Beyond the Selected Completely At Random Assumption for Learning from Positive and Unlabeled Data

Jessa Bekker[0000−0003−1928−7374] (✉), Pieter Robberechts[0000−0002−3734−0047], and Jesse Davis[0000−0002−3748−9263]

Dept of Computer Science, KU Leuven, Belgium
`first.lastname@cs.kuleuven.be`

Abstract. Most positive and unlabeled data is subject to selection biases. The labeled examples can, for example, be selected from the positive set because they are easier to obtain or more obviously positive. This paper investigates how learning can be enabled in this setting. We propose and theoretically analyze an empirical-risk-based method for incorporating the labeling mechanism. Additionally, we investigate under which assumptions learning is possible when the labeling mechanism is not fully understood and propose a practical method to enable this. Our empirical analysis supports the theoretical results and shows that taking into account the possibility of a selection bias, even when the labeling mechanism is unknown, improves the trained classifiers.

Keywords: PU Learning · Unlabeled Data · Classification.

1 Introduction

Positive and unlabeled learning focuses on the setting where the training data contains some labeled positive examples and unlabeled examples, which could belong to either the positive or negative class. This contrasts to supervised learning, where a learner has a fully labeled training set and to semi-supervised learning, where a learner (usually) observes some labeled examples from each class. Positive and unlabeled (PU) data naturally arises in many applications. Electronic medical records (EMR) list diseases that a patient has been diagnosed with, however, many diseases are undiagnosed. Therefore, the absence of a diagnosis does not imply that a patient does not have the disease. Similarly, automatically constructed knowledge bases (KBs) are incomplete, and hence any absent tuple may be either true (i.e., belong in the knowledge base) or false [37].

Elkan and Noto [12] formalized one of the most commonly made assumptions in PU learning: the observed positive examples were selected completely at random from the set of all positive examples. This assumption means that the probability of observing the label of a positive example is constant (i.e., the same for all positive examples), which facilitates and simplifies both theoretical analysis and algorithmic design. This setting has been extensively explored in the literature [10, 22, 24, 9, 26, 7, 16, 5, 4, 21, 27, 15].

Unfortunately, the "selected completely at random" assumption is often violated in real-world data sets. For example, a patient's EMR will only contain a diagnosis if she visits a doctor, which will be influenced by factors such as the severity of the symptoms and her socio-economic status. The problem of biases in the observed labels has been recognized in the recommender systems and retrieval literature [32, 25, 19]. However, these works differ from PU learning in that the labels for some examples from each "class" are observed. Still, within the context of PU learning, there has been little (or no) work that focuses on coping with biases in the observed positive labels.

The contribution of this paper is to take a step towards filling that gap by proposing and analyzing a new, less restrictive assumption for PU learning: the *Selected At Random (SAR)* assumption. Instead of assuming a constant probability for all positive examples to be labeled, it assumes that the probability is a function of a subset of an example's attributes. To help analyze this new setting, we leverage the idea of a propensity score, which is a term originating from the causal inference literature [17]. Intuitively, the propensity score can be thought of as an instance-specific probability that an example was selected to be labeled. We show theoretically how using propensity scores in a SAR setting provides benefits. Then, we discuss a practical approach for learning the propensity scores from the data and using them to learn a model. Empirically, we show that for SAR PU data, our approach results in improved performance over making the standard selected completely at random assumption.

2   Preliminaries

PU learning entails learning a binary classifier only given access to positive examples and unlabeled data. This paper considers the single-training set scenario, where the data can be viewed as a set of triples (x, y, s) with x a vector of the attributes, y the class and s a binary variable representing whether the tuple was selected to be labeled. While y is always missing, information about it can be derived from the value of s. If $s = 1$, then the example belongs to the positive class as $\Pr(y = 1|s = 1) = 1$. If $s = 0$, the instance can belong to either class.

In PU learning, it is common to make the *Selected Completely at Random (SCAR)* assumption, which assumes that the observed positive examples are a random subset of the complete set of positive examples. Selecting a positive example is therefore independent of the example's attributes $\Pr(s = 1|y = 1, x) = \Pr(s = 1|y = 1)$. The probability for selecting a positive example to be labeled is known as the label frequency $c = \Pr(s = 1|y = 1)$. A neat advantage of the SCAR assumption is that, using the label frequency, a model that predicts the probability of an example being labeled can be transformed to the classifier: $\Pr(y = 1|x) = \Pr(s = 1|x)/c$.

Knowing the label frequency is equivalent to knowing the class prior $\alpha = \Pr(y = 1)$ as one can be derived from the other: $c = \Pr(s = 1)/\alpha$. Under the SCAR assumption, PU learning can therefore be reduced to estimating the class prior or label frequency and training a model to predict the observed labels.

Estimating the label frequency is an ill-defined problem because it is not identifiable: the absence of a label can be explained by either a small prior probability for the positive class or a low label frequency [33]. For the class prior to be identifiable, additional assumptions are necessary. Different assumptions have been proposed, but they are all based on attributing as many missing classes as possible to a lower label frequency as opposed to a lower positive class probability. The following assumptions are listed from strongest to strictly weaker. The strongest assumption is that the classes are non-overlapping, which makes the class prior and the labeled distribution match the unlabeled one as closely as possible [12, 29]. Others make the assumption that there exists a positive subdomain of the instance space, but the classes can overlap elsewhere [33, 28, 1]. Ramaswamy et al. [30] assumes that a function exists which only selects positive instances. Finally, the irreducibility assumption states that the negative distribution cannot be a mixture that contains the positive distribution [3, 18].

