Learning from Positive and Unlabeled Data without Explicit Estimation of Class Prior

Chenguang Zhang,[1,2] **Yuexian Hou,**[1*] **Yan Zhang**[2]
[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]School of Science, Hainan University, Haikou, China
{chenguang_zhang, yxhou}@tju.edu.cn, zhangyanouc@sina.com

Abstract

Learning a classifier from positive and unlabeled data may occur in various applications. It differs from the standard classification problems by the absence of labeled negative examples in the training set. So far, two main strategies have typically been used for this issue: the likely negative examples-based strategy and the class prior-based strategy, in which the likely negative examples or the class prior is required to be obtained in a preprocessing step. In this paper, a new strategy based on the Bhattacharyya coefficient is put forward, which formalizes this learning problem as an optimization problem and does not need a preprocessing step. We first show that with the given positive class conditional probability density function (PDF) and the mixture PDF of both the positive class and the negative class, the class prior can be estimated by minimizing the Bhattacharyya coefficient of the positive class with respect to the negative class. We then show how to use this result in an implicit mixture model of restricted Boltzmann machines to estimate the positive class conditional PDF and the negative class conditional PDF directly to obtain a classifier without the explicit estimation of the class prior. Many experiments on real and synthetic datasets illustrated the superiority of the proposed approach.

Learning a classification from positive and unlabeled data (LPU) is a task of great importance since it has many practical applications, such as land-cover classification and document retrieval, where positive samples, i.e., labeled objects of interest, and unlabeled samples are readily available, but negative samples, i.e., the objects we are not interested in, are too diverse to be labeled. Generally, traditional classification methods are usually inapplicable to LPU problems, as they assume the availability of explicit negative samples. In this paper, we focus on using both labeled positive samples and unlabeled samples to build binary classifiers.

The research on LPU problems dates back to at least the work of Denis (1998), in which he proved that LPU is possible as soon as the weight of the target concept is known by the learner. Liu et al. provided a learning rule for LPU and analyzed its sample complexity bounds for VC classes (Liu et al. 2002). Despite the diversity of these analyses, one common assumption of LPU methods is that unlabeled data can help to extract extra information to compensate for the lack of labeled negative data. Under this assumption, two general approaches have been proposed: the likely negative examples-based strategy and the class prior-based strategy.

The former transforms LPU problems to standard classification problems by using heuristics to identify a set of likely negative examples from unlabeled data. Papers using this idea include (Yu, Han, and Chang 2004; Li and Liu 2003; Wang et al. 2006; Yu 2005). The disadvantage of this approach is the difficulty in deciding the size of the extracted negative set, which may make the final classifier tend to overfit or underfit, particularly when there is a significant overlap between the classes. The latter uses the estimated class information, namely, the class prior, as weights to train a classifier on the positive data and the unlabeled dataset directly (Liu et al. 2003; Elkan and Noto 2008; du Plessis, Niu, and Sugiyama 2014; Kiryo et al. 2017; du Plessis, Niu, and Sugiyama 2015). These methods have been reported to be better than the likely negative examples-based methods. However, the two-step learning strategy may make the classification performance heavily rely on the estimation of the class prior, which is biased due to the overlap between the classes and is quite possibly distorted when the set of labeled positive samples is small. Some effective class prior estimation methods can be found in the papers (Elkan and Noto 2008; du Plessis and Sugiyama 2014; Jain, White, and Radivojac 2016; du Plessis, Niu, and Sugiyama 2017; Bekker and Jesse 2018).

In this paper, we propose a Bhattacharyya coefficient-based new strategy for LPU problems. This strategy is based on the observation that the negative samples should not or seldom appear in the places where the positive samples often appear and vice versa, which implies that the negative class conditional PDF should have a small overlap with the positive class conditional PDF. To formalize this observation, a distance measure between the distributions is needed. Compared with other measures such as the KL divergence, the Bhattacharyya coefficient, known as an intuitive and direct description of the overlap, is a bounded measure, which equals to zero if there is no overlap between the two distributions (Comaniciu, Ramesh, and Meer 2003;

*Corresponding author.

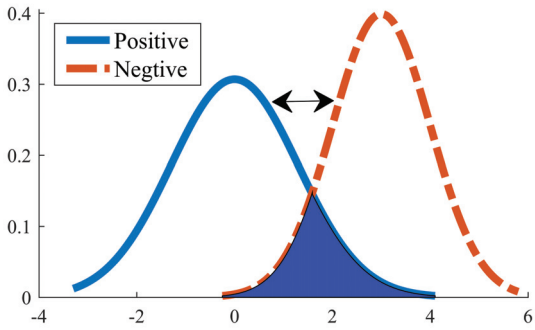


Figure 1: An illustration of the minimization of the overlap. Minimizing the overlap (the shaded area) between the model positive class conditional PDF (the left line) and the model negative class conditional PDF (the right line) makes them far away from each other.

Fukunaga 2013); moreover, the Bayesian error probability is up bounded by the Bhattacharyya coefficient, then the minimization of which may lead to the decrease of the Bayesian error (Ray 1989). Actually, the unbounded divergences such as KL divergence are more suitable for the similarity maximization problems.

We prove that the class prior can be estimated by minimizing the Bhattacharyya coefficient between the positive class and the negative class if the positive class conditional PDF and the mixture PDF of both classes are given. Next, rather than determine the class prior, we consider determining the negative class conditional PDF directly. To this end, by using the implicit mixture model of restricted Boltzmann machines (RBMs) (Nair and Hinton 2009) as the underlying learning model, the LPU problem is formalized as an optimization problem, where the Bhattacharyya coefficient between the classes is set as the optimization objective; moreover, the model PDFs are required to fit the positive samples and the unlabeled samples simultaneously. Solving the optimization problem naturally gives the estimations of the positive class conditional PDF and the negative class conditional PDF, which are guaranteed to be as far away from each other as possible due to the learning goal (see Fig. 1).

Compared to the existing methods, the proposed strategy has the following traits: 1) It is a new learning strategy for the LPU problem by formulating LPU as an optimization problem without the preprocessing step to extract the likely negative samples or to estimate the class prior. 2) The model's mixture PDF of the positive class and the negative class is required to fit the unlabeled examples. This feature enables the proposed method to learn with a small set of labeled positive samples.

