for the loss function estimator on classification tasks based on an assumption on the ratio . We list our main theorem that compares the biases of the two methods as follows:

Theorem 1: Given a training dataset with labeled and unlabeled tuples $D_{tr} = D_s \cup D_u$, suppose $\hat{f}(y_i|x_i^{(p)})$ takes the form of logistic regression, and there is no bias caused by the estimated selection model for both Greene's method and BiasCorr. If the ratio of the unlabeled training data is larger than $1/(2 - \epsilon)$, we have

$$Bias(\hat{L}^0) < Bias(\hat{L})$$

TABLE I

DATASET ATTRIBUTES AND STATISTICS. PREDICTION FEATURES ARE IN ITALIC FONT WHILE SELECTION FEATURES ARE IN EITHER ITALIC OR REGULAR FONT. TARGET ATTRIBUTE IS BOLDED FOR EACH DATASET.

Dataset	Attrbutes	$\|D_{tr}\|$	Adult
	Age, Target, Education-Num,, Cap Gain, Hrs per week, Country_Canada, Rel_Not-in-fam, Occ_Adm-clerical, Occ_Sales, Rel_Husband, Occ_Craft-repair, Rel_Unmarried, Rel_Other-rel, Occ_Armed-Forces, Rel_Own-child, Occ_Other-service, Occ_Protect-serv, Cap Loss, Occ_Prof-spec, Occ_Tech, Rel_Wife, Occ_Exec-manager, Occ_Farm-fish, Marital Status, Occ_Mach-op-inspct, Occ_Priv-serv, Occ_Handlers-cleaners, Occ_Transp, Workclass	45,222	0.7476
German	status checking, duration, credit history, credit amt, savings acct, telephone, liable, other plans, last employment, age, status and sex, foreign worker, last residence, property, existing credits, good customer	1,000	0.2314
Drug	Age, Gender, Education, Country, Cscore, Impulsive, Ethnicity, Nscore, Escore, Oscore, Ascore, SS, Benzos	1,885	0.6520

To obtain the result in Theorem 1 we consider the difference between the two biases and analyze the terms after subtracting $Bias(\hat{L})$ by $Bias(\hat{L}^0)$. We first decompose the difference and derive the inequality as follows:

$$Bias(\hat{L}) - Bias(\hat{L}^0) \propto \underbrace{\frac{1}{n} \sum_{i=1}^{n} -2p(s_i) + \epsilon}_{\text{term 1}} + \underbrace{\frac{1}{n} \sum_{i=1}^{n} \hat{f}(y_i|x_i^{(p)})(2p(s_i) + \epsilon)}_{\text{term 2}} \quad (21)$$

According to Eq. (21), we find that if terms 1 and 2 are positive, the BiasCorr estimator is guaranteed to achieve lower bias than the estimator for Greene's method. Both $\hat{f}(y_i|x_i^{(p)})$ and $p(s_i)$ lie in $(0; 1)$ for each tuple t_i, so term 2 is positive after summation and averaging over all t_i. Our theoretical analysis shows that to guarantee the positivity of term 1, the proportion of the unlabeled training data needs to be larger than $1/(2 - \epsilon)$. Notice that the condition $1/(2 - \epsilon)$ does not necessarily imply $Bias(L^0)$ is larger than $Bias(L)$. We still need to compare the magnitude of term 1 and term 2, and the value of term 2 heavily depends on the estimated selection and outcome models.

Proof details of Lemma 1, Lemma 2 and Theorem 1 can be found in the Appendix in [14].

C. Extending BiasCorr to BiasCorr

Most algorithms that have been proposed to learn classification under sample selection bias are trained under the assumption that the training set $D_s = \{(x_i; y_i)\}_{i=1}^{m}$ contains labeled samples that come from a biased source distribution. Additionally, they assume that there exists a set $D_N = \{x_i\}_{i=1}^{N}$ of testing samples drawn from an unbiased target distribution. We propose an extension of BiasCorr, BiasCorr, for this setting. We do so by augmenting the original set of labeled training samples using the set of unlabeled samples from the target distribution. Specifically, given D_s and D_N, we construct an augmented training set $D_{aug} = D_s \cup D_u$ of n

TABLE II
PERFORMANCE OF BASELINES COMPARED TO BIASCORR. HIGHEST TEST ACCURACIES AMONG SSBIAS, GREENE'S METHOD, IPS, DOUBLY ROBUST, AND THE FOUR BIASCORR SETTINGS ARE IN BOLD.

