Class Prior Estimation in Active Positive and Unlabeled Learning

Lorenzo Perini, **Vincent Vercruyssen** and **Jesse Davis**
DTAI Research group, KU Leuven, Belgium
lorenzo.perini@kuleuven.be, vincent.vercruyssen@kuleuven.be, jesse.davis@kuleuven.be

Abstract

Estimating the proportion of positive examples (i.e., the class prior) from positive and unlabeled (PU) data is an important task that facilitates learning a classifier from such data. In this paper, we explore how to tackle this problem when the observed labels were acquired via active learning. This introduces the challenge that the observed labels were not selected completely at random, which is the primary assumption underpinning existing approaches to estimating the class prior from PU data. We analyze this new setting and design an algorithm that is able to estimate the class prior for a given active learning strategy. Empirically, we show that our approach accurately recovers the true class prior on a benchmark of anomaly detection datasets and that it does so more accurately than existing methods.

1 Introduction

Positive and unlabeled (PU) learning [Elkan and Noto, 2008] is a special case of binary classification where a learner only has access to labeled positive examples and unlabeled examples. The unlabeled data contains both positive and negative examples. PU data arises naturally in many different applications, such as anomaly detection, where the goal is to detect abnormal examples in a dataset and one typically only has access to normal and unlabeled data to construct the classifier [Trittenbach *et al.*, 2019].

The typical PU learning setup assumes the learner is given a PU dataset with a fixed set of observed positive labels (i.e., it cannot acquire any new labels). Moreover, it is commonly assumed that the observed positive labels were "selected completely at random". This is known as the SCAR assumption and it states that the probability of observing a positive example's label is a constant [Elkan and Noto, 2008]. Under the SCAR assumption, there are a number of different ways to enable learning from PU data; see Bekker and Davis [2020] for an overview. One of the most prominent ones involves estimating the class prior from data [Du Plessis and Sugiyama, 2014; du Plessis *et al.*, 2015; Bekker and Davis, 2018; Jain *et al.*, 2016b; Jain *et al.*, 2016a].

This paper explores a different setting that combines PU learning with *active learning*. The learner initially has access to only unlabeled data, and positive labels are gradually acquired using an active learning strategy. In anomaly detection applications, for instance, one often starts with a completely unlabeled dataset and labels are acquired via active learning because the labeling process is costly [Vercruyssen *et al.*, 2018]. Because anomalies are rare events and not well-understood, the user almost always ends up labeling only normal examples (i.e., examples from the positive class). Another class of problems where active learning would only return positive labels arises when measuring the interestingness of ads, likes, or Facebook friend suggestions based on click data. Using active learning introduces the challenge that the SCAR assumption no longer holds as the active learning strategy has a clear bias when selecting examples to be labeled.

This paper analyzes the problem of *estimating the class prior from PU data* where the positive labels were acquired using active learning. The class prior is the proportion of positive examples in the data. We make four contributions. First, we show how to estimate the class prior using each example's *propensity score*, which is the probability that a positive example is selected by the active learning strategy to be labeled. Second, we prove that the estimate of the class prior converges to the true class prior. Third, we propose a method called CAPE (*Class prior estimation in Active Pu lEarning*) for estimating the class prior in practice. Finally, we empirically evaluate CAPE in the context of anomaly detection. The experiments highlight that CAPE is able to make more accurate estimates of the class prior than current state-of-the-art methods.

2 Preliminaries

In a PU learning, one example can be seen as a triple $\{x, y, s\}$, where x is the vector of features, y the binary class and s indicates if the example has been selected to be labeled. More formally, a PU dataset X_D is a set of examples in a probability space $(\mathcal{X}, \Im, \mu)$. Here, $\mathcal{X} = X \times \mathcal{Y} \times \mathcal{S}$, where $X = \mathbb{R}^d$ is the feature space, $\mathcal{Y} = \{0, 1\}$ is the class label space, and $\mathcal{S} = \{0, 1\}$ is label choice space. Finally, $\Im$ represents a σ-algebra over $\mathcal{X}$ and $\mu = \mu(x, y, s)$ is the probability distribution over possible triples drawn from $\mathcal{X}$.

In PU learning, the true label of any labeled example is assumed to be positive. It is widely assumed that labels are selected completely at random (SCAR), meaning that the probability of an example to be labeled depends only on its *true* label. With the introduction of the label frequency c, the proportion of labeled examples, the class prior $\mu(y=1)$ can be estimated in a straightforward fashion as $\mu(s=1)/c$ [Bekker and Davis, 2018; Bekker and Davis, 2020].