3   Labeling Mechanisms for PU Learning

The labeling mechanism determines which positive examples are labeled. To date, PU learning has largely focused on the SCAR setting. However, labels are not missing completely at random in most real-world problems. For example, facts in automatically constructed KBs are biased in several ways. One, they are learned from Web data, and only certain types of information appear on the Web (e.g., there is more text about high-level professional sports teams than low-level ones). Two, the algorithms that extract information from the Web employ heuristics to ensure that only information that is likely to be accurate (e.g., redundancy) is included in the KB. Similarly, biases arise when people decide to like items online, bookmark web pages, or subscribe to mail lists. Therefore, we believe it is important to consider and study other labeling mechanisms.

When Elkan and Noto [12] first formalized the SCAR assumption, they noted the similarity of the PU setting to the general problem of learning in the presence of missing data. Specifically, they noted that the SCAR assumption is somewhat analogous with the missing data mechanism called *Missing Completely At Random (MCAR)* [31]. Apart from MCAR, the two other classes of missing data mechanisms are *Missing At Random (MAR)* and *Missing Not At Random (MNAR)*. To complete this analogy, we propose the following corresponding classes of PU labeling mechanisms:

SCAR *Selected Completely At Random*: The labeling mechanism does not depend on the attributes of the example, nor on the probability of the example being positive: each positive example has the same probability to be labeled.

SAR *Selected At Random*: The labeling mechanism depends on the values of the attributes of the example, but given the attribute values it does not depend on the probability of the example being positive.

SNAR *Selected Not At Random*: All other cases: The labeling mechanism
   depends on the real probability of this example being positive, even given
   the attribute values.

There is one very important difference between PU labeling mechanisms
and missingness mechanisms in that the labeling always depends on the class
value: only positive examples can be selected to be labeled. According to the
missingness taxonomy, all PU labeling mechanisms are therefore MNAR. SNAR
is a peculiar class because it depends on the real class probability, while the class
needs to be positive by definition. The class probability refers to the probability
of an identical instance to this one being positive. Consider, for example, the
problem of classifying pages as interesting. If a page is moderately interesting to
you, some days you might like it while other days you might not. The labeling
mechanism in this case could depend on how much you like them and therefore
on the instance's class probability.

4   Learning with SAR Labeling Mechanisms

In this paper, we focus on SAR labeling mechanisms, where the key question
is how we can enable learning from SAR PU data. Our key insight is that the
labeling mechanism is also related to the notion of a propensity score from
causal inference [17]. In causal inference, the propensity score is the probability
that an instance is assigned to the treatment or control group. This probability
is instance-specific and based on a set of the instance's attributes. We use an
analogous idea and define the propensity score as the labeling probability for
positive examples:

Definition 1 (Propensity Score). *The propensity score for x, denoted $e(x)$,
is the label assignment probability for positive instances with attributes x,*

$$e(x) = \Pr(s = 1 | y = 1, x).$$

A crucial difference with the propensity score from causal inference is that our
score is conditioned on the class being positive.

We incorporate the propensity score when learning in a PU setting by using
the propensity scores to reweight the data. Our scheme generalizes an approach
taken for SCAR data [34, 11, 21] to the SAR setting. In causal inference, inverse-
propensity-scoring is a standard method where the examples are weighted with
the inverse of their propensity score [23, 17, 32]. This cannot be applied when
working with positive and unlabeled data, because we have zero probability for
labeling negative examples. But we can do a different kind of weighting. The
insight is that for each labeled example $(x_i, s = 1)$ that has a propensity score
e_i, there are expected to be $\frac{1}{e_i}$ positive examples, of which $\frac{1}{e_i} - 1$ did not get
selected to be labeled. This insight can be used in algorithms that use counts,
to estimate the correct count from the observed positives and their respective
propensity scores. In general, this can be formulated as learning with negative

weights: every labeled example gets a weight $\frac{1}{e_i}$ and for every labeled example a negative example is added to the dataset that gets a negative weight $1 - \frac{1}{e_i}$.

We now provide a theoretical analysis of the propensity-weighted method, to characterize its appropriateness. We consider two cases: (1) when we know the true propensity scores and (2) when we must estimate them from data. All the proofs are deferred to the appendix (`https://dtai.cs.kuleuven.be/software/sar`).