Preliminaries

In this section, we briefly review the definitions and properties of the Bhattacharyya coefficient and the implicit mixture model of RBMs (IRBM), which will be used later as the optimization goal and the data descriptor.

Bhattacharyya coefficient

The Bhattacharyya coefficient between two probability densities $p_1(\mathbf{v})$ and $p_2(\mathbf{v})$, with $\mathbf{v} \in \mathbb{R}^d$, is defined as

$$B = \int_{\mathbb{R}^d} \sqrt{p_1(\mathbf{v})p_2(\mathbf{v})}\mathrm{d}\mathbf{v}. \tag{1}$$

Obviously, the values of B are always confined within the $[0, 1]$ interval.

IRBM

Here, IRBM is used as a mixture model of two RBMs (denoted by a positive RBM and a negative RBM) with the mixed weights implicitly parameterized. A binary indicator $\mathbf{q} = [q_1, q_2]$ was introduced additionally, where $q_1 = 1$ ($q_2 = 1$) represents that the positive (negative) RBM is activated. Let $\mathbf{v} \in \mathbb{R}^d$ be a vector of visible (observed) variables and $\mathbf{h}$ be a vector of hidden variables. The energy function of IRBM is

$$E(\mathbf{v}, \mathbf{h}, \mathbf{q}) = \frac{1}{2}\sum_i (v_i - c_i)^2 - \sum_j h_j d_j - \sum_k q_k \sum_{i,j} W_{ijk} v_i h_j, \tag{2}$$

where the Gaussian-binary RBM is adopted; $c_i = \sum_k q_k C_{ik}$, $d_j = \sum_k q_k D_{jk}$; W_{ijk},C_{ik} and D_{jk} are model parameters needed to be trained. The joint distribution defined by the mixture model is

$$p(\mathbf{v}, \mathbf{h}, \mathbf{q}) = \exp\left(-E(\mathbf{v}, \mathbf{h}, \mathbf{q})\right)/Z, \tag{3}$$

where $Z = \Sigma_{\mathbf{v},\mathbf{h},\mathbf{q}} \exp\left(-E(\mathbf{v}, \mathbf{h}, \mathbf{q})\right)$ is the partition function. Let $q_k = 1 (k = 1, 2)$; by Eq. (3), the PDF defined by the k^{th} component can also be obtained.

Let θ be the collection of model parameters needed to be trained. Similar to the RBM, training the IRBM is essentially minimizing the Kullback–Leibler divergence (KL) between the empirical data distribution $p_{data}(\mathbf{v})$ and the model distribution $p(\mathbf{v}; \theta)$, which can be fulfilled by the contrastive divergence algorithm (CD): 1) sampling latent variables from $p(\mathbf{h}, \mathbf{q}|\mathbf{v})$ and 2) reconstructing data from $p(\mathbf{v}|\mathbf{h}, \mathbf{q})$. The reconstructing process can be simply performed on the RBM appointed by $\mathbf{q}$. The sampling process includes two steps and is slightly different from that of RBMs. The first step is to sample $\mathbf{q}$ from $p(\mathbf{q}|\mathbf{v})$. The second step is to sample $\mathbf{h}$ from the conditional distribution $p(\mathbf{h}|\mathbf{v})$ on the selected RBM determined by $\mathbf{q}$. $p(\mathbf{q}|\mathbf{v})$ is given by

$$p\left(q_k = 1|\mathbf{v}\right) = \frac{\exp\left(-F(\mathbf{v}, q_k = 1)\right)}{\sum_m \exp\left(-F(\mathbf{v}, q_m = 1)\right)}, \tag{4}$$

where

$$F(\mathbf{v}, q_k = 1) = \frac{1}{2}\sum_i (v_i - c_i)^2 - \sum_j \log\left(1 + \exp\left(\sum_i W_{ijk} v_i\right)\right). \tag{5}$$

The Proposed Strategy

Theoretical Motivation

Let $\mathcal{Y} = \{+1, -1\}$ be the set of possible labels. Without loss of generality, we suppose only the first l cases in $\left\{\mathbf{v}^1, ..., \mathbf{v}^N\right\}$ are labeled with the positive label +1 and the rest are unlabeled. Let $P = \left\{\mathbf{v}^1, ..., \mathbf{v}^l\right\}$ be the set of labeled positive samples, and $U = \left\{\mathbf{v}^{l+1}, ..., \mathbf{v}^N\right\}$ be the set of unlabeled samples. The goal of our approach is to learn a function that is equal to $p(q_1 = 1|\mathbf{v})$ as closely as possible. From the Bayesian rule, we have

$$p(q_1 = 1|\mathbf{v}) = \frac{p(\mathbf{v}|q_1 = 1)p(q_1 = 1)}{p(\mathbf{v})}. \tag{6}$$

Suppose $p(\mathbf{v})$, $p(\mathbf{v}|q_1 = 1)$ can be learned from the *P* set and the *U* set; then, given the fact that $p(\mathbf{v}) = p(\mathbf{v}|q_1 = 1)p(q_1 = 1) + p(\mathbf{v}|q_2 = 1)\left(1 - p(q_1 = 1)\right)$, all that is left is to estimate $p(\mathbf{v}|q_2 = 1)$ or equivalently to estimate $p(q_1 = 1)$. However, because of the lack of negative samples, their estimation is not straightforward and easy. In this paper, we follow the intuitive idea that the negative samples should not appear in the places where the positive samples often appear; i.e., $p(\mathbf{v}|q_2 = 1)$ should be located as far away from $p(\mathbf{v}|q_1 = 1)$ as possible, which leads to the following theorem. This theorem, which shows how to obtain an adequate estimation of $p(q_1 = 1)$, in the absence of negative traning samples, is our central result.

Theorem 1. *Given $p(\boldsymbol{v})$ and $p(\boldsymbol{v}|q_1 = 1)$ with $p(\boldsymbol{v}|q_2 = 1)$ and $p(q_1 = 1)$ unknown, a positive-biased estimator of the class prior $p(q_1 = 1)$, denoted by $\hat{\alpha}$, can be achieved at the minimum point by minimizing the Bhattacharyya coefficient between $p(\boldsymbol{v}|q_1 = 1)$ and $p(\boldsymbol{v}|q_2 = 1)$ with respect to the unknown class prior $p(q_1 = 1)$. This estimator has a upper bound, i.e.,*

$$\hat{\alpha} \leq \left(\int_{\mathbb{R}^d} \frac{p^2(\boldsymbol{v}|q_1 = 1)}{p(\boldsymbol{v})} dv\right)^{-1}. \tag{7}$$

Furthermore, $\hat{\alpha}$ equals $p(q_1 = 1)$ if the intersection of $Supp\left\{p(\boldsymbol{v}|q_2 = 1)\right\}$ and $\left(Supp\left\{p(\boldsymbol{v}|q_1 = 1)\right\}\right)^c$ is not a null set, where $Supp\left\{\cdot\right\}$ is the support set of a PDF and $(A)^c$ is the complement of set A.