More recent work relaxes the SCAR assumption by assuming that the labels are selected at random (SAR). This means that the probability of an example to be labeled depends not only on its true label but also on its features [Bekker *et al.*, 2019]. The class prior of a dataset is now computed as a function of the *propensity score* [Bekker *et al.*, 2019], where an example's propensity score $e(x) = \mu(s=1|x, y=1)$ is defined as the probability that the example is labeled. In our setting, however, the examples are only labeled when they are queried by the active learning strategy, which we show slightly alters the definition of propensity score.

3 Class Prior Estimation in Active PU Learning

This paper tackles the following problem:

Given: a dataset $X_D = \{x_1, \ldots, x_n\} \sim_{i.i.d.} \mu_X$ drawn $i.i.d.$ from the population with distribution μ_X; a tradition classifier h trained on PU data; an active learning strategy to obtain k labeled examples;

Estimate: the class prior of X_D.

Applying the active learning strategy results in a set of k labeled positive examples and $n-k$ unlabeled examples. Subsection 3.1 describes how to use this partially labeled data to estimate the *class prior*. However, computing the class prior requires knowing the propensity scores, which is the probability of the example being selected by the active learning strategy to be labeled. Subsection 3.2 discusses how to tackle this issue. Finally, Subsection 3.3 shows that if the computed propensity scores are accurate, the estimated class prior theoretically converges to the true class prior.

3.1 Estimating the Class Prior

For now we assume that the propensity scores are known. Intuitively, the class prior can be derived by combining the proportion of examples labeled positive by the user, and the expected proportion of positive examples among the remaining unlabeled examples:

$$\begin{aligned}\mu(y=1) =&\mathbb{E}_x \mu(y=1|x) = \\ &\mathbb{E}_x[s(x)] + \mathbb{E}_x\left[(1-s(x))\frac{\widehat{y}(1-e(x))}{1-\widehat{y}\,e(x)}\right],\end{aligned}$$

where $s(x)$ is 1 if the example has been labeled and 0 otherwise, and $\widehat{y} = \widehat{h}(x) = \mu\,(y=1|x, e(x), \widehat{y}) \in (0,1)$ is the probability that the example x belongs to the positive class according to the trained classifier h trained on the PU data.

We derived the previous identity from Bekker et al. [2019] by applying the mean operator on both sides. The main assumption is that h returns *accurate* estimates of the class probabilities. Thus, class prior estimates also depend on the correctness of this assumption. Roughly speaking, a labeled example contributes fully towards the positive class prior, while an unlabeled example only contributes its probability of being positive weighted by its probability of being labeled.

3.2 Estimating the Propensity Scores

The propensity score for an example x is the proportion of all datasets containing x that can be drawn from the distribution μ_X, in which x's label is observed. Whether x's label is observed depends on both the dataset and the active learning strategy: whether x is selected to be labeled will depend on whether another more informative unlabeled example (according to the active learning strategy) is present in the dataset. This is further complicated by the fact that, in practice, we only have one dataset from which to estimate propensity scores.

Conceptually an example's propensity score can be computed as:

$$\begin{aligned}e(x) &= \mu(s=1|x, y=1)\\ &= \int_{\mathbb{R}^d} \mu(s=1|x, y=1, X)\; d\mu(X|x, y=1)\\ &= \sum_{\{X\subset\mathbb{R}^d:\ x\in X\}} e(x|X)\cdot d\mu(X|x, y=1), \qquad (1)\end{aligned}$$

where the sum is over the infinitely many possible datasets X that can be drawn from $\mathbb{R}^d$, $d\mu(X|x, y=1)$ is the infinitesimal probability to draw a specific sample X, and $e\left(x|\,X\right)$ is what we call the *grounded propensity score* given the observed dataset X. Our key insight is that for a fixed dataset X, $e(x|X)$ is either 0 or 1: either the example is in the top-k most informative examples of the dataset according to the active learning strategy (1) or it is not (0).[1]

Next, we tackle the problem of summing over infinitely many possible datasets. This can be solved by decomposing the problem into: (1) an inner loop summing over all subsets of $\mathbb{R}^d$ with a given cardinality m, and (2) an outer loop summing over all possible cardinalities.

Inner loop: summing over the subsets. Considering only those subsets of $\mathbb{R}^d$ with cardinality m, we define:

$$\overline{e_m(x)} := \sum_{\{X\subset\mathbb{R}^d, |X|=m, x\in X\}} e(x|X)\cdot d\mu(X|x, y=1). \quad (2)$$

Taking the sum over all possible subsets of cardinality m poses two questions. First, is this sum actually countable, even if the set is apparently *uncountable*? And second, if so, how can we compute it? The countability of the sum is proven using the following theorem.