Methods	Adult		German		Drug	
	Train Acc. (%)	Test Acc. (%)	Train Acc. (%)	Test Acc. (%)	Train Acc. (%)	Test Acc. (%)
NoBias	86.57 0.00	86.57 0.00	73.29 0.00	72.67 0.00	69.83 0.00	69.08 0.00
SSBias	77.56 0.00	62.44 0.00	75.28 0.00	69.33 0.00	77.78 0.00	66.78 0.00
Greene's method [8]	62.94 0.07	62.89 0.09	72.77 0.47	69.67 0.30	68.89 0.27	66.71 0.33
IPS [12]	77.84 0.21	71.86 0.23	75.62 0.27	70.06 0.32	77.84 0.10	67.40 0.26
Doubly Robust [1]	93.69 0.05	85.21 0.06	81.88 0.14	70.46 0.27	89.14 0.48	67.62 0.15
BiasCorr (probit, LR)	86.84 0.02	70.05 0.04	79.97 0.14	71.60 0.13	87.93 0.07	69.22 0.02
BiasCorr (LR, LR)	87.36 0.04	69.84 0.04	80.11 0.25	71.07 0.13	88.89 0.09	67.81 0.17
BiasCorr (probit, MLP)	94.08 0.02	85.68 0.01	79.69 0.40	71.27 0.25	86.19 0.06	67.39 0.09
BiasCorr (LR, MLP)	93.45 0.01	85.79 0.02	79.69 0.50	71.00 0.21	85.97 0.13	67.77 0.14

samples, where D_u contains samples that are uniformly drawn from D_N and $n > m$.

To obtain D_{aug}, we first randomly draw n samples uniformly from D_N, where $n > m$. Let D_n denote this set of n samples[1]. To construct D_u, we compare the empirical frequencies of D_s and D_n, which follows a similar procedure as [3]. For a distinct sample t, let D^t be a subset of D_s that contains all instances of t and $a_t = |D^t|$. We similarly define D^t and b_t for D_n. Until D_u contains $n - m$ samples, we add $b_t \frac{n}{s} - a_t$ random samples from D^t_n to D_u for each t such that $b_t > a_t$.

We note that choosing n as the size of D_{aug} is significant in determining the performance of estimating the selection probability and the efficiency of BiasCorr. First, the following lemma from [3] shows the error of using $\frac{a_t}{b_t}$ as an estimate of the selection probability $P(s_i = 1|t)$.

Lemma 3: [3] Let > 0. Let a^0 be the number of distinct samples in D_s and $p_0 = \min_{t \in D_{aug}} P(t) = 0$. Then, with probability at least 1 , the following inequality holds for all distinct $t \in D$:

$$P(s_i = 1|t) \quad \frac{a_t}{b_t} \quad \sqrt{\frac{\log 2a^0 + \log \frac{1}{}}{p_0 n}} \tag{22}$$

Here we see that for a given number of distinct samples in D_s, the error of estimating $P(s_i = 1|t)$ depends on the value of $p_0 n$, which equals the number of occurrences of the least frequent sample in D_{aug}. This value is dependent on the set D_u, which may include samples t that are not in D_s. Second, the computational complexity of generating D_{aug} is bounded by n, where in the worst case D_n has n distinct samples and the last $n - m$ samples in D_n are added to D_u.

V. EXPERIMENTS

A. Experiments on BiasCorr

We evaluate the performance of our proposed algorithms on the Adult, German, and Drug datasets [6]. The attributes and statistics used for each dataset are listed in Table I. We choose 70% of samples in each dataset to generate the original training set D_{tr}. We work with two different bias scenario types in our experiments: one where the condition of listed in Theorem 1 is satisfied and another where the condition is not met. To create the sample selection bias on D_{tr} for the Adult dataset, we select a training sample to have an observed label if the years of education is more than 12. For the German dataset, we select a training sample to be fully observed if the person has been employed for more than 1 year. For the Drug dataset, we create the sample selection bias scenario for D_{tr} by selecting individuals whose Oscore is at most 43.

Baselines and Implementation. We compare BiasCorr to the following baselines: (a) logistic regression without sample selection bias (NoBias), which is trained using D_{tr} where all samples in D_{tr} are fully observed, (b) logistic regression with sample selection bias (SSBias), which is trained using D_s, (c) logistic regression with sample selection bias correction based on Greene's method, which is trained using the set $D_s \cup D_u$ where all samples in D_u have non-randomly missing labels, (d) inverse propensity scoring (IPS) [12], where the optimized loss function is reweighted with the reciprocal of the selection probability, and (e) Doubly Robust [1], where all labels in D_u are imputed and the loss is reweighted based on IPS. Compared to BiasCorr, IPS and Doubly Robust do not consider the correlation between the prediction and selection equations. Our models and all baselines are implemented using Pytorch. The prediction and selection coefficients and are initialized to zero while and are initialized to 0:01. The number of random draws R is set to 200. Our source code can be downloaded using the link https://tinyurl.com/4kvux87n.

