Domain adaptation of weighted majority votes via perturbed variation-based self-labeling

Emilie Morvant*
University Jean Monnet of Saint-Etienne, Laboratoire Hubert Curien, UMR CNRS 5516
18 rue du Professeur Benoît Lauras, 42000 Saint-Etienne, France
emilie.morvant@univ-st-etienne.fr

October 2, 2014

Abstract

In machine learning, the domain adaptation problem arrives when the test (target) and the train (source) data are generated from different distributions. A key applied issue is thus the design of algorithms able to generalize on a new distribution, for which we have no label information. We consider the specific PAC-Bayesian situation focused on learning classification models defined as a weighted majority vote over a set of real-valued functions. In this context, we present PV-MinCq a new framework that generalizes the non-adaptive algorithm MinCq. PV-MinCq follows the next principle. Justified by a theoretical bound on the target risk of the vote, we provide to MinCq a target sample labeled thanks to a perturbed variation-based self-labeling focused on the regions where the source and target marginals appear similar. We also study the influence of our self-labeling, from which we deduce an original process for tuning the hyperparameters. Finally, our experiments show very promising results.

Keywords: Machine learning; Classification; Domain adaptation; Majority vote; PAC-Bayes

1 Introduction

Nowadays, due to the expansion of Internet a large amount of data is available. Then, many applications need to make use of supervised machine learning methods able to transfer knowledge from different information sources, which is known as transfer learning[1]. In such a situation, we cannot follow the strong standard assumption in machine learning that supposes the learning and test data drawn from the same unknown distribution. For instance, one of the tasks of the common spam filtering problem consists in adapting a model from one user to a new one who receives significantly different emails. This scenario, called domain adaptation (DA), arises when we aim at learning from a source distribution a well performing model on a different target distribution, for which one considers an unlabeled sample (or few labels)[2]. In this paper we design a new DA framework when we have no target label. This latter situation is known to be challenging (Ben-David and Urner, 2012).

To address this kind of issues, several approaches exist in the literature[3]. Among them, the instance weighting-based methods allow us to deal with the covariate-shift where the distributions differ only in their marginals (*e.g.* Huang et al. (2007)). Another technique is to exploit self-labeling procedures. However, it often relies on iterative and heavy self-labeling. For example, one of the reference methods is DASVM (Bruzzone and Marconcini, 2010). Concretely at each iteration, DASVM learns a SVM classifier from the labeled source examples, then some of them are replaced by target data auto-labeled with this SVM classifier[4]. A third popular solution is to take advantage of a distance between distributions, with the intuition that we want to minimize this divergence while preserving good performance on the source

*Most of the work in this paper was carried out while Emilie Morvant was affiliated with Institute of Science and Technology (IST) Austria, 3400 Klosterneuburg.

[1] See (Pan and Yang, 2010; Quionero-Candela et al., 2009) for surveys on transfer learning

[2] The task with few target labels is sometimes referred to as semi-supervised DA, and the one without target label as unsupervised DA.

[3] See (Margolis, 2011) for a survey on DA.

[4] In DASVM, the self-labeled points correspond to those with the lowest confidence, and the deleted source points are those with the highest confidence.

data: If the distributions are close under this measure, then generalization ability may be "easier" to quantify. The most popular divergences, such as the $\mathcal{H}\Delta\mathcal{H}$-divergence of Ben-David et al. (2007, 2010) and the discrepancy of Mansour et al. (2008), involve the disagreement between classifiers. Although they lead to different analyses, they enhance to the same conclusion that is the disagreement between classifiers must be controlled while keeping a good source performance. Obviously, other divergences for evaluating how much two distributions differ exist in the literature and could be investigated in a DA scenario. For example, we can cite the perturbed variation (PV) (Harel and Mannor, 2012) on which we will pay our attention for designing a non-iterative self-labeling process. This measure is based on the following principle: Two samples are similar if each instance of one sample is close to an instance of the other sample.

In this work, we investigate the special issue of PAC-Bayesian DA introduced by Germain et al. (2013), which focuses on learning target weighted majority votes over a set of classifiers (or voters). Their analysis stands in the class of approaches based on a divergence between distributions. This latter, called the domain disagreement, has been justified by a tight bound over the risk of the majority vote— the C-bound (Lacasse et al., 2007) —and has the advantage to take into account the expectation of the disagreement between pairs of voters. Although their theoretical analysis is elegant and well-founded, the algorithm derived is restricted to linear classifiers. We then intend to design a learning framework able to deal with weighted majority votes over real-valued voters in this PAC-Bayesian DA scenario. With this aim in mind and knowing the C-bound has lead to a simple and well performing algorithm for supervised classification, called MinCq (Laviolette et al., 2011), we extend it to DA thanks to a non-iterative self-labeling. Firstly, we propose a new formulation of the C-bound suitable for every self-labeling function (which associates a label to an example). Then, we design such a function with the help of the empirical PV. Concretely, our PV-based self-labeling focuses on the regions where the source and target marginals are closer, then it labels the (unlabeled) target sample only in these regions (see Figure 1, in Section 3.2). This self-labeled sample is then provided to MinCq. Afterwards, we highlight the influence of our self-labeling, and deduce an original validation procedure. Finally, our framework, named PV-MinCq, implies good and promising results, better than a nearest neighborhood-based self-labeling, and than other DA methods.