Theorem 1. *Let I be any set, $g\colon I \to [0, +\infty)$. Let's define*

$$\sum_{i\in I} g(i) = \sup\left\{\sum_{i\in J} g(i)\colon\ J\subset I, |J| < +\infty\right\}.$$

Then, if $\sum_{i\in I} g(i) < +\infty$, the set

$$A = \{i \in I\colon g(i) \neq 0\}$$

is at most countable.

[1] That is under the assumption that there is no randomization in the active learning strategy or learning algorithm.

Proof. Let's consider $\varepsilon > 0$ and $A_\varepsilon = \{i \in I\colon g(i) > \varepsilon\}$. Without loss of generality, we suppose that $|A_\varepsilon| = +\infty$ and that its cardinality is countable. Then there exists a sequence $\{x_n\}_{n\in\mathbb{N}} \subseteq A_\varepsilon$ such that $g(x_n) > \varepsilon$ for all $n \in \mathbb{N}$. So, since ε is a constant, the inequality

$$\sum_{n=1}^{\infty} g(x_n) > \sum_{n=1}^{\infty} \varepsilon = +\infty$$

holds. This leads to a contradiction:

$$\sup\left\{\sum_{i\in J} g(i)\colon J \subset I, |J| < +\infty\right\} = +\infty.$$

As a result the set A_ε is finite. Because of the arbitrary choice of ε, let's choose $\varepsilon = \frac{1}{t}$, for $t \in \mathbb{N}$. Now it is evident that

$$A = \{i \in I\colon g(i) \neq 0\} = \bigcup_{t\in\mathbb{N}} A_{\frac{1}{t}}$$

is countable, since it is countable union of finite sets. □

To answer the second question, the sum over all possible subsets with cardinality m can be *approximated* through a sequence of r subsets. Since the sum is actually countable, there exists a sequence of sets $X^{(m_1)}, \ldots, X^{(m_r)}, \ldots$ with non-zero values such that

$$\sum_{i=m_1}^{m_r} e\left(x|X^{(i)}\right) \cdot d\mu\left(X^{(i)}|x, y=1\right) \xrightarrow{r\to\infty} \overline{e_m(x)}.$$

Subsections 4.1 and 4.2 explain how to compute this sequence in practice.

Outer loop: summing over the cardinalities. Next, we need to sum over all possible cardinalities to arrive at the propensity score for an example x. So, we define

$$e_m(x) := \sum_{j=1}^{m} \overline{e_j(x)}$$

where $\overline{e_j(x)}$ is defined in Equation 2. This sequence converges to the actual propensity score for m going to $+\infty$.

Proof.

$$\begin{aligned}
&\sum_{j=1}^{m} \overline{e_j(x)} = \sum_{j=1}^{m} \sum_{\{X\subset\mathbb{R}^d\colon |X|=j,\, x\in X\}} e(x|X)\cdot d\mu(X|x, y=1)\\
&= \sum_{\{X\subset\mathbb{R}^d\}} e(x|X)\cdot d\mu(X|x, y=1) \sum_{j=1}^{m} \mathbb{1}_{\{|X|=j,\, x\in X\}}(X)\\
&= \sum_{\{X\subset\mathbb{R}^d\}} e(x|X)\cdot d\mu(X|x, y=1) \mathbb{1}_{\bigcup_{j=1}^{m}\{|X|=j,\, x\in X\}}(X)\\
&\longrightarrow \sum_{\{X\subset\mathcal{X},\, x\in X\}} e(x|X)\cdot d\mu(X|x, y=1) \quad \text{for } m\to\infty.
\end{aligned}$$

□

So, theoretically, it is possible to determine $\widetilde{m}$ and some small error ε such that, for $\widetilde{m}$ "large enough",

$$\|e_{\widetilde{m}}(x) - e(x)\| = \left\|\sum_{j=1}^{\widetilde{m}} \overline{e_j(x)} - e(x)\right\| < \varepsilon.$$

Practical computation of the propensity scores. In practice, in order to compute the outer and inner loop of the sum in Equation 1, we need to choose the parameters m and r. Their values, however, are restricted by the observed dataset X_D: m is maximally equal to n and the r subsets can only be drawn from X_D. Moreover, computing the inner loop over all possible subsets of a certain cardinality is prohibitively expensive. For instance, if X_D contains 2000 examples and the cardinality is 1000, we would need to loop over $> 10^{600}$ possible subsets. We circumvent this issue by applying standard counting techniques on the available dataset to directly estimate $\overline{e_m(x)}$. Then, through an average approximation of the probabilities, the inner loop can be completely avoided. Section 4 explains this in detail.