Results. Table II shows the training/testing accuracy of each model. We report average accuracies and their standard deviations over 5 runs. We first see that while the change in training accuracies is different for each dataset when comparing NoBias and SSBias, NoBias outperforms SSBias by 24.13%, 3.34%, and 2.30% when considering the testing accuracy for the Adult, German, and Drug datasets, respectively. This shows that the utility of the logistic regression model is reduced when trained on D_s. We also see that Greene's method does not outperform SSBias by much when evaluated on the testing set. For instance, when looking at the results for the Adult dataset in Table II, the testing accuracy of Greene's method is 0.55% higher than SSBias while the testing accuracy of NoBias is 24.13% higher. This demonstrates that a classifier is not robust to MNAR sample selection bias when learning to optimize Eq. (11).

[1]We note that in some cases, $N < m$. To obtain D_n, we draw n samples from D_N with replacement.

TABLE III
EMPIRICAL MISSINGNESS RATIO COMPARISON.

Dataset		1=(2 $\hat{}$ (probit for g_s)	1=(2 $\hat{}$ (LR for g_s)
Adult	0.7476	0.5868	0.5738
German	0.2314	0.6345	0.6233
Drug	0.6520	0.7159	0.5976

TABLE IV
EXECUTION TIMES (IN SECONDS).

Method	Adult	German	Drug
Greene's method	93.53	2.06	3.14
BiasCorr (probit, LR)	94.59	1.84	2.85
BiasCorr (LR, LR)	99.62	1.87	2.28
BiasCorr (probit, MLP)	112.17	1.86	2.69
BiasCorr (LR, MLP)	112.90	1.87	2.26

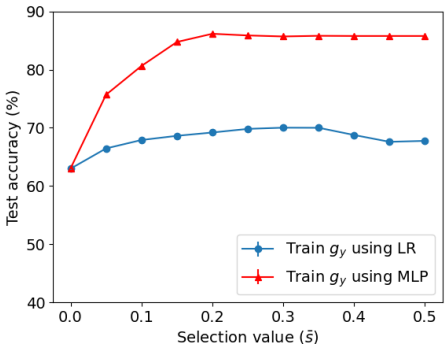
More importantly, we observe that BiasCorr, under all 4 pairs of settings for g_s and g_y, outperforms SSBias and Greene's method. Using the German dataset as an example, BiasCorr(LR, MLP) has the lowest test accuracy out of the four BiasCorr settings after training on the dataset. Despite this, BiasCorr(LR, MLP) outperforms SSBias by 1.67% on the testing set. This difference is higher than the 0.34% margin when comparing Greene's method to SSBias.

Furthermore, we see that BiasCorr outperforms IPS and Doubly Robust for most pairs of settings for g_s and g_y. For instance, on the Drug dataset, the average testing accuracies of BiasCorr(probit, LR), BiasCorr(LR, LR), and BiasCorr(LR, MLP) are higher than that of IPS and Doubly Robust.

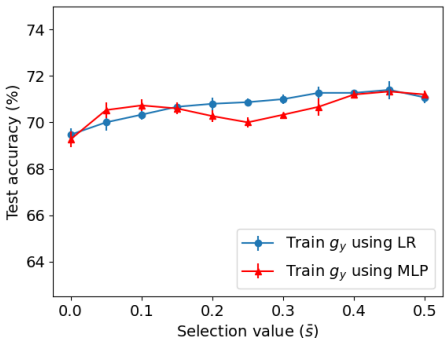
We also examine the values of and 1=(2 $\hat{}$ in Table III based on this experiment. Using g_s on probit as an example, we see that the value of 1=(2 $\hat{}$ is 0.5868 for the Adult dataset. We also observe that, as shown in Table II, BiasCorr(probit, LR) and BiasCorr(probit, MLP) outperform Greene's method by 7.16% and 22.79%, respectively. As > 1=(2 $\hat{}$ for the Adult dataset, the result validates our theoretical comparison of BiasCorr and Greene's method. For the other two datasets, we see that the value of 1=(2 $\hat{}$ is not less than . However, BiasCorr still outperforms Greene's method across all 4 combinations of settings for g_s and g_y. This shows that our BiasCorr algorithm, which improves Greene's method by incorporating pseudolabel generation and a soft selection assignment on samples in D_u, produces a more robust classifier against MNAR sample selection bias.