The proof is given at the end of the paper .

It is worth noting that the upper bound of $\hat{\alpha}$ is precisely the positive biased estimator of the class prior proposed in (du Plessis and Sugiyama 2014), which has been proved to be unbiased when the class conditional PDFs are completely nonoverlapping.

A possible approach using the theorem for LPU is to first estimate $p(\mathbf{v})$ and $p(\mathbf{v}|q_1 = 1)$ to obtain the class prior $\hat{\alpha}$. This, however, does not work well since the theorem requires an accurate estimation of $p(\mathbf{v})$ and $p(\mathbf{v}|q_1 = 1)$ at all possible points; the division by $p(\mathbf{v}|q_1 = 1)$ in the estimation formula of $\hat{\alpha}$ may exacerbate the estimation error.

Instead of estimating the class prior, we consider using the theorem to develop a new strategy for LPU problems to obtain the negative class conditional PDF directly. Let $p(\mathbf{v}|q_1 = 1; \theta_1)$ and $p(\mathbf{v}|q_2 = 1; \theta_2)$ be the model positive and negative PDF respectively, where θ_1 and θ_2 are model parameters. By setting the minimization of the Bhattacharyya coefficient between the positive and negative PDFs as the learning goal and requiring the model mixture PDF $p(\mathbf{v}; \theta_1, \theta_2)$ and the positive class PDF $p(\mathbf{v}|q_1 = 1; \theta_1)$ fit the samples over the U set and the samples over the P set, respectively, this strategy can be formalized as

$$\begin{aligned} &\min_{\theta} \left\{ B(\theta) = \int_{\mathbb{R}^d} \sqrt{p(\mathbf{v}|q_1 = 1; \theta_1) p(\mathbf{v}|q_2 = 1; \theta_2)} \mathrm{d}\mathbf{v} \right\} \\ &s.t. \;\; D\left(p_{_{data}}(\mathbf{v}|q_1 = 1), p(\mathbf{v}|q_1 = 1; \theta_1)\right) = 0 \\ &\qquad D\left(p_{_{data}}(\mathbf{v}), p(\mathbf{v}; \theta_1, \theta_2)\right) = 0, \end{aligned} \tag{8}$$

where $\theta = \{\theta_1, \theta_2\}$, D is a divergence measure between distributions with a zero value when the input distributions are identical, and $p_{_{data}}$ represents the corresponding empirical data PDF. According to Theorem 1, when the constrained optimization reaches a minimum, an adequate estimation of the negative as well as the positive class conditional PDF associated with $\hat{\alpha}$ can be obtained.

A Practical Approach

We then aim to use the proposed strategy to create a practical approach. There are two obstacles to this approach. First, to formalize the constrained optimization problem (8), a data descriptor is needed to describe the multi-PDFs; second, the optimization problem (8) with the constraints imposed on PDFs is not a standard optimization problem. For the first issue, IRBM is used as the data descriptor since IRBM captures the empirical data PDF without the class prior explicitly parameterized. The data descriptor can also be any other model that is a realization of multi-statistical hypothesis, such as generative neural networks, probabilistic graphical models and so on. However, additional manipulations are needed on the class prior if it is explicitly parameterized. For the second issue, we solve it by introducing the Heaviside function $H(z)$ and one-dimensional Dirac measure $\delta_0(z)$, thereby transforming the constrained optimization into an unconstrained optimization problem, which is

$$\begin{aligned} \min_{\theta} \{ &B(\theta) + H\left(K\left(p_{_{data}}(\mathbf{v}), p(\mathbf{v}; \theta)\right)\right) \\ &+ H\left(K\left(p_{_{data}}(\mathbf{v}|q_1 = 1), p(\mathbf{v}|q_1 = 1; \theta_1)\right)\right)\}, \end{aligned} \tag{9}$$

where $\theta = \{\theta_1, \theta_2\}$ is the set of parameters of the IRBM, respectively; $K(\cdot)$ is the KL divergence, and

$$H(z) = \begin{cases} 1, & if\ z > 0 \\ 0, & if\ z \leq 0, \end{cases} \qquad \delta_0(z) = \frac{\mathrm{d}}{\mathrm{dz}} H(z). \tag{10}$$

where $z \in \mathbb{R}$. As $0 \leq B(\theta) \leq 1$, except that D is externalized as KL divergence, the above optimization is equivalent to the initial problem (8) if the KL divergence between the empirical data distributions and the corresponding estimated distributions can be 0. However, due to the limited presentation capacity of two-layer RBMs, the KL divergence in most cases is usually not exactly equal to 0, which would make the values of the two H terms always equal to 1. In addition, the nondifferentiability of functions $H(z)$ and δ_0 at the point 0 also introduces difficulties in solving the optimization. For these two reasons, we consider replacing the

hard H and δ_0 with soft H_ε and δ_ε to obtain an soft version of the optimization problem. Then we obtain

$$\begin{aligned}\min_{\theta} \{&B(\theta) + H_\varepsilon\left(K\left(p_{_{data}}(\mathbf{v}), p(\mathbf{v};\theta)\right)\right) \\ &+H_\varepsilon\left(K\left(p_{_{data}}(\mathbf{v}|q_1=1), p(\mathbf{v}|q_1=1;\theta_1)\right)\right)\},\end{aligned} \tag{11}$$

where we follow the work of (Chan and Vese 2001) and define

$$H_\varepsilon(z) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{z}{\varepsilon}\right)\right), \delta_\varepsilon = \frac{\mathrm{d}H_\varepsilon(z)}{\mathrm{d}z}, \tag{12}$$

and ε is a small positive real number.