3.3 Convergence to the true Class Prior

If we obtain an accurate estimate of the propensity scores, the convergence of the estimated class prior to the true class prior follows from the following theorem:

Theorem 2. *Assume that there exists a sequence $e_m(x)$ of functions which converges to the propensity score $e(x)$ for all $x \in X$. Then, given the sequence of class priors*

$$\mu_m(y=1) := \mathbb{E}_x\left[s + (1-s)\frac{\widehat{y}\,(1-e_m(x))}{1-\widehat{y}\,e_m(x)}\right],$$

the following result holds

$$\mu_m(y=1) \longrightarrow \mu(y=1) \qquad \textit{for } m\to\infty.$$

Proof. The hypothesis means that

$$\lim_{m\to\infty} e_m(x) = e(x) \qquad \forall x\ \in X.$$

Then,

$$\begin{aligned}
&\lim_{m\to\infty} \mu_m(y=1) = \lim_{m\to\infty} \mathbb{E}_x\left[s + (1-s)\frac{\widehat{y}\,(1-e_m(x))}{1-\widehat{y}\,e_m(x)}\right]\\
&= \mathbb{E}_x \lim_{m\to\infty}\left[s + (1-s)\frac{\widehat{y}\,(1-e_m(x))}{1-\widehat{y}\,e_m(x)}\right]\\
&= \mathbb{E}_x\left[s + (1-s)\frac{\lim_{m\to\infty}\widehat{y}\,(1-e_m(x))}{\lim_{m\to\infty} 1-\widehat{y}\,e_m(x)}\right]\\
&= \mathbb{E}_x\left[s + (1-s)\frac{\widehat{y}\,(1-e(x))}{1-\widehat{y}\,e(x)}\right] = \mu(y=1),
\end{aligned}$$

where the first step is due to the dominated convergence theorem (the sequence of functions is bounded because of probabilities) and the second equality holds since both the factors are non zero and their limit exists for any x. □

4 Active PU Learning

The active learning strategy asks the user to label those examples that are the most *informative*, according to some criterion, for learning the classifier. While it is perfectly possible that strategy will query the labels for examples belonging to both classes, as discussed in the Introduction situations will arise where a user will only label positive examples. We look at both an ideal and a realistic case. In the ideal case, we assume that the user is a *perfect oracle* (subsection 4.1). The queried examples are labeled only if their real class is positive, and in all other cases, the user does not know the true label and the queried examples remain unlabeled. In the realistic case, we assume that the user is an *imperfect oracle* (subsection 4.2). The user is not always able to recognize the examples, so that with a certain probability the queried example might not be labeled. In addition, there is a low probability that the user might label a queried example as positive while its true label is negative.

Next, we describe CAPE (*Class prior estimation in Active Pu lEarning*) which is our practical approach for estimating the class prior from data. Its estimate depends on whether the user is a perfect or imperfect oracle which changes how $\overline{e_m(x)}$ is estimated. In the ideal case, the probability to label an example only depends on its true label. In the realistic case, it also depends on a probability measure that represents the user's uncertainty about its true label.

4.1 Propensity Scores under Perfect Oracles

The direct computation of $\overline{e_m(x)}$ breaks down into three parts. First, we compute the probability that x is labeled in a given subset with cardinality m. Second, we multiply this probability with the count of how many times x is part of a subset of size m sampled from the dataset X_D. Third, we compute the expected probability of sampling these subsets.

Label probability. In the ideal case, the user is a perfect oracle. If a truly positive example is selected to be labeled, the user always labels it correctly.[2] Therefore, given a subset X of the dataset X_D, an example in this subset is labeled if it is in the top-k most informative examples of X (denoted as X_k) according to the active learning strategy because those are the examples that will be queried. If we use the active learning strategy to construct a *global ranking* of all the examples in X_D where the higher ranked examples are queried first, we can reasonably assume that this ranking is preserved for the examples in any subset of X_D. Let x_{j+1} be $j+1$-th example in the global ranking (G.R.). The probability that x_{j+1} is queried is equal to the probability that it is in the top-k of a sampled subset X:

$$\mathbb{P}(x_{j+1} \in X_k) = \begin{cases} 1 & \text{if } x_{j+1} \in \text{top-}k \text{ of G.R.} \\ \displaystyle\sum_{t=\max\{0,m+j-N\}}^{k-1} \frac{\binom{j}{t} \cdot \binom{n-j-1}{m-t-1}}{\binom{n-1}{m-1}} & \text{otherwise.} \end{cases} \tag{3}$$

where t is the number of examples ranked higher in the global ranking than x_{j+1} in any given subset X of X_D.