4.1   Case 1: True Propensity Scores Known

Standard evaluation measures, such as Mean Absolute Error (MAE), Mean Square Error (MSE) and log loss, can be formulated as follows:

$$R(\hat{\mathbf{y}}|\mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n} y_i\delta_1(\hat{y}_i) + (1 - y_i)\delta_0(\hat{y}_i),\,^1$$

where $\mathbf{y}$ and $\hat{\mathbf{y}}$ are vectors of size n containing, respectively, the true labels and predicted labels. The function $\delta_y(\hat{y})$ represents the cost for predicting $\hat{y}$ when the class is y, for example:

$$\text{MAE} : \delta_y(\hat{y}) = |y - \hat{y}|,$$
$$\text{MSE} : \delta_y(\hat{y}) = (y - \hat{y})^2,$$
$$\text{Log Loss} : \delta_1(\hat{y}) = -\ln\hat{y} \,,\ \delta_0(\hat{y}) = -\ln(1 - \hat{y}).$$

We can formulate propensity-weighted variants of these cost functions as:

Definition 2 (Propensity-Weighted Estimator). *Given the propensity scores* $\mathbf{e}$ *and PU labels* $\mathbf{s}$*, the propensity weighted estimator of* $R(\hat{\mathbf{y}}|\mathbf{y})$ *is*

$$\hat{R}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s}) = \frac{1}{n}\sum_{i=1}^{n} s_i\left(\frac{1}{e_i}\delta_1(\hat{y}_i) + (1 - \frac{1}{e_i})\delta_0(\hat{y}_i)\right) + (1 - s_i)\delta_0(\hat{y}_i),$$

where $\mathbf{y}$ *and* $\hat{\mathbf{y}}$ *are vectors of size n containing, respectively, the true labels and predicted labels. The function* $\delta_y(\hat{y})$ *represents the cost for predicting* $\hat{y}$ *when the class is y.*

This estimator is unbiased:

$$\mathbb{E}[\hat{R}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s})])$$
$$= \frac{1}{n}\sum_{i=1}^{n} y_i e_i\left(\frac{1}{e_i}\delta_1(\hat{y}_i) + (1 - \frac{1}{e_i})\delta_0(\hat{y}_i)\right) + (1 - y_i e_i)\delta_0(\hat{y}_i)$$
$$= \frac{1}{n}\sum_{i=1}^{n} y_i\delta_1(\hat{y}_i) + (1 - y_i)\delta_0(\hat{y}_i)$$
$$= R(\hat{\mathbf{y}}|\mathbf{y}).$$

To characterize how much the estimator can vary from the expected value, we provide the following bound:

---
[1] We assume that $0 < \hat{y} < 1$

Proposition 3 (Propensity-Weighted Estimator Bound). *For any predicted classes $\hat{\mathbf{y}}$ and real classes $\mathbf{y}$ of size n, with probability $1 - \eta$, the propensity-weighted estimator $\hat{R}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s})$ does not differ from the true evaluation measure $R(\hat{\mathbf{y}}|\mathbf{y})$ more than*

$$|\hat{R}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s}) - R(\hat{\mathbf{y}}|\mathbf{y})| \leq \sqrt{\frac{\delta_{max}^2 \ln \frac{2}{\eta}}{2n}},$$

with δ_{max} the maximum absolute value of cost function δ_y.

The propensity-weighted estimator can be used as the risk for Empirical Risk Minimization (ERM), which searches for a model in the hypothesis space $\mathcal{H}$ by minimizing the risk:

$$\hat{\mathbf{y}}_{\hat{R}} = \underset{\hat{\mathbf{y}} \in \mathcal{H}}{\operatorname{argmin}} \hat{R}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s}).$$

The following proposition characterizes how much the estimated risk for hypothesis $\hat{\mathbf{y}}_{\hat{R}}$ can deviate from its true risk.

Proposition 4 (Propensity-Weighted ERM Generalization Error Bound). *For a finite hypothesis space $\mathcal{H}$, the difference between the propensity-weighted risk of the empirical risk minimizer $\hat{\mathbf{y}}_{\hat{R}}$ and its true risk is bounded, with probability $1 - \eta$, by:*

$$R(\hat{\mathbf{y}}_{\hat{R}}|\mathbf{y}) \leq \hat{R}(\hat{\mathbf{y}}_{\hat{R}}|\mathbf{e}, \mathbf{s}) + \sqrt{\frac{\delta_{max}^2 \ln \frac{|\mathcal{H}|}{\eta}}{2n}}.$$

4.2   Case 2: Propensity Scores Estimated from Data

Often the exact propensity score is unknown, but we have an estimate $\hat{e}$ of it. In this case, the bias of the propensity-weighted estimator is:

Proposition 5 (Propensity-Weighted Estimator Bias).

$$bias(\hat{R}(\hat{\mathbf{y}}|\hat{\mathbf{e}}, \mathbf{s})) = \frac{1}{n} \sum_{i=1}^{n} y_i (1 - \frac{e_i}{\hat{e}_i}) \left(\delta_1(\hat{y}_i) - \delta_0(\hat{y}_i) \right).$$

From the bias, it follows that the propensity scores only need to be accurate for positive examples. An incorrect propensity score has a larger impact when the predicted classes have more extreme values (i.e., tend towards zero or one). Underestimated propensity scores are expected to result in a model with a higher bias. Lower propensity scores result in learning models that estimate the positive class to be more prevalent than it is, which results in a larger $(\delta_1(\hat{y}_i) - \delta_0(\hat{y}_i))$ for positive examples.