Similar to the training process of IRBM, the gradient descent method is employed to address the above optimization problem. For the sake of brevity, the three terms of Eq. (11) are denoted by $B(\theta)$, $H_1(\theta)$, and $H_2(\theta_1)$; the KL divergences in $H_1(\theta)$ and $H_2(\theta_1)$ are denoted by $K_1(\theta)$ and $K_2(\theta_1)$, respectively. Given the samples $\mathbf{v}^s \in U$, the estimation value of $B(\theta)$ is

$$B(\theta) = \sum_{\mathbf{v}^s} \sqrt{p(\mathbf{v}^s|q_1=1;\theta_1)p(\mathbf{v}^s|q_2=1;\theta_2)}. \tag{13}$$

The derivative of $B(\theta)$ with respect to θ_k is

$$\begin{aligned}\frac{\partial B}{\partial \theta_k} = &\left[\sum_{\mathbf{v}^s} p(\mathbf{v}^s)\sqrt{p(q_1=1|\mathbf{v}^s)p(q_2=1|\mathbf{v}^s)}\frac{\partial F_k(\mathbf{v}^s)}{\partial \theta_k}\right. \\ &- \left(\frac{\sum_{\mathbf{v}^s} p(\mathbf{v}^s)\sqrt{p(q_1=1|\mathbf{v}^s)p(q_2=1|\mathbf{v}^s)}}{\sum_{\mathbf{v}} p(\mathbf{v})p(q_k=1|\mathbf{v})}\right) \\ &\left.\left(\sum_{\mathbf{v}} p(\mathbf{v})p(q_k=1|\mathbf{v})\frac{\partial F_k(\mathbf{v})}{\partial \theta_k}\right)\right]\frac{-1}{2\sqrt{p(q_1=1)p(q_2=1)}},\end{aligned} \tag{14}$$

where $F_k(\mathbf{v}^s) = F_k(\mathbf{v}^s, q_k=1)$ (see Eq.(5) and see (Taylor et al. 2010)) for the details of the derivative computation of F_k with respect to the model parameters). Obviously, computing the terms associated with the variable $\mathbf{v}$ in Eq. (14) exactly requires summing over the joint space of all possible visible variables, which is intractable actually. Here, we circumvent this problem by using the CD learning algorithm (Hinton 2002). Specially, we sample $\mathbf{v}$ from the mixture PDF to obtain the value of the corresponding expectation terms, then to obtain the estimated value of the derivative of $B(\theta)$.

The derivative of $H_1(\theta)$ with respect to θ is

$$\frac{\partial H_1}{\partial \theta} = \delta_\varepsilon\left(K_1(\theta)\right)\frac{\partial K_1(\theta)}{\partial \theta}. \tag{15}$$

The second derivative term on the right side of the equation, given $\mathbf{v}^s \in U$, can be computed by the CD algorithm as presented in the preliminaries. However, the calculation of the first term is not ready-made due to the time and difficulty of estimating $K_1(\theta)$. An alternative way is replacing the KL divergence by the reconstruction error, which is based on the following considerations. Note the δ_ε term serves as a guide in Eq. (15), e.g., it would be bigger if the current model PDFs fit the data PDFs better. Since the KL divergence and the reconstruction error have some degree of consistence, i.e., a smaller KL divergence usually leads to a smaller reconstruction error, this replacing preserves the effect of the the δ_ε term and simplifies the calculation process. Indeed, as we are interested in the minimizing of the optimization problem and not concerned with its precise value, the reconstruction error may offer favorable trade-offs. Specially, we use the average reconstruction error (formalized as the L2 norm) over the unlabeled set U as a substitution of the divergence $K_1(\theta)$, which is defined as

$$\mathcal{R}_1(\theta) = \frac{1}{|U|}\sum_{\mathbf{v}^s \in U}\|\mathbf{v}^s - \mathbf{v}^r\| \tag{16}$$

where $|U|$ is the size of U, and $\mathbf{v}^r$ is the reconstructed sample of $\mathbf{v}^s$ by IRBM.

Similarly, we have the derivative of $H_2(\theta_1)$ with respect to θ_1 is

$$\frac{\partial H_2}{\partial \theta_1} = \delta_\varepsilon\left(K_2(\theta_1)\right)\frac{\partial K_2(\theta_1)}{\partial \theta_1}, \tag{17}$$

where the derivative term can be computed by the CD algorithm over the labeled set P since the derivative is actually only related to the positive RBM. The δ_ε term is substituted by $\delta_\varepsilon(\mathcal{R}_2(\theta_1))$, where

$$\mathcal{R}_2(\theta_1) = \frac{1}{|P|}\sum_{\mathbf{v}^s \in P}\|\mathbf{v}^s - \mathbf{v}^r\|, \tag{18}$$

where $\mathbf{v}^s \in P$ and $\mathbf{v}^r$ is the reconstructed sample of $\mathbf{v}^s$ by the positive RBM.

After the computation of the derivative, the parameters of our model are iteratively updated as shown below:

$$\theta_{new} = \theta_{old} - \eta\Delta\theta, \tag{19}$$

where η is the learning rate and

$$\Delta\theta = \frac{\partial(B + H_1 + H_2)}{\partial \theta} \tag{20}$$

Finally, for any given sample $\mathbf{v}$, following Bayes decision theory, if $p(q_1=1|\mathbf{v}) > p(q_2=1|\mathbf{v})$, its label is positive. Otherwise, its label is negative. The $p(q_1=1|\mathbf{v})$ can be computed by Eq. (4). Notice that the computation of Eq. (20) is just applying the CD algorithm on the *P* set and on the *U* set respectively, so the asymptotic time complexity of the proposed method is the same as IRBM.

Experiments

In this section, we experimentally evaluate the performance of the proposed method on both the real benchmark datasets and the artificial datasets by comparing it with the cost-sensitive LPU method (CSLPU) (du Plessis, Niu, and Sugiyama 2014), the decision tree based class prior estimation method (TIcE) (Bekker and Jesse 2018), the LPU method with non-negative risk estimator (nnPU) (Kiryo et al. 2017) and the commonly used one-class method, namely the Gaussian domain descriptor (GDD) (David 2001), where we tune the hyparameters of GDD by assuming that the labels of negative samples are known.