Counting the subsets. The number of times x_{j+1} will be chosen among n elements by simultaneously selecting m examples is:

$$|\{X \subseteq X_D \colon x_{j+1} \in X,\ |X| = m\}| = \binom{n-1}{m-1}. \tag{4}$$

Expected probability of sampling the subsets. Assuming that the samples are drawn independently, the mean measure of a sample of m elements drawn from the population with distribution μ_X is:

$$\mathbb{E}_X\left[\widehat{d\mu}_X(X)\right] = \mathbb{E}_X\left[\prod_{i=1}^{m} \widehat{d\mu}_X(x_i)\right] = \prod_{i=1}^{m} \mathbb{E}_X\left[\widehat{d\mu}_X(x_i)\right], \tag{5}$$

where $\widehat{\mu}_X$ is estimated using a kernel density estimator. Note, we can only draw samples consisting of examples in X_D. Finally, the $\overline{e_m(x)}$ for any example x is derived as:

$$\mathbb{P}(x_{j+1} \in X_k) \times \binom{n-1}{m-1} \times \prod_{i=1}^{m} \mathbb{E}_X\left[\widehat{d\mu}_X(x_i)\right].$$

4.2 Propensity Scores under Imperfect Oracles

In the real world, the user is an imperfect oracle. If a truly positive example is selected to be labeled, she may be unsure of its label and decide not to label it. If a truly negative example is selected to be labeled, there is a small probability she labels it incorrectly as a positive. This requires changing the label probability to include the probability that the user is able to label the example:[3]

$$\mu(s=1|x,X) = \mu(q=1|x,X)\mu(s=1|x,X,q=1),$$

where q is the binary variable that is 1 if x is queried and 0 otherwise. Note that the previous identity is obtained because the probability to label an example not queried is 0.

Query probability. The probability to query an example depends on whether or not the example is in the top-k of a subset X according to the active learning strategy:

$$\begin{aligned} &\mu(q=1|x,X) = \mu(q=1|X, x \in X_k) \cdot \mu(x \in X_k|X) \\ &+ \mu(q=1|X, x \notin X_k) \cdot \mu(x \notin X_k|X) = \\ &= \mu(x \in X_k|X) + \mu(q=1|X, x \notin X_k) \cdot \mu(x \notin X_k|X). \end{aligned}$$

First, if an example is in the top-k of X, it is always queried. Second, if an example is not in the top-k, whether it is queried now depends on the user's uncertainty about the labels of the higher ranked examples in X. Let x_{j+1} be $j+1$-th example in the general ranking. To compute the probability that x_{j+1} is queried, we first simplify the problem by approximating the user's uncertainty about any example x with the mean of the user's uncertainty over all the examples in the observed dataset X_D. Then, the probability can be computed as:

$$\mu(q=1|X, x_{j+1} \notin X_k) = \sum_{t=\max\{k,m-n+j\}}^{\min\{j,m-1\}} \binom{j}{t}\binom{n-j-1}{m-t-1} \sum_{s=t-k+1}^{t} (1-\overline{p})^s \overline{p}^{\,t-s-1} \tag{6}$$

where t is the number of examples ranked higher than x_{j+1} in a given subset X, s is the number of examples out of t that the user cannot label, and $\bar{p}$ is the user's uncertainty for any example. Intuitively, Equation 6 considers all possible scenarios where the user fails to label enough examples such that x_{j+1} is queried and sum the probabilities of these scenarios.

[2] If a truly negative example is selected, no label is given.

[3] For brevity, we omit $|y=1$ everywhere in this section, even though all the events in the equations are conditioned on $y=1$.

Label probability under user's uncertainty. Finally, the label probability of an example x in position $j+1$ is:

$$\begin{aligned}\mu(s=1|x,X) &= \mu(s=1|x,X,q=1)\cdot\mu(q=1|x,X) = \\ &= \mu(s=1|x,X,q=1)\cdot[\mu(x\in X_k|X)+ \\ &+ \mu(q=1|X,x\notin X_k)\cdot\mu(x\notin X_k|X)],\end{aligned} \tag{7}$$

where $\mu(s=1|x,X,q=1)$ is the user uncertainty of that example, $\mu(x\in X_k|X)$ is as in 3, $\mu(q=1|X,x\notin X_k)$ is as in 6 and $\mu(x\notin X_k|X)$ is $1-\mu(x\in X_k|X)$. The final propensity score for an example x under user uncertainty is the product between the factors in Equations 4, 5, and 7.

5 Experiments

We empirically evaluate the effectiveness of CAPE to recover the true class prior in the context of anomaly detection because it matches our setting: a handful of *normal* (positive) labels are acquired through an active learning strategy, the remaining examples are unlabeled. Furthermore, most anomaly detection algorithms require an estimate of the class prior to make binary predictions (an example is normal or abnormal).[4] We address the following empirical questions:

1. Can CAPE accurately estimate the true class prior?
2. How does user uncertainty affect the CAPE's ability to estimate the class prior?
3. Does a more accurate estimate of the class prior improve the performance of an anomaly detector?