Side Note on Sub-Optimality of Expected Risk Another method that one might be inclined to use when incorporating the propensity score is to minimize the expected risk[2]:

---

[2] Derivation in appendix, available on `https://dtai.cs.kuleuven.be/software/sar`

$$\hat{R}_{\exp}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s}) = \mathbb{E}_{\mathbf{y}|\mathbf{e}, \mathbf{s}, \hat{\mathbf{y}}}\left[R(\hat{\mathbf{y}}|\mathbf{y})\right]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\left(s_i + (1 - s_i)\frac{\hat{y}_i(1 - e_i)}{1 - \hat{y}_i e_i}\right)\delta_1(\hat{y}_i) + (1 - s_i)\frac{1 - \hat{y}_i}{1 - \hat{y}_i e_i}\delta_0(\hat{y}_i).$$

However, the expected risk is not an unbiased estimator of the true risk and as a result, $\hat{\mathbf{y}}_{\hat{R}_{\exp}} = \mathrm{argmin}_{\hat{\mathbf{y}} \in \mathcal{H}}\hat{R}_{\exp}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s})$ is not expected to be the best hypothesis. In fact, the hypothesis of always predicting the positive class $\forall_i : \hat{y}_i = 1$ always has an expected risk $\hat{R}_{\exp}(\hat{\mathbf{y}}|\mathbf{e}, \mathbf{s}) = 0$.

5 Learning under the SAR Assumption

If the propensity scores for all examples are known (i.e., the exact labeling mechanism is known), they can be directly incorporated into the learning algorithm. However, it is more likely that they are unknown. Therefore, this section investigates how to permit learning in the SAR setting when the exact propensity scores are unknown. We discuss two such settings. The first is interesting from a theoretical perspective and the second from a practical perspective.

5.1 Reducing SAR to SCAR

Learning the propensity scores from positive and unlabeled data requires making additional assumptions: if any arbitrary instance can have any propensity score, then it is impossible to know if an instance did not get labeled because of a low propensity score or a low class probability. Therefore, the propensity score needs to depend on fewer attributes than the final classifier [17]. A simple way to accomplish this is to assume that the propensity function only depends on a subset of the attributes x_e called the *propensity attributes*:

$$\Pr(s = 1|y = 1, x) = \Pr(s = 1|y = 1, x_e)$$
$$e(x) = e(x_e).$$

Often, this a realistic assumption. It is trivially true if the labeling mechanism does not have access to all attributes (e.g., because some were collected later). It may also arise if a labeler cannot interpret some attributes (e.g., raw sensor values) or only uses the attributes that are known to be highly correlated with the class.

To see why this can be a sufficient assumption for learning in a SAR setting, consider the case where the propensity attributes x_e have a finite number of configurations, which is true if these attributes are all discrete. In this case, it is possible to partition the data into strata, with one stratum for each configuration of x_e. Within a stratum, the propensity score is a constant (i.e., all positive examples have the same propensity score) and can thus be determined using

standard SCAR PU learning techniques. Note that, as previously discussed, the SCAR assumption alone is not enough to enable learning from PU data, and hence one of the additional assumptions [3, 33, 28, 30, 1] must be made.

Reducing SAR to SCAR is interesting because it demonstrates that learning in the SAR setting is possible. However, it is suboptimal in practice as it does not work if x_e contains a continuous variable. Even for the discrete case, the number of configurations grows exponentially as the size of x_e increases. Furthermore, information is lost by partitioning the data. Some smoothness of the classifier over the propensity attributes is expected, but this is not encouraged when learning different classifiers for each configuration. Similarly, the propensity score itself is expected to be a smooth function over the propensity variables.

5.2   EM for Propensity Estimation

The problems with reducing the SAR to the SCAR case motivate the need to jointly search for a classifier and lower dimensional propensity score function that best explain the observed data. This approach also offers the advantage that it relaxes the additional assumptions: if they hold in the majority of the propensity attributes' configurations, the models' smoothness helps to overcome potential issues arising in the configurations where the assumptions are violated. This subsection presents a simple expectation-maximization method for simultaneously training the classification and the propensity score model. It aims to maximize the expected log likelihood of the combination of models.

Expectation Given the expected classification model $\hat{f}$ and propensity score model $\hat{e}$, the expected probability of the positive class $\hat{y}_i$ of instance x_i with label s_i is:[3]

$$\hat{y}_i = \Pr(y_i = 1 | s_i, x_i, \hat{f}, \hat{e})$$
$$= s_i + (1 - s_i) \frac{\hat{f}(x_i) \left(1 - \hat{e}(x_i)\right)}{1 - \hat{f}(x_i)\hat{e}(x_i)}.$$

Maximization Given the expected probabilities of the positive class $\hat{y}_i$, the models f and e are trained to optimize the expected log likelihood:

$$\operatorname{argmax}_{f,e} \sum_{i=1}^{n} \mathbb{E}_{y_i | x_i, s_i, \hat{f}, \hat{e}} \ln \Pr(x_i, s_i, y_i | f, e)$$
$$= \operatorname{argmax}_f \sum_{i=1}^{n} \left[\hat{y}_i \ln f(x_i) + (1 - \hat{y}_i) \ln(1 - f(x_i))\right],$$
$$\operatorname{argmax}_e \sum_{i=1}^{n} \hat{y}_i \left[s_i \cdot \ln e(x_i) + (1 - s_i) \cdot \ln(1 - e(x_i))\right]$$

---

[3] All derivations for this section are in the appendix, available on `https://dtai.cs.kuleuven.be/software/sar`.