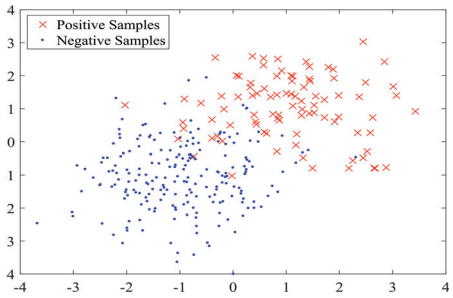


Figure 2: An example of the synthetic dataset with the positive class prior of 0.3 and $\mu = 1$.

Dataset Description

Synthetic Datasets. The synthetic datasets also used in (du Plessis and Sugiyama 2014; du Plessis, Niu, and Sugiyama 2014) were created by sampling from predefined class-conditional 2D normal distributions, which are

$$p(\mathbf{v}|q_1 = 1) = \mathcal{N}(\mu, \Sigma_0);$$
$$p(\mathbf{v}|q_2 = 1) = \mathcal{N}(\mu_0, \Sigma_0). \quad (21)$$

where $\mu_0 = [-1, -1]$, $\mu = [\mu, \mu]$ and Σ_0 is a diagonal matrix with the vector $[1, 1]$ on the main diagonal. The value of μ controls the overlap between the class conditional PDFs, where a small μ implies a high overlap, while a large μ means a low overlap. By varying the value of the class prior and the value of μ, we obtain some datasets with different class priors and different overlaps (see Fig. 2).

Real Datasets. The MNIST dataset (LeCun et al. 1998) and the scene dataset (Boutell et al. 2004) are used. MNIST is a handwritten digit dataset that includes ten classes of 0-9 and contains 60,000 training images and 10,000 test images. Additionally, the scene dataset is an image dataset with 2407 samples in 6 classes. To make them appropriate for LPU problems, we extracted the samples from different class pairs to form new datasets. Specifically, for the MNIST dataset, in every experiment, six new datasets were obtained by extracting the samples from the class pairs, including 7vs1, 6vs9, 0vs6, 1vs5, 3vs5 and 5vs8. Every dataset contains the *P* set, the *U* set and the *T* set. The *P* set consists of 1% of the samples selected randomly from the first digit of one pair in the training set. The *U* set consists of another randomly selected 10% of the samples of the pair in the training set. The *T* set contains all the samples of this pair in the testing set. For the scene dataset, in every experiment, six new datasets were obtained by considering one in the six classes as the positive class. For the positive class, 25% of the samples were randomly selected to comprise the *P* set. The rest of the samples comprise the *U* set as well as the *T* set.

Model Development

The Proposed Method (called LPUb below). The value of the hyperparameter ε in (12) was fixed at a small value of 0.1 to guarantee that the soft version of H_ε and δ_ε is close enough to their hard version. We used the models with 200 latent variables; the learning rate in (19) was set to 1e-3; and the momentum was set to 0.5 in the first 5 cycles and increased to 0.9 after that. The models were trained using CD-1 until the number of iterations reached 4000. The temperature parameter introduced to scale free energies as in (Nair and Hinton 2009) was set to 100.

CSLPU. It was implemented by du Plessis, Niu, and Sugiyama (2014). We used a Gaussian RBF kernel and followed the empirical approach in (du Plessis, Niu, and Sugiyama 2014) to tune the hyperparameters. Moreover, CSLPU needs the class prior to be known first. We used the method in (du Plessis and Sugiyama 2014) to estimate the class prior, with the parameters tuned under the same setting as CSLPU.

TIcE. TIcE is actually a class prior estimation method implemented by Bekker and Jesse (2018). We combined it with CSLPU to obtain a LPU classifier, which is still called TIcE here to emphasize its class prior estimation method. The hyperparamters of TIcE were set to be the same as in the paper (Bekker and Jesse 2018).

GDD. It was implemented with the data description toolbox (dd_tools) (Tax 2005), where we used the simple Gaussian target distribution and tuned two hyperparameters by grid searching: the error on the target class in the range [0.1, 1] with a step of 0.1 and the regularization parameter in the range [0.1, 1] with a step of 0.1. GDD did not need the labeled negative data for training. However, to investigate its best achievable performance, we tuned the hyperparameters using both the *P* set and the *T* set with labels.

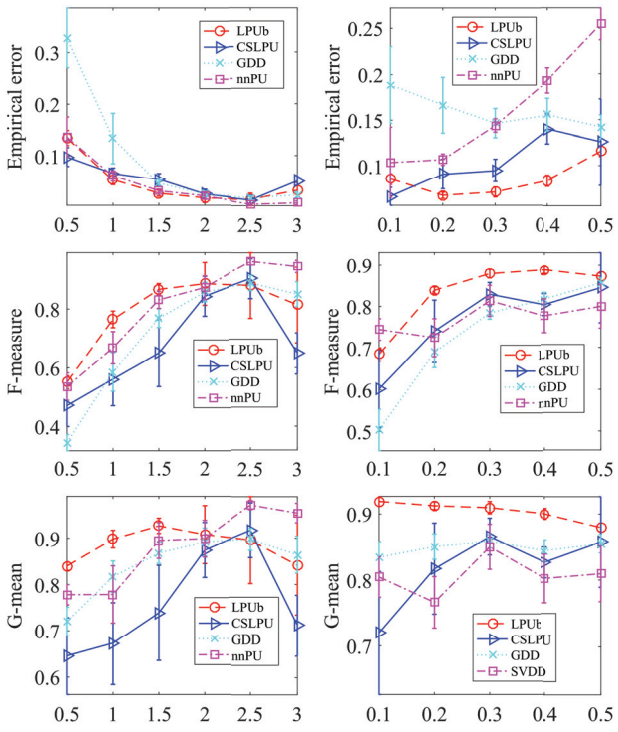
nnPU. It was implemented by Kiryo et al. (2017), where a 6-layer network structure with ReLU (more specifically, a structure of d-300-300-300-300-1 with d being the number of input dimensions) was adopted by the model; the hyperparamters β and γ were set to 0 and 1 respectively. Also, nnPU needs knowing the class prior beforehand, which would be estimated by the method in (du Plessis and Sugiyama 2014) as in the above implementation of CSLPU. The number of iterations was set to its default number 100 for the real datasets and 1000 for the synthetic datasets to promise a better performance.

Evaluation Metrics

The empirical error rate w.r.t. the true positive class prior π is defined as $\pi FP + (1 - \pi)FN$, where *FP* is the false positive rate and *FN* is the false negative rate. We used it as one of the evaluations as it can reflect how much the classifier model fits the data appropriately. Additionally, for LPU problems, without the aid of the labeled negative data, the classifier easily obtains a small precision or a small sensitivity. Therefore, we also report the F-measure and the G-mean results. The F-measure and G-mean will be large only when both precision and sensitivity are large.