5.1 Experimental Setup

Methods. We compare: our proposed method CAPE[5], TICE which estimates the class prior using decision tree induction [Bekker and Davis, 2018], and KM1 and KM2 which compute the class prior by modeling the distribution of the positive examples [Ramaswamy *et al.*, 2016].

Data. The benchmark consists of 9 standard anomaly detection datasets from [Campos *et al.*, 2016].[6] The datasets are listed in Table 1. They contain more normals than anomalies with normal class priors varying between 0.64 and 0.99.

Setup. In all experiments, SSDO with its default parameters is used as the semi-supervised anomaly detector [Vercruyssen *et al.*, 2018].[7] SSDO learns a classifier from unlabeled and normal examples. We use ISOLATION FOREST [Liu *et al.*, 2008] as its unsupervised prior. Using the method from [Kriegel *et al.*, 2011], the anomaly scores are mapped to probabilities. We use *uncertainty sampling* as active learning strategy [Settles, 2012]. We model the user's uncertainty using the the kernel density estimate as implemented in SCIKIT-LEARN. For each dataset and compared method, the following procedure is repeated five times. First, the dataset is split into training and test sets using a stratified 5-fold split. All training data are initially unlabeled. Then iteratively, the user is queried until $k=5$ new labels are added to the training set in accordance with the active learning strategy and the probability of labeling the example correctly is equal to the user's uncertainty. After adding new labels to the training data, the class prior is estimated on the training data, the SSDO classifier is retrained, and its performance on the test set is measured (using the estimated class prior to obtain binary predictions for the test data). The process stops when 150 examples are labeled. We report the results averaged over all five runs.

Dataset	# Examples (s)	# Vars	$\mathbb{P}(y=1)$
WBC	454	9	0.9780
Shuttle	507	9	0.9862
WDBC	367	30	0.9728
Stamps	340	9	0.9088
Ionosphere	351	32	0.6410
Cardiotocography	434	21	0.9493
PageBlocks	421	10	0.8979
Pima	625	8	0.8000
Annthyroid	713	21	0.9257

Table 1: Benchmark anomaly detection datasets.

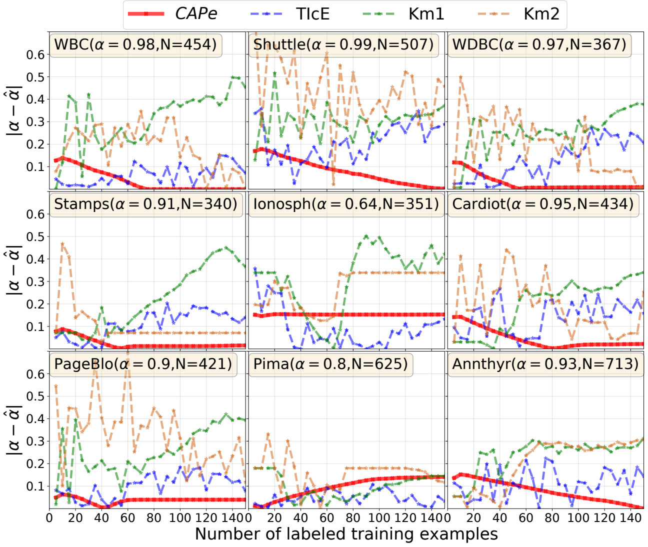

Figure 1: MAE of class prior estimates as a function of the number of labels, under no user uncertainty. Lower numbers are better.

Hyperparameters. The parameters of TICE, KM1, and KM2 are set to the values recommended in the original papers. CAPE has only one hyperparameter: the range of cardinalities m in the outer loop, which is minimally 1 and maximally n (the cardinality of the dataset). In the experiments, we set the range to $n\cdot\{0.02, 0.04, 0.06, \ldots, 0.4, 0.5, \ldots 0.9\}$.

5.2 Results

Q1: Recovering the class prior. Figure 1 shows the mean absolute error (*MAE*) of the estimated class prior as a function of the number of labeled training examples with no user uncertainty. On seven of the nine datasets, CAPE outperforms

[4] The anomaly detection algorithms implemented in SCIKIT-LEARN or PYOD require the *contamination factor* (1− class prior) to be able to make binary predictions.