From the maximization formula, it can be seen that to optimize the log likelihood, both models need to optimize the log loss of a weighted dataset. The classification model f receives each example i twice, once as positive, weighted by the expected probability of it being positive $\hat{y}_i$ and once as negative, weighted by the expected probability of it being negative $(1 - \hat{y}_i)$. The propensity score model e receives each example once, positive if the observed label is positive and negative otherwise, weighted by the expected probability of it being positive $\hat{y}_i$.

Because this approach minimizes log loss, it will work best if the classes are separable. If the classes are not separable, then the trained classification model is not expected to be the optimal one for the trained propensity model (see previous section). In that case, it is advisable to retrain the classifier with the obtained propensity score, using the propensity-weighted risk estimation method.

The classification model is initialized by fitting a balanced model which considers the unlabeled examples as negative. The propensity score model is initialized by using the classification model to estimate the probabilities of each unlabeled example being positive.

Classic EM converges when the log likelihood stops improving. However, the likelihood could stop improving before the propensity score model has converged. Convergence is therefore formulated as convergence of both the log likelihood and the propensity model. We measure the change in the propensity score model by the average slope of the minimum square error line through the propensity score prediction of the last n iterations.

6   Empirical Evaluation

This section illustrates empirically that the SAR assumption facilitates better learning from SAR PU data. We compare both SAR and SCAR methods, so that the gain of using an instance-dependent propensity score over a constant label frequency can be observed. More specifically, we address the following questions:

Q1. Does propensity score weighting (SAR) improve classification performance over assuming that data is SCAR and using class prior weighting?
Q2. Can the propensity score function be recovered?
Q3. Does the number of propensity attributes affect the performance?

6.1   Data

We use eight real-world datasets which cover a range of application domains such as text, images and tabular data. These datasets are summarized in Table 1. Since the 20 News Groups, Cover Type, Diabetes and Image Segmentation datasets are originally multi-class datasets, we first transformed them by either grouping or ignoring classes. For 20 News Groups,[4] we distinguish between computer (pos) and recreational (neg) documents. After removing their headers, footers, quotes, and English stop words, the documents were transformed to

---

[4] http://archive.ics.uci.edu/ml/

Table 1: Datasets

Dataset	# Instances	# Attrib	Pr($y = 1$)
20 Newsgroups	3,979	200	0.55
Adult	48,842	14	0.24
Breast Cancer	683	9	0.35
Cover Type	581,0124	54	0.49
Diabetes	99,492	127	0.11
Image Segm.	2,310	18	0.43
Mushroom	8,124	111	0.48
Splice	3,175	60	0.52

200 word occurrence attributes using the Scikit-Learn[5] count vectorizer. For Cover Type,[4] we distinguish the Lodgepole Pine (pos) from all other cover types (neg). The Diabetes[4] data was preprocessed in a similar manner to Strack et al. [35]. Additionally, we dropped attributes with the same value in 95% of the examples, and replaced uncommon attribute values by "other". The positive class is patients being readmitted within 30 days. Image Segmentation[4] was used to distinguish between nature (sky, grass or foliage) and other scenes (brickface, cement, window, path). Adult,[4] Breast Cancer,[4] Mushroom,[4] and Splice[6] were used as is. To enable using logistic regression, all the datasets were further preprocessed to have exclusively continuous attributes, scaled between -1 and 1. Multivalued attributes were binarized.

6.2   Methodology and Approaches

Constructing Datasets. First, we extended each dataset with four artificial binary propensity attributes $x_e^{(i)} \in \{0, 1\}$. Therefore, we clustered each dataset into five groups (based on the attribute values) and generated for each group a random distribution between propensity attribute values $\{0, 1\}$. Intuitively, this corresponds to a scenario where examples that are in the same cluster have the same prior probability of belonging to the positive class. However, which examples are labeled depend on the propensity attributes.

Next, the datasets were randomly partitioned into train (80%) and test (20%) sets five times. For each of the five train-test splits, we transformed the data into positive and unlabeled datasets by sampling the examples to be labeled according to the following propensity score:

$$e(x_e) = \prod_{i=1}^{k} \left(p_{\text{low}}^{1 - x_e^{(i)}} \cdot p_{\text{high}}^{x_e^{(i)}} \right)^{\frac{1}{k}}$$

---

[5] http://scikit-learn.org

[6] Available on LIBSVM Data repository https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

This gives propensity scores between p_{low} and p_{high}, with all artificial propensity attributes x_e attributing equally to it. In our experiments the propensity scores were between 0.2 and 0.8. We generated five of such labelings for each of the five train-test splits and report the average performance over these 25 experiments.