Experiments on Synthetic Datasets

Effect of Class Overlap. Every experiment was repeated ten times. In every experiment, given the positive class prior fixed at 0.1, varying μ in Eq. (21) from 0.5 to 3 with a step



(a) Varying overlap (b) Varying class prior

Figure 3: Comparison of empirical error (the first row), F-measure (the second row) and G-mean (the third row) obtained by different methods on synthetic datasets with (a) varying the overlap of two classes and (b) varying the positive class prior.

of 0.5 yielded six synthetic datasets with different overlaps. Every dataset contains the P set, including 10 positive samples, and the U set (simultaneously seen as the T set), including 800 unlabelled samples, where the positive samples are sampled independently from the unlabelled samples. The experimental results represented as the average values of each indicator are recorded in Fig. 3(a). Fig. 3(a) shows that with increasing μ, the classification results of all methods improve more or less because the class overlap of the datasets decreases. Further, we can see that LPUb provides a relatively better classification results. Particularly, when $\mu = 0.5, 1, 1.5, 2$, LPUb achieves an higher F-measure and G-mean, and almost the minimal empirical error than that of the other methods. This result indicates that the model trained by LPUb can fit the data well, thereby implying an appropriate estimation of the class prior may be obtained implicitly by LPUb under the condition that only a small set of labelled positive samples exists and there is a significant overlap between the two classes. CSLPU and nnPU are inferior to LPUb at these μ values, which may be attributed to the two-step strategies adopt by the class prior-based methods. Too few positive samples with a high overlap between classes may lead to the estimation bias of the class prior, thereby causing the performance decline. Additionally, TIcE is not compared here. TIcE tends to give a very small class prior estimation in these experiments, due to the violation of the selected-completely-at-random assumption on the synthetic datasets. The selected-completely-at-random assumption is the basic assumption of many class prior estimation methods such as (Bekker and Jesse 2018; Elkan and Noto 2008). We will compare TIcE with the other methods on the real datasets.

Effect of Class Prior. Given $\mu = 1$, we sampled from the mixture PDF samples by varying the positive class prior π from 0.1 to 0.5 with a step of 0.1 to acquire five datasets in every experiment. Every dataset has 800 samples that comprise the U set (also seen as the T set), and an extra 30 independently sampled positive samples to comprise the P set. Every experiment was performed ten times. The results are recorded as the average values of the indicators and shown in Fig. 3(b). As seen from Fig. 3(b), LPUb outperforms the other three methods with the F-measure, G-mean and empirical error except that at $\pi = 0.1$, CSLPU is better in the empirical error; and nnPU is better in the F-measure. This observation testifies that LPUb can perform very well with different class prior, even when the datasets are highly imbalanced. CSLPU is the suboptimal method. It outperforms GDD and nnPU with empirical error and F-measure at almost all values of π. However, we still observe a high variation of the performance of CSLPU when the positive class prior is relatively small. One reason for this phenomenon is the inaccurate estimation of class prior. Especially, when $\pi = 0.1$, the number of labelled positive samples may be too sufficient to get a small positive class prior. Nonetheless, such problems are not severe for LPUb because LPUb requires that the model PDF meets the mixture PDF of unlabelled samples, which makes LPUb explore the data manifold and rectify the current class prior implicitly.

Experiments on Real Datasets

Experiments were performed on the MNIST and scene datasets. Every experiment was repeated five times. The average performance results of the newly created datasets are reported in Fig. 4(a) and Fig. 4(b) seperately according to the data sources. From both (a) and (b) of Fig. 4, we can see LPUb provides the best performance regarding the F-measure, G-mean and empirical error rate on almost each of the new 12 datasets, except that on a few datasets, the other methods are slightly better than LPUb in some indicators. This shows that LPUb is an effective practical method. CSLPU as well as TIcE on the MNIST datasets and nnPU on the scene datasets are the suboptimal methods. The ability to learn using unlabeled data makes them generally perform much better than the one-class method, namely GDD, although GDD tuned the hyperparameters in both the P set and the T set with labels. Notably, Kiryo et al. (2017) has shown that nnPU is expected to get a better performance than other LPU methods, e.g. CSLPU., as it solved the overfitting problem caused by the negative loss functions. However, we found that the performance of nnPU on some reconstructed MNIST datasets is not good and varies largely. This may be due to the distorted estimation of class prior ob-

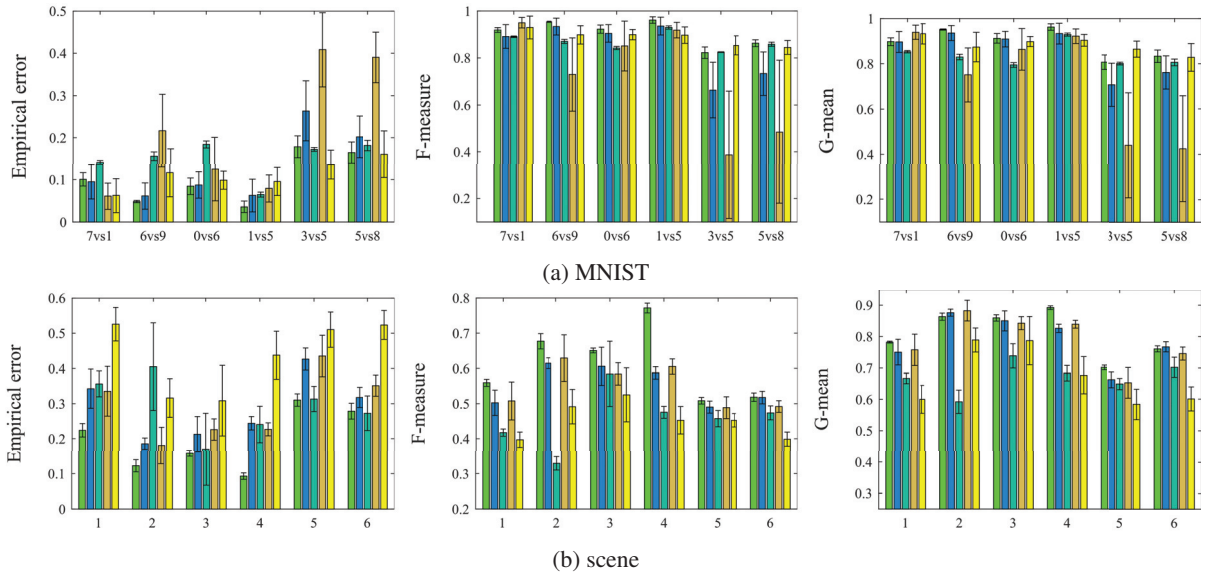


Figure 4: Comparison of empirical error (the first column), F-measure (the second column) and G-mean (the third column) obtained by different methods on the new datasets from (a) MNIST and (b) scene. The methods, in turn, in every group are LPUb, CSLPU, GDD, nnPU and TIcE

tained under the current experiment settings, which severely influences the consequent training process of nnPU.