The rest of the paper is organized as follows. Section 2 recalls the PAC-Bayesian DA setting of (Germain et al., 2013), and then MinCq and its theoretical basis in the supervised setting (Laviolette et al., 2011). In Section 3 we present PV-MinCq, our adaptive MinCq based on a PV-based self-labeling procedure. Before conlude, we experiment our framework on a synthetic problem in Section 4.

2 Notations and background

In this section, we first review the PAC-Bayesian setting in a non-adaptive setting, and then the results of Germain et al. (2013) and Laviolette et al. (2011).

2.1 PAC-Bayesian setting in supervised learning

We recall the usual setting of the PAC-Bayesian theory—introduced by McAllester (1999)—which offers generalization bounds (and algorithms) for weighted majority votes over a set of real-valued functions, called voters.

Let $X \subseteq \mathbb{R}^d$ be the input space of dimension d and $Y = \{-1, +1\}$ be the output space, *i.e.* the set of possible labels. P_S is an unknown distribution over $X \times Y$, that we called a domain. $(P_S)^{m_s} = \bigotimes_{s=1}^{m_s} P_S$ stands for the distribution of a m_s-sample. The marginal distribution of P_S over X is denoted by D_S. We consider $S = \{(\mathbf{x}_s, y_s)\}_{s=1}^{m_s}$ a m_s-sample independent and identically distributed (*i.i.d.*) according to $(P_S)^{m_s}$, commonly called the learning sample. Let $\mathcal{H}$ be a set of n (bounded) real-valued voters such that: $\forall h \in \mathcal{H},\ h : X \to \mathbb{R}$. Given $\mathcal{H}$, the ingredients of the PAC-Bayesian approache are a prior distribution π over $\mathcal{H}$, a learning sample S and a posterior distribution ρ over $\mathcal{H}$. Prior distribution π models an *a priori* belief on what are the best voters from $\mathcal{H}$, before observing the learning sample S. Then, given the information provided by S, the learner aims at finding a posterior distribution ρ leading to a ρ-weighted majority vote B_ρ over $\mathcal{H}$ with nice generalization guarantees. B_ρ and its true and empirical risks are defined as follows.

Definition 1. *Let $\mathcal{H}$ be a set of real-valued voters. Let ρ be a distribution over $\mathcal{H}$. The ρ-weighted*

majority vote B_ρ (sometimes called the Bayes classifier) is:

$$\forall \mathbf{x} \in X,\ B_\rho(\mathbf{x}) = \operatorname{sign}\left[\mathop{\mathbf{E}}_{h\sim\rho} h(\mathbf{x})\right].$$

The true risk of B_ρ on a domain P_S and its empirical risk[5] on a m_s-sample S are respectively:

$$\mathbf{R}_{P_S}(B_\rho) = \frac{1}{2}\left(1 - \mathop{\mathbf{E}}_{(\mathbf{x}_s,y_s)\sim P} y_s B_\rho(\mathbf{x}_s)\right),$$
$$\mathbf{R}_S(B_\rho) = \frac{1}{2}\left(1 - \frac{1}{m_s}\sum_{s=1}^{m_s} y_s B_\rho(\mathbf{x}_s)\right).$$

Usual PAC-Bayesian analyses[6] do not directly focus on the risk of B_ρ, but bound the risk of the closely related stochastic Gibbs classifier G_ρ. It predicts the label of an example $\mathbf{x}$ by first drawing a classifier h from $\mathcal{H}$ according to ρ, and then it returns $h(\mathbf{x})$. The risk of G_ρ corresponds thus to the expectation of the risks over $\mathcal{H}$ according to ρ:

$$\mathbf{R}_P(G_\rho) = \mathop{\mathbf{E}}_{h\sim\rho} \mathbf{R}_P(h) = \frac{1}{2}\left(1 - \mathop{\mathbf{E}}_{(\mathbf{x}_s,y_s)\sim P}\ \mathop{\mathbf{E}}_{h\sim\rho} y_s h(\mathbf{x}_s)\right). \tag{1}$$