Approach. We compare the classification performance of our EM method under the SAR assumption[7] against four baselines. First, we assume the data is SCAR and compare against two state-of-the art methods to estimate the class prior: KM2 from Ramaswamy et al. [30][8] and TI*c*E from Bekker and Davis [1][9] with standard settings. Second, we use the naive baseline which assumes that all unlabeled examples belong to the negative class (denoted Naive). Finally, as an illustrative upper bound on performance, we show results when given fully supervised data (denoted Sup.). All approaches use logistic regression with default parameters from Scikit-Learn[10] as the base classifier for the classification model and also for the propensity score model in the SAR setting.
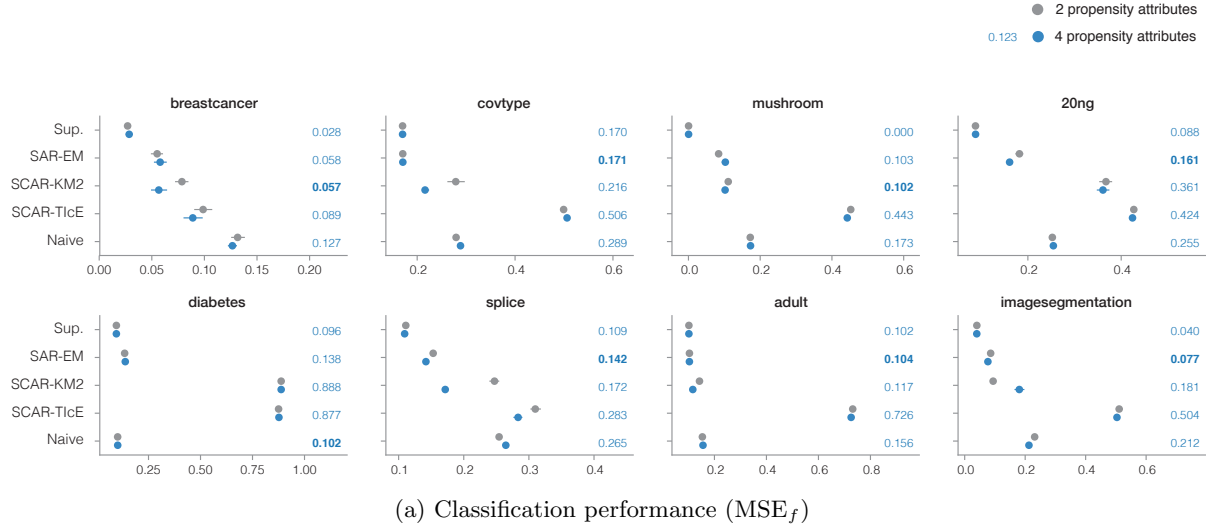
6.3   Results

Q1. Figure 1 compares SAR-EM to all baselines. Because we are considering models that predict probabilities for binary classification problems, we report two standard metrics. First, we report MSE which measures the quality of the model's probability estimates [13]. Second, we report ROC-AUC, which measures predictive performance. When the propensity attributes are known, learning both the propensity score and the classification model from the data outperforms assuming the data is SCAR and learning under that assumption. Based on each method's average ROC-AUC ranks over all eight benchmark datasets, the Friedman test [8] rejects the null-hypothesis that all methods perform the same ($p < 0.001$), regardless of the number of propensity attributes used. Moreover, using the Nemenyi post-hoc test on the ranks, the performance of our SAR-EM method is significantly better ($p < 0.01$) than the naive approach, KM2 and TI*c*E. Note that the naive approach sometimes outperforms the SCAR approaches. This can be explained by the SCAR methods' goal of predicting the correct ratio of the instance space as positive. When it picks the wrong subpsace to get to this ratio, it results in both false positives and false negatives, where the more conservative naive approach would only give the false negatives.
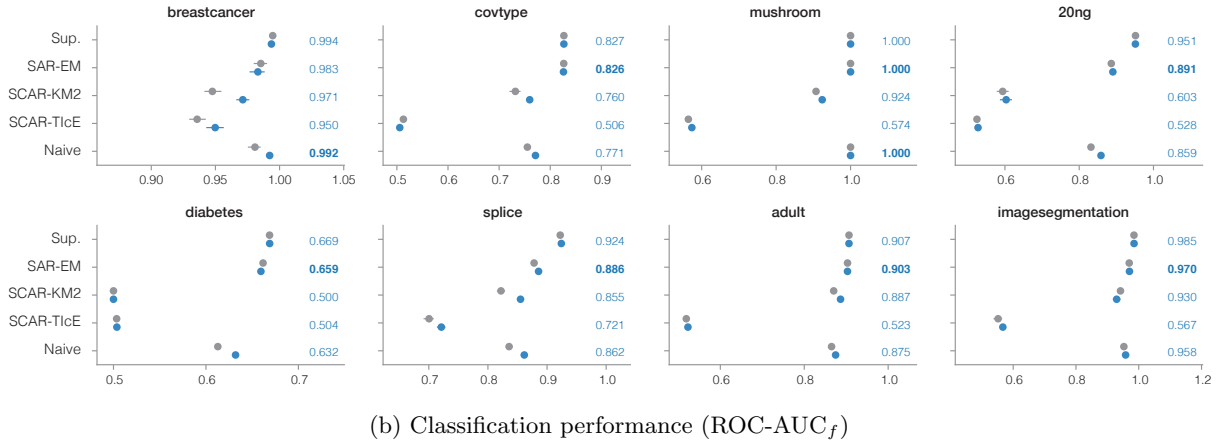
---

[7] Implementation available on `https://dtai.cs.kuleuven.be/software/sar`

[8] `http://web.eecs.umich.edu/~cscott/code/kernel_MPE.zip`

[9] `https://dtai.cs.kuleuven.be/software/tice/`

[10] https://scikit-learn.org/stable/

(a) Classification performance (MSE_f)

(b) Classification performance ($\mathrm{ROC\text{-}AUC}_f$)

Fig. 1: Given SAR data, jointly learning both the unknown propensity scores and the classification model almost always outperforms learning under the SCAR assumption. The dots correspond to the mean performance for respectively two (grey) and four (blue) propensity attributes. The error bars represent a 95% confidence interval around the mean. The exact performance metric value is given on the right for the setting with four propensity attributes, with the best performing algorithm highlighted in bold (ignoring Supervised).