Conclusion

In this paper, we addressed the problem of learning from positive and unlabeled data and presented a new learning strategy. In theory, we proved that this strategy can lead to a good estimation of the negative class conditional density for the class prior and proposed an implicit RBM model-based LPU method. A series of experiments verified the superiority of the new method. In future work, we will consider applying the kernel trick (Kondor and Jebara 2003; Jebara and Kondor 2003) or the deep learning architecture to further improve the performance of LPU methods (Chiaroni et al. 2018).

Acknowledgments

This work is funded in part by the National Key R&D Program of China (2017YFE0111900), the National Natural Science Foundation of China (61650303, 61876129) and Hainan Provincial Natural Science Foundation of China (119MS004).

Proof of Theorem 1

Proof. Let α be $p(q_1 = 1)$. The Bhattacharyya coefficient, viewed as a function of α, is expressed as

$$B(\alpha) = \int_{\mathbb{R}^d} \sqrt{\frac{p(\mathbf{v}|q_1 = 1)\left(p(\mathbf{v}) - \alpha p(\mathbf{v}|q_1 = 1)\right)}{1 - \alpha}} \mathrm{d}\mathbf{v}, \tag{22}$$

where $0 \leq \alpha \leq \hat{\alpha} < 1$ and

$$\hat{\alpha} = \inf_{p(\mathbf{v}|q_1=1)\neq 0} \left\{ \frac{p(\mathbf{v})}{p(\mathbf{v}|q_1 = 1)} \right\}. \tag{23}$$

We first prove $\hat{\alpha}$ is the minimum point of $B(\alpha)$. The derivative of $B(\alpha)$ with respect to α is proportional to

$$(1 - \alpha)^{-\frac{3}{2}} \int_{\mathbb{R}^d} \sqrt{p(\mathbf{v}|q_1 = 1)} \frac{p(\mathbf{v}) - p(\mathbf{v}|q_1 = 1)}{\sqrt{p(\mathbf{v}) - \alpha p(\mathbf{v}|q_1 = 1)}} \mathrm{d}\mathbf{v}. \tag{24}$$

Since $\alpha < 1$ and

$$\begin{aligned}
&\int_{\mathbb{R}^d} \sqrt{p(\mathbf{v}|q_1 = 1)} \frac{p(\mathbf{v}) - p(\mathbf{v}|q_1 = 1)}{\sqrt{p(\mathbf{v}) - \alpha p(\mathbf{v}|q_1 = 1)}} \mathrm{d}\mathbf{v} \\
&= \int_A \frac{p(\mathbf{v}) - p(\mathbf{v}|q_1 = 1)}{\sqrt{\frac{p(\mathbf{v})}{p(\mathbf{v}|q_1=1)} - \alpha}} \mathrm{d}\mathbf{v} - \int_{A'} \frac{p(\mathbf{v}|q_1 = 1) - p(\mathbf{v})}{\sqrt{\frac{p(\mathbf{v})}{p(\mathbf{v}|q_1=1)} - \alpha}} \mathrm{d}\mathbf{v} \\
&< \int_A \frac{p(\mathbf{v}) - p(\mathbf{v}|q_1 = 1)}{\sqrt{1 - \alpha}} \mathrm{d}\mathbf{v} - \int_{A'} \frac{p(\mathbf{v}|q_1 = 1) - p(\mathbf{v})}{\sqrt{1 - \alpha}} \mathrm{d}\mathbf{v} \\
&= 0,
\end{aligned} \tag{25}$$

then $B'(\alpha) < 0$ and $B(\alpha)$ is a monotonically decreasing function, where $A = \{\mathbf{v}|p(\mathbf{v}) \geq p(\mathbf{v}|q_1 = 1)\}$ and $A' = \{\mathbf{v}|p(\mathbf{v}) < p(\mathbf{v}|q_1 = 1)\}$. Recall that $\alpha \leq \hat{\alpha}$, then $B(\alpha)$ has the mimimum value at the point $\hat{\alpha}$. Next, we prove

$$\hat{\alpha} \in \left[p(q_1 = 1), \left(\int_{\mathbb{R}^d} \frac{p^2(\mathbf{v}|q_1 = 1)}{p(\mathbf{v})} \mathrm{d}\mathbf{v} \right)^{-1} \right]. \tag{26}$$

Note that $p(q_1 = 1)$ is within the feasible region of α, so we have $p(q_1 = 1) \leq \hat{\alpha}$ directly. On the other hand, following $\hat{\alpha} \leq \frac{p(\mathbf{v})}{p(\mathbf{v}|q_1=1)}$ $(\forall \mathbf{v})$, we obtain

$$\int_{\mathbb{R}^d} \hat{\alpha} \frac{p(\mathbf{v}|q_1 = 1)}{p(\mathbf{v})} p(\mathbf{v}|q_1 = 1) \mathrm{d}\mathbf{v} \leq \int_{\mathbb{R}^d} p(\mathbf{v}|q_1 = 1)\, \mathrm{d}\mathbf{v}. \tag{27}$$

Since

$$\int_{\mathbb{R}^d} p(\mathbf{v}|q_1 = 1) \mathrm{d}\mathbf{v} = 1. \tag{28}$$

Then,by Eq. (27), we obtain

$$\hat{\alpha} \leq \left(\int_{\mathbb{R}^d} \frac{p^2(\mathbf{v}|q_1=1)}{p(\mathbf{v})} \mathrm{d}\mathbf{v} \right)^{-1}. \tag{29}$$