[5] Code: https://github.com/Lorenzo-Perini/Active_PU_Learning

[6] Data: www.dbs.ifi.lmu.de/research/outlier-evaluation

[7] Code: https://github.com/Vincent-Vercruyssen/anomatools

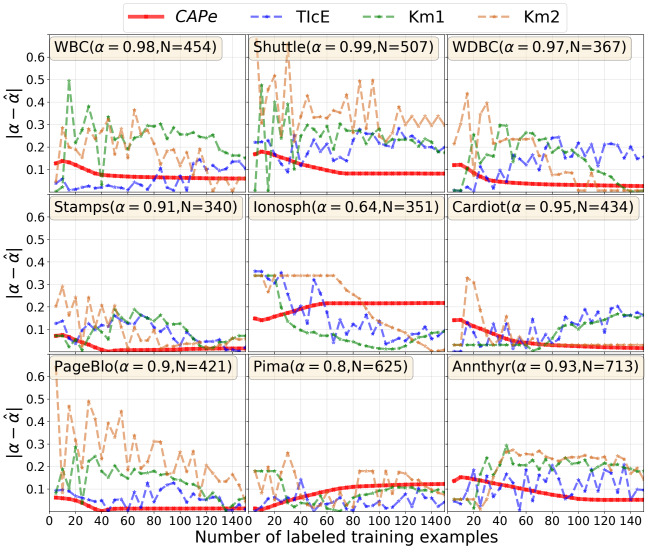

Figure 2: MAE of class prior estimates as a function of the number of labels, assuming the user's uncertainty. Lower numbers are better.

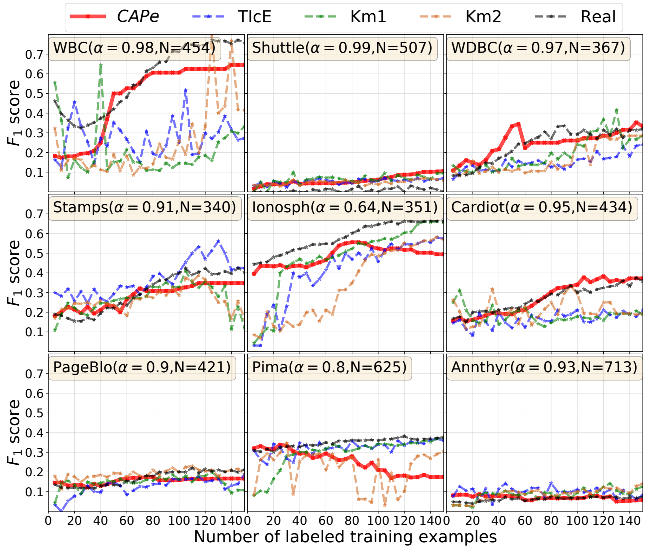

Figure 3: The F_1 score when using each approach's estimated class prior to threshold SSDO's numeric output into a decision rule.

the three baselines. On these datasets our estimate converges to the correct one, often with < 100 labels. On two datasets, TICE results in (slightly) better performance than CAPE. In these two datasets, our approach tends to overestimate the class prior, likely due to inaccuracies in the underlying SSDO model (i.e., anomalous examples have a high predicted probability of being normal). In addition, as more examples are labeled, CAPE's estimate of the class prior converges to the true class prior smoothly. In contrast, acquiring a small number of labels (e.g., 5) may cause its competitors' estimates of the class prior to change dramatically.

Q2: impact of user uncertainty. We repeat the previous experiment in the more realistic setting where the user is uncertain and makes mistakes in the labeling. Figure 2 shows how the mean absolute error (*MAE*) of the estimated class prior varies as a function of the number of labeled instances for each method. Again, CAPE results in the most accurate estimates on seven of the nine datasets. Again, its estimates fluctuate less than its competitors.

Q3: class prior impact on anomaly detection. Most anomaly detectors require knowing the proportion of anomalies in the dataset either for training the detector itself or for thresholding the detector's numeric outlier scores in order to be able to make a decision in practice. In this experiment, we consider the second scenario and use the estimated class prior to convert SSDO's numeric output into a decision rule. Figure 3 shows the F1 score for SSDO's model when using the class prior estimated by each method to set the decision threshold. Here, the results are more mixed as CAPE yields equivalent or more accurate estimates on a small majority of the cases. Note that the black dashed line represents performance when using the true class prior: using the true and estimated priors result in similar predictive performance.

6 Related Work

Several papers have studied the combined setting of PU learning and active learning. However, while we focus on estimating the class prior, these papers have a different focus. [Ghasemi *et al.*, 2016] designed an uncertainty sampling active learning strategy specifically tailored to PU datasets. [Barnabé-Lortie *et al.*, 2015] developed an active learning strategy for one-class classification by querying the examples which match the learned class the least. There are a number of different ways to apply active learning strategies when dealing with one-class classification problems [Trittenbach *et al.*, 2018; Trittenbach *et al.*, 2019]. Finally, [He *et al.*, 2015] applied active learning to PU time series data by querying the examples with both high uncertainty and high utility.