Execution Time. We also report the execution times of training h using Greene's method and BiasCorr in Table IV, where the experiments were conducted on the Dell XPS 8950 9020 with an Nvidia GeForce RTX 3080 Ti. We see that BiasCorr trains slower than Greene's method for the Adult dataset while BiasCorr has a slightly faster execution time than Greene's method for the German and Drug datasets.

Sensitivity Analysis. We further evaluate the performance of BiasCorr by considering different assignments for the soft selection value son samples in D^0_u, up to s= 0:5. In this particular study, we run Algorithm 1 except we ignore the training of g_s using probit or logistic regression. Figure 2 shows the results of this experiment over the Adult and German datasets.



(a) Adult. s= 0:2957 (probit) and s= 0:2571 (LR).



(b) German. s= 0:4240 (probit) and s= 0:3957 (LR).

Fig. 2. Evaluation of BiasCorr using different assignments of son samples in D_u. Estimates of sobtained after training g_s are also given.

TABLE V
PERFORMANCE OF BASELINES ACROSS DIFFERENT VALUES OF COMPARED TO BIASCORR USING THE DRUG DATASET.

Method	1=(2 $\hat{}$	Test Acc. (%)	F1 Score (%)
= 0:5			
SSBias	-	65.72 0.00	56.70 0.00 Greene's
method	-	65.90 0.27	55.67 0.24 BiasCorr
(probit, LR)	0.6365	68.23 0.38	62.77 0.60 BiasCorr
(LR, LR)	0.6224	67.95 0.52	61.97 0.82
= 0:6			
SSBias	-	68.55 0.00	62.61 0.00 Greene's
method	-	67.63 0.32	60.38 0.51 BiasCorr
(probit, LR)	0.6279	69.40 0.07	65.16 0.06 BiasCorr
(LR, LR)	0.6167	69.43 0.00	65.19 0.00
= 0:7			
SSBias	-	68.37 0.00	59.78 0.00 Greene's
method	-	67.10 0.44	56.92 0.69 BiasCorr
(probit, LR)	0.6258	69.22 0.13	64.92 0.13 BiasCorr
(LR, LR)	0.6072	69.43 0.16	65.05 0.13

We see that when training g_y under both logistic regression and an MLP, the performance of BiasCorr peaks within the range of the estimates we obtain by computing the average of predictions given by g_s on samples in D_u.

We also evaluate how modifying on the training set affects the performance of BiasCorr on the testing set. We look at the values of 0.5, 0.6, and 0.7 for . We train our method using the Drug dataset for this experiment. Using testing accuracy and F1 score as evaluation metrics and logistic regression to train g_y, we report the results of this sensitivity analysis in Table V. We first see that > 1=(2 $\hat{}$ when = 0:7. For =

TABLE VI
PERFORMANCE OF BASELINES COMPARED TO BIASCORR.

Methods	Adult		German		Drug	
	Train Acc. (%)	Test Acc. (%)	Train Acc. (%)	Test Acc. (%)	Train Acc. (%)	Test Acc. (%)
RFLearn[1] [4]	78.04 0.00	69.68 0.00	76.02 0.00	69.67 0.00	75.82 0.00	65.02 0.00
RBA [13]	77.69 0.00	69.59 0.00	75.84 0.00	67.33 0.00	75.82 0.00	65.55 0.00
BiasCorr (probit, LR)	87.10 0.02	69.84 0.07	80.57 0.09	70.47 0.34	87.70 0.13	68.52 0.14
BiasCorr (LR, LR)	87.37 0.03	69.75 0.02	80.57 0.16	70.67 0.30	87.98 0.11	68.34 0.21
BiasCorr (probit, MLP)	94.00 0.35	85.75 0.01	79.66 0.21	70.07 0.13	87.20 0.10	68.23 0.13
BiasCorr (LR, MLP)	93.78 0.36	85.62 0.02	79.40 0.17	69.87 0.16	87.40 0.15	67.99 0.26

0:7, BiasCorr(probit, LR) and BiasCorr(LR, LR) outperform SSBias and Greene's method based on testing accuracy and F1 score. For the other two values of , where the condition is not satisfied, BiasCorr(probit, LR) and BiasCorr(LR, LR) still outperform SSBias and Greene's method.