Q2. To evaluate the quality of the learned propensity scores for each example, we report the MSE [13]. Except for the mushroom dataset, the EM method always obtains very accurate propensity score estimates with MSEs below 0.1 (Figure 2). Furthermore, the MSEs are often very close to zero.
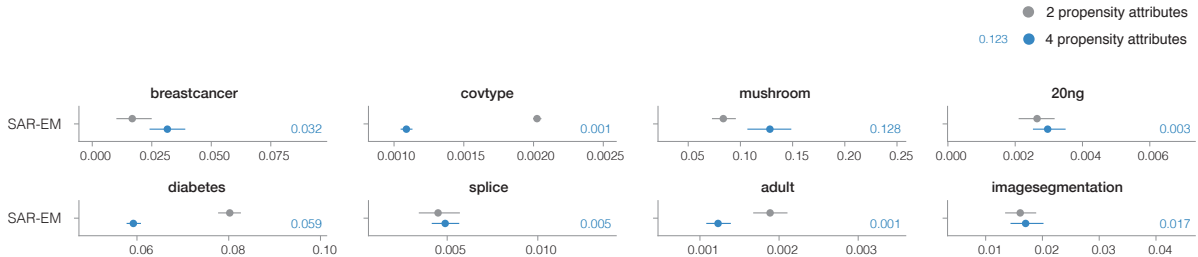

Fig. 2: Accuracy of the propensity score estimates (MSE_e). The dots correspond to the mean performance for respectively two (grey) and four (blue) propensity attributes. The error bars represent a 95% confidence interval around the mean. The exact performance metric value is given on the right for the setting with four propensity attributes.

Q3. Finally, we observe no correlation between the number of propensity attributes and the MSE and ROC-AUC of the classification model, nor the MSE of the propensity score estimates (Figure 1a).

7   Related Work

PU learning is an active area and for a broad overview see [2]. This work focuses on approaches that modify learning methods by exploiting the assumptions about the labeling mechanism (e.g., [9, 12, 16, 22, 27, 36]) for the single training set scenario. The key difference is that this paper generalizes past work, which has focused on the SCAR assumption, to the less restrictive SAR setting. The weighting scheme used in this paper has been used under the SCAR assumption [34, 11, 21]. Furthermore, a special case of this method has been applied matrix completion [16].

Almost all PU learning work that we are aware of focuses on the SCAR setting. One recent exception assumes that the probability of observing a label for a positive example depends on how difficult the example is to label [14]. That is, the more similar a positive example is to a negative one, the less likely it is to be labeled. The difficulty of labeling is defined by the *probabilistic gap* $\Delta \Pr(x) = \Pr(y = 1|x) - \Pr(y = 0|x)$ [14]. Based on properties of the probabilistic gap, it is possible to identify reliable positive and negative examples [14]. Because the probabilistic gap labeling mechanism depends on the attribute values x, it

is a specific case of SAR. Concretely, it assumes a propensity score that is a non-negative, monotonically decreasing function of the probabilistic gap $\Delta \Pr(x)$.

PU learning is a special case of semi-supervised learning [6]. It is also related to one-class learning [20]. The work on dealing with biases in the observed ratings for recommender systems [32] and implicit feedback [19] is closely related to ours. They also make use of propensity scores to cope with the biases. However, there is a crucial difference: they perform inverse propensity weighting, which is not possible in our setting. In those works, the propensity score for each example is non-zero. In contrast, in PU learning, the propensity score for any negative example is zero: you never observe these labels. Moreover, they assume that examples for the full label space are available (e.g., observe at least one rating of each category for recommender systems) to learn the propensity model, which is not the case for PU learning because we have no known negative examples. This necessitates different ways to learn the propensity scores and weigh the data in our setting.

8   Conclusions

We investigated learning from SAR PU data: positive and unlabeled data with non-uniform labeling mechanisms. We proposed and theoretically analyzed an empirical-risk-minimization based method for weighting PU datasets with the propensity scores to achieve unbiased learning. We explored which assumptions are necessary to learn from SAR PU data generated by an unknown labeling mechanism and proposed a practical EM-based method for this setting. Empirically, for SAR PU data, our proposed propensity weighted method offers superior predictive performance over making the SCAR assumption. Moreover, we are able to accurately estimate each example's propensity score.

Acknowledgments

Bibliography

[1] Bekker, J., Davis, J.: Estimating the class prior in positive and unlabeled data through decision tree induction. In: Proceedings of the 32th AAAI Conference on Artificial Intelligence (2018)

[2] Bekker, J., Davis, J.: Learning from positive and unlabeled data: A survey. arXiv preprint arXiv:1811.04820 (2018)

[3] Blanchard, G., Lee, G., Scott, C.: Semi-supervised novelty detection. Journal of Machine Learning Research 11, 2973–3009 (2010)

[4] Blockeel, H.: Pu-learning disjunctive concepts in ilp. In: ILP 2017 late breaking papers (2017)