7 Conclusion

We proposed a CAPE, a method that estimates the class prior in a PU setting where the positive labels were acquired through active learning. CAPE derives the class prior by first estimating each unlabeled example's propensity score, which is the probability that the active learning approach will query the example's label. Theoretically, we proved that our estimate of the class prior will converge to its true value if we obtain accurate propensity scores. Practically, we showed how to estimate the propensity scores in two settings. In the first, the user never makes mistakes and only labels positive examples whereas the second considers modeling the user's uncertainty. Empirically, we demonstrated that CAPE recovers the class prior more accurately than existing approaches.

Acknowledgements

This work is supported by FWO-Vlaanderen grant G0D8819N (JD) and the Flemish government under the "*Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen*" programme (JD, LP, VV).

References

[Barnabé-Lortie *et al.*, 2015] Vincent Barnabé-Lortie, Colin Bellinger, and Nathalie Japkowicz. Active learning for one-class classification. In *14th International Conference on Machine Learning and Applications*, pages 390–395. IEEE, 2015.

[Bekker and Davis, 2018] Jessa Bekker and Jesse Davis. Estimating the class prior in positive and unlabeled data through decision tree induction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[Bekker and Davis, 2020] Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: A survey. *Machine Learning*, 2020.

[Bekker *et al.*, 2019] Jessa Bekker, Pieter Robberechts, and Jesse Davis. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *Proceedings of 2018 Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2019.

[Campos *et al.*, 2016] Guilherme O Campos, Arthur Zimek, Jörg Sander, Ricardo JGB Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E Houle. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, 30(4):891–927, 2016.

[Du Plessis and Sugiyama, 2014] Marthinus Christoffel Du Plessis and Masashi Sugiyama. Class prior estimation from positive and unlabeled data. *IEICE Transactions on Information and Systems*, 97(5):1358–1362, 2014.

[du Plessis *et al.*, 2015] M.C. du Plessis, G. Niu, and M. Sugiyama. Class-prior estimation for learning from positive and unlabeled data. *Proceedings of the 7th Asian Conference on Machine Learning*, pages 221–236, 2015.

[Elkan and Noto, 2008] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 213–220, 2008.

[Ghasemi *et al.*, 2016] Alireza Ghasemi, Hamid R. Rabiee, Mohsen Fadaee, Mohammad Taghi Manzuri, and Mohammad H. Rohban. Active learning from positive and unlabeled data. *CoRR*, abs/1602.07495, 2016.

[He *et al.*, 2015] Guoliang He, Yong Duan, Yifei Li, Tieyun Qian, Jinrong He, and Xiangyang Jia. Active learning for multivariate time series classification with positive unlabeled data. In *Proceedings of the 27th International Conference on Tools with Artificial Intelligence*, pages 178–185, 2015.

[Jain *et al.*, 2016a] Shantanu Jain, Martha White, and Predrag Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. In *Advances in Neural Information Processing Systems*, pages 2693–2701, 2016.

[Jain *et al.*, 2016b] Shantanu Jain, Martha White, Michael W Trosset, and Predrag Radivojac. Nonparametric semi-supervised learning of class proportions. *arXiv preprint arXiv:1601.01944*, 2016.

[Kriegel *et al.*, 2011] Hans-Peter Kriegel, Peer Kroger, Erich Schubert, and Arthur Zimek. Interpreting and unifying outlier scores. In *Proceedings of the SIAM International Conference on Data Mining*, pages 13–24, 2011.

[Liu *et al.*, 2008] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Eighth IEEE International Conference on Data Mining*, pages 413–422. IEEE, 2008.

[Ramaswamy *et al.*, 2016] Harish Ramaswamy, Clayton Scott, and Ambuj Tewari. Mixture proportion estimation via kernel embeddings of distributions. In *International Conference on Machine Learning*, pages 2052–2060, 2016.

[Settles, 2012] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

[Trittenbach *et al.*, 2018] Holger Trittenbach, Adrian Englhardt, and Klemens Böhm. An overview and a benchmark of active learning for outlier detection with one-class classifiers. *arXiv preprint arXiv:1808.04759*, 2018.

[Trittenbach *et al.*, 2019] Holger Trittenbach, Adrian Englhardt, and Klemens Böhm. Validating one-class active learning with user studies–a prototype and open challenges. In *ECML PKDD Workshop*, page 17, 2019.

[Vercruyssen *et al.*, 2018] Vincent Vercruyssen, Meert Wannes, Verbruggen Gust, Maes Koen, Bäumer Ruben, and Davis Jesse. Semi-supervised anomaly detection with an application to water analytics. In *Proceedings of the IEEE International Conference on Data Mining*, 2018.