Finally, we will prove $\hat{\alpha}$ is equal to $p(q_1=1)$ when the intersection of $Supp\left\{p(\mathbf{v}|q_2=1)\right\}$ and $\left(Supp\left\{p(\mathbf{v}|q_1=1)\right\}\right)^c$ is not a null set. Following from the law of total probability, we have

$$\begin{aligned} p(\mathbf{v}) &= p(q_1=1)p(\mathbf{v}|q_1=1) \\ &\quad + (1-p(q_1=1))\, p(\mathbf{v}|q_2=1). \end{aligned} \tag{30}$$

Then

$$\begin{aligned} &\inf_{p(\mathbf{v}|q_1=1)\neq 0} \left\{ \frac{p(\mathbf{v})}{p(\mathbf{v}|q_1=1)} \right\} = p(q_1=1) \\ &+ (1-p(q_1=1)) \inf_{p(\mathbf{v}|q_1=1)\neq 0} \left\{ \frac{p(\mathbf{v}|q_2=1)}{p(\mathbf{v}|q_1=1)} \right\}. \end{aligned} \tag{31}$$

Apply the restrictions on the supports of densities, we see

$$\inf_{p(\mathbf{v}|q_1=1)\neq 0} \left\{ \frac{p(\mathbf{v}|q_2=1)}{p(\mathbf{v}|q_1=1)} \right\} = 0. \tag{32}$$

So we have $\hat{\alpha} = p(q_1=1)$. □

References

Bekker, J., and Jesse, D. 2018. Estimating the class prior in positive and unlabeled data through decision tree induction. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, 2712–2719. AAAI Press.

Boutell, M. R.; Luo, J.; Shen, X.; and Brown, C. M. 2004. Learning multi-label scene classification. *Pattern recognition* 37(9):1757–1771.

Chan, T. F., and Vese, L. A. 2001. Active contours without edges. *IEEE Transactions on image processing* 10(2):266–277.

Chiaroni, F.; Rahal, M.-C.; Hueber, N.; and Dufaux, F. 2018. Learning with a generative adversarial network from a positive unlabeled dataset for image classification. In *Proceedings of the 25th IEEE International Conference on Image Processing (ICIP)*, 1368–1372. IEEE.

Comaniciu, D.; Ramesh, V.; and Meer, P. 2003. Kernel-based object tracking. *IEEE Transactions on pattern analysis and machine intelligence* 25(5):564–577.

David, MJ, T. 2001. *One-class classification, Concept-learning in the absence of counter-examples*. Thesis, Delft Univ.Technol.

Denis, F. 1998. *PAC learning from positive statistical queries*. Algorithmic Learning Theory. Berlin: Springer berlin heidelberg.

du Plessis, M. C., and Sugiyama, M. 2014. Class prior estimation from positive and unlabeled data. *IEICE Transactions on Information and Systems* 97(5):1358–1362.

du Plessis, M. C.; Niu, G.; and Sugiyama, M. 2014. Analysis of learning from positive and unlabeled data. In *Advances in Neural Information Processing Systems 27 (NIPS)*, 703–711. MIT Press.

du Plessis, M. C.; Niu, G.; and Sugiyama, M. 2015. Convex formulation for learning from positive and unlabeled data. In *Proceedings of the 32th International Conference on Machine Learning (ICML)*, 1386–1394. JMLR.org.

du Plessis, M. C.; Niu, G.; and Sugiyama, M. 2017. Class-prior estimation for learning from positive and unlabeled data. *Machine Learning* 106(4):463–492.

Elkan, C., and Noto, K. 2008. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM international conference on Knowledge discovery and data mining (SIGKDD)*, 213–220. ACM.

Fukunaga, K. 2013. *Introduction to statistical pattern recognition*. Academic Press.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation* 14(8):1771–1800.

Jain, S.; White, M.; and Radivojac, P. 2016. Estimating the class prior and posterior from noisy positives and unlabeled data. In *Advances in Neural Information Processing Systems 29 (NIPS)*, 2685–2693. MIT Press.

Jebara, T., and Kondor, R. 2003. Bhattacharyya and expected likelihood kernels. In *Learning theory and kernel machines*. Springer. 57–71.

Kiryo, R.; Niu, G.; du Plessis, M. C.; and Sugiyama, M. 2017. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems 30 (NIPS)*, 1675–1685. MIT Press.

Kondor, R., and Jebara, T. 2003. A kernel between sets of vectors. In *Proceedings of the 20th International Conference on Machine Learning (AAAI)*, 361–368. AAAI Press.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11):2278–2324.

Li, X., and Liu, B. 2003. Learning to classify texts using positive and unlabeled data. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI)*, volume 3, 587–592. Morgan Kaufmann.

Liu, B.; Lee, W. S.; Yu, P. S.; and Li, X. 2002. Partially supervised classification of text documents. In *Proceedings of the 19th International Conference on Machine Learning (ICML)*, 387–394. Morgan Kaufmann.

Liu, B.; Dai, Y.; Li, X.; Lee, W. S.; and Yu, P. S. 2003. Building text classifiers using positive and unlabeled examples. In *Proceedings of the 3th IEEE International Conference on Data Mining (ICDM)*, 179–186. IEEE.

Nair, V., and Hinton, G. E. 2009. Implicit mixtures of restricted boltzmann machines. In *Advances in Neural Information Processing Systems 22 (NIPS)*, 1145–1152. MIT Press.

Ray, S. 1989. On a theoretical property of the bhattacharyya coefficient as a feature evaluation criterion. *Pattern Recognition Letters* 9(5):315–319.

Tax, D. 2005. Ddtools, the data description toolbox for matlab. *Delft University of Technology ed*.

Taylor, G. W.; Sigal, L.; Fleet, D. J.; and Hinton, G. E. 2010. Dynamical binary latent variable models for 3d human pose tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 631–638. IEEE.

Wang, C.; Ding, C.; Meraz, R. F.; and Holbrook, S. R. 2006. Psol: a positive sample only learning algorithm for finding non-coding rna genes. *Bioinformatics* 22(21):2590–2596.

Yu, H.; Han, J.; and Chang, K.-C. 2004. Pebl: Web page classification without negative examples. *IEEE Transactions on Knowledge and Data Engineering* 16(1):70–81.

Yu, H. 2005. Single-class classification with mapping convergence. *Machine Learning* 61(1-3):49–69.