[5] Chang, S., Zhang, Y., Tang, J., Yin, D., Chang, Y., Hasegawa-Johnson, M.A., Huang, T.S.: Positive-unlabeled learning in streaming networks. In: Proceedings of the 22nd ACM Conference on Knowledge Discovery and Data Mining. pp. 755–764 (2016)

[6] Chapelle, O., Scholkopf, B., Zien, A.: Semi-supervised learning. IEEE Transactions on Neural Networks 20(3), 542–542 (2009)

[7] Claesen, M., De Smet, F., Suykens, J.A.K., De Moor, B.: A robust ensemble approach to learn from positive and unlabeled data using svm base models. Neurocomputing 160, 73–84 (2015)

[8] Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine learning research 7(Jan), 1–30 (2006)

[9] Denis, F., Gilleron, R., Letouzey, F.: Learning from positive and unlabeled examples. Theoretical Computer Science 348(1), 70–83 (2005)

[10] Denis, F.: Pac learning from positive statistical queries. In: ALT (1998)

[11] Du Plessis, M., Niu, G., Sugiyama, M.: Convex formulation for learning from positive and unlabeled data. In: International Conference on Machine Learning. pp. 1386–1394 (2015)

[12] Elkan, C., Noto, K.: Learning classifiers from only positive and unlabeled data. In: KDD (2008)

[13] Flach, P.: Performance evaluation in machine learning: The good, the bad, the ugly and the way forward. In: Proceedings of the 33th AAAI Conference on Artificial Intelligence (2019)

[14] He, F., Liu, T., Webb, G.I., Tao, D.: Instance-dependent pu learning by bayesian optimal relabeling. arXiv preprint arXiv:1808.02180 (2018)

[15] Hou, M., Chaib-draa, B., Li, C., Zhao, Q.: Generative adversarial positive-unlabelled learning. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. pp. 2255–2261 (7 2018)

[16] Hsieh, C.J., Natarajan, N., Dhillon, I.: PU learning for matrix completion. In: International Conference on Machine Learning. pp. 2445–2453 (2015)

[17] Imbens, G.W., Rubin, D.B.: Causal inference in statistics, social, and biomedical sciences. Cambridge University Press (2015)

[18] Jain, S., White, M., Radivojac, P.: Estimating the class prior and posterior from noisy positives and unlabeled data. In: NIPS (2016)

[19] Joachims, T., Swaminathan, A., Schnabel, T.: Unbiased learning-to-rank with biased feedback. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. pp. 781–789 (2017)

[20] Khan, S., Madden, M.: One-class classification: taxonomy of study and review of techniques. The Knowledge Engineering Review (2014)

[21] Kiryo, R., Niu, G., du Plessis, M.C., Sugiyama, M.: Positive-unlabeled learning with non-negative risk estimator. In: Advances in Neural Information Processing Systems. pp. 1675–1685 (2017)

[22] Lee, W.S., Liu, B.: Learning with positive and unlabeled examples using weighted logistic regression. In: ICML. pp. 448–455 (2003)

[23] Little, R., Rubin, D.: Statistical analysis with missing data. John Wiley & Sons (2002)

[24] Liu, B., Dai, Y., Li, X.L., Lee, W., Yu, P.: Building text classifiers using positive and unlabeled examples. In: ICDM. pp. 179–186 (2003)

[25] Marlin, B.M., Zemel, R.S.: Collaborative prediction and ranking with non-random missing data. In: Proceedings of the 2009 ACM Conference on Recommender Systems. pp. 5–12 (2009)

[26] Mordelet, F., Vert, J.P.: A bagging svm to learn from positive and unlabeled examples. Pattern Recognition Letters pp. 201–209 (2014)

[27] Northcutt, C.G., Wu, T., Chuang, I.L.: Learning with confident examples: Rank pruning for robust classification with noisy labels. In: Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence (2017)

[28] du Plessis, M.C., Niu, G., Sugiyama, M.: Class-prior estimation for learning from positive and unlabeled data. Machine Learning 106, 463–492 (2015)

[29] du Plessis, M.C., Sugiyama, M.: Class prior estimation from positive and unlabeled data. IEICE Transactions 97-D, 1358–1362 (2014)

[30] Ramaswamy, H., Scott, C., Tewari, A.: Mixture proportion estimation via kernel embedding of distributions. In: ICML (2016)

[31] Rubin, D.: Inference and missing data. Biometrika (1976)

[32] Schnabel, T., A.Swaminathan, A.Singh, Chandak, N., Joachims, T.: Recommendations as treatments: Debiasing learning and evaluation. In: ICML (2016)

[33] Scott, C.: A rate of convergence for mixture proportion estimation, with application to learning from noisy labels. In: AISTATS (2015)

[34] Steinberg, D., Cardell, N.S.: Estimating logistic regression models when the dependent variable has no variance. Communications in Statistics-Theory and Methods 21(2), 423–450 (1992)

[35] Strack, B., DeShazo, J., Gennings, C., Olmo, J., Ventura, S., Cios, K., Clore, J.: Impact of hba1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. BioMed research international (2014)

[36] Ward, G., Hastie, T., Barry, S., Elith, J., Leathwick, J.R.: Presence-only data and the em algorithm. Biometrics 65(2), 554–563 (2009)

[37] Zupanc, K., Davis, J.: Estimating rule quality for knowledge base completion with the relationship between coverage assumption. In: Proceedings of the Web Conference (2018)