B. Experiments on BiasCorr

For the biased training set of labeled samples, we use the same set D_s that was used in the experiments on BiasCorr and leave the rest of the samples unlabeled as part of the set D_N. We fix the number of samples n drawn from D_N to be the number of samples obtained after splitting each dataset.

Baselines. We compare BiasCorr to the following baselines that were proposed to learn classification under MAR sample selection bias where samples from the unbiased target distribution are unlabeled: (a) a robust non-fair version of RFLearn[1] [4], which considers the empirical frequencies of each record in D_s and the unlabeled testing set to estimate the true probability of selection, and (b) the Robust Bias Aware (RBA) classifier [13], which uses minimax estimation to learn against a worst-case conditional label distribution.

Results. As shown in Table VI, BiasCorr, under all combinations of settings for g_s and g_y, outperforms the baselines when trained on the three datasets. For instance, the testing accuracy for BiasCorr (probit, LR) is 3.14% higher than RBA for the German dataset. These results suggest that BiasCorr can outperform other classifiers trained under sample selection bias regardless of the type of model chosen for g_s and g_y or the proportion of unbiased, unlabeled samples in D_{aug}.

VI. CONCLUSION

In this paper, we have proposed a framework, BiasCorr, to learn a classifier that is robust against MNAR sample selection bias on the label. As a significant improvement to a formulation previously proposed to model MNAR sample selection bias, BiasCorr trains a robust classifier after learning separate classifiers to predict pseudolabels and estimate a soft selection value assignment for these samples. Theoretical analysis of the bias of BiasCorr provides a guarantee for this improvement based on the level of missingness in the training set. Experimental results on real-world datasets demonstrate not only the robustness of classifiers under this framework, but also their better performance than baselines. In the future, we plan to extend this framework to learn more complex nonlinear regression models such as kernel ridge regression.

ACKNOWLEDGEMENT

This work was supported in part by NSF 1946391 and 2137335.

REFERENCES

[1] Heejung Bang and James M Robins. Doubly robust estimation in missing data and causal inference models. Biometrics, 61(4):962–973, 2005.

[2] Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative learning for differing training and test distributions. In ICML, 2007.

[3] Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In ALT, 2008.

[4] Wei Du and Xintao Wu. Fair and robust classification under sample selection bias. In CIKM, 2021.

[5] Wei Du, Xintao Wu, and Hanghang Tong. Fair regression under sample selection bias. In Big Data, 2022.

[6] Dheeru Dua and Casey Graff. UCI machine learning repository. http://archive.ics.uci.edu/ml, 2017.

[7] Lung fei Lee. Asymptotic bias in simulated maximum likelihood estimation of discrete choice models. Econometric Theory, 1995.

[8] William Greene. A General Approach to Incorporating Selectivity in a Model. Working Papers 06-10, New York University, 2006.

[9] James J. Heckman. Sample selection bias as a specification error. Econometrica: Journal of the econometric society, pages 153–161, 1979.

[10] Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama. Does distributionally robust supervised learning give robust classifiers? In ICML, 2018.

[11] Xinting Hu, Yulei Niu, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. On non-random missing labels in semi-supervised learning. arXiv preprint arXiv:2206.14923, 2022.

[12] Roderick JA Little and Donald B Rubin. Statistical analysis with missing data, volume 793. John Wiley & Sons, 2019.

[13] Anqi Liu and Brian Ziebart. Robust classification under sample selection bias. In NeurIPS, 2014.

[14] Huy Mai, Wen Huang, Wei Du, and Xintao Wu. A robust classifier under missing-not-at-random sample selection bias. arXiv preprint arXiv:2305.15641, 2023.

[15] Alfonso Miranda and Sophia Rabe-Hesketh. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata J., 6(3):285–308, 2006.

[16] Jose G. Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. Pattern recognition, 45(1):521–530, 2012.

[17] Patrick Puhani. The heckman correction for sample selection and its critique. Journal of economic surveys, 14(1):53–68, 2000.

[18] Joseph V. Terza. Parametric nonlinear regression with endogenous switching. Econometric Reviews, 2009.

[19] Kenneth E. Train. Discrete choice methods with simulation. Cambridge University Press, 2009.

[20] Jesper E. Van Engelen and Holger H. Hoos. A survey on semi-supervised learning. Machine Learning, 109(2):373–440, 2020.

[21] Francis Vella. Estimating models with sample selection bias: a survey. Journal of Human Resources, pages 127–169, 1998.

[22] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. Doubly robust joint learning for recommendation on data missing not at random. In ICML, 2019.

[23] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In ICML, 2004.