Note that it is well-known in the PAC-Bayesian literature that the deterministic B_ρ and the stochastic G_ρ are related by:

$$\mathbf{R}_P(B_\rho) \leq 2\mathbf{R}_P(G_\rho). \tag{2}$$

2.2 PAC-Bayesian domain adaptation of the Gibbs classifier

Throughout the rest of this paper, we consider the PAC-Bayesian DA setting introduced by Germain et al. (2013). The main difference between supervised learning and DA is that we have two different domains over $X \times Y$: the source domain P_S and the target domain P_T (D_S and D_T are the respective marginals over X). The aim is then to learn a good model on the target domain P_T knowing that we only have label information from the source domain P_S. Concretely, in the setting described in Germain et al. (2013), we have a labeled source m_s-sample $S = \{(\mathbf{x}_s, y_s)\}_{t=1}^{m_s}$ *i.i.d.* from $(P_S)^{m_s}$ and a target unlabeled m_t-sample $T = \{\mathbf{x}_t\}_{t=1}^{m_t}$ *i.i.d.* from $(D_T)^{m_t}$. One thus desires to learn from S and T a weighted majority vote with lowest possible expected risk on the target domain $\mathbf{R}_{P_T}(B_\rho)$, *i.e.* with good generalization guarantees on P_T. Recalling that usual PAC-Bayesian generalization bound study the risk of the Gibbs classifier, Germain et al. (2013) have done an analysis of its target risk $\mathbf{R}_{P_T}(G_\rho)$. Their main result is the following theorem.

Theorem 1 (Theorem 4 of (Germain et al., 2013) applied to real-valued voters)**.** *Let $\mathcal{H}$ be a set of real-valued voters. For every distribution ρ over $\mathcal{H}$, we have:*

$$\mathbf{R}_{P_T}(G_\rho) \leq \mathbf{R}_{P_S}(G_\rho) + \operatorname{dis}_\rho(D_S, D_T) + \lambda_\rho, \tag{3}$$

where $\operatorname{dis}_\rho(D_S, D_T)$ is the domain disagreement between the marginals D_S and D_T, and is defined by:

$$\operatorname{dis}_\rho(D_S, D_T) = \left|\mathop{\mathbf{E}}_{(h,h')\sim\rho^2}\left(\mathop{\mathbf{E}}_{\mathbf{x}_t\sim D_T} h(\mathbf{x}_t)h'(\mathbf{x}_t) - \mathop{\mathbf{E}}_{\mathbf{x}_s\sim D_S} h(\mathbf{x}_s)h'(\mathbf{x}_s)\right)\right|, \tag{4}$$

and λ_ρ is related[7] to the true labeling on P_S and P_T.

Note that this bound reflects the usual philosophy in DA: It is well known that a favorable situation for DA arrives when the divergence between the domains is small while achieving good source performance (Ben-David et al., 2007, 2010; Mansour et al., 2008). Germain et al. (2013) have then derived a first

[5] We express the risk with the linear loss since we deal with real-valued voters, but in the special case of B_ρ the linear loss is equivalent to the $0-1$-loss.

[6] Usual PAC-Bayesian analyses can be found in (McAllester, 2003; Seeger, 2002; Langford, 2005; Catoni, 2007; Germain et al., 2009).

[7] In practice, we cannot compute λ_ρ, since it depends greatly on the unavailable target labels. We then suppose that it is negligible. Thus, we do not develop this point here, but more details can be found in (Germain et al., 2013).

promising algorithm called PBDA for minimizing this trade-off between source risk and domain disagreement. Although PBDA has shown the usefulness of PAC-Bayes for tackling DA, it remains specific to linear classifiers, it does not directly focus on the majority vote, and does not provide the best empirical results regarding to state-of-the-art methods.

In this paper, our goal is to tackle this drawbacks to propose a novel algorithm for learning an adaptive weighted majority vote over a set of real-valued voters. To do so, the point which calls our attention here is the domain disagreement of Equation (4). Indeed, it finds its root in the theoretical bound (the C-bound (Lacasse et al., 2007)) over the (source) risk of the majority vote, from which (Laviolette et al., 2011) have derived an elegant and performing non-adaptive algorithm for learning a weighted majority vote over a set of real-valued voters (MinCq). We recall now these non-DA results, then we extend them to DA in Section 3.1.

2.3 MinCq a supervised algorithm for learning majority votes

The classical relation between the stochastic G_ρ and the majority vote B_ρ (Equation (2)) can be very loose. To tackle this drawback, Lacasse et al. (2007) and Laviolette et al. (2011) have proven a recent tighter relation stated in the following in Theorem 2 (the C-bound). This result is based on the notion of ρ-margin defined as follows.

Definition 2 (Laviolette et al. (2011))**.** *The ρ-margin of an example $(\mathbf{x}, y) \in X \times Y$ realized on the distribution ρ of support $\mathcal{H}$ is given by: $\mathop{\mathbf{E}}_{h\sim\rho} yh(\mathbf{x})$.*

By definition of B_ρ, it is easy to see that B_ρ correctly classifies an example $\mathbf{x}_s$ if the ρ-margin is strictly positive. Thus, under the convention that if $y_s \mathbf{E}_{h\sim\rho}\, h(\mathbf{x}_s) = 0$, then B_ρ commits an error on $(\mathbf{x}_s, y_s)$, for every domain P_S on $X \times Y$, we have:

$$\mathbf{R}_{P_S}(B_\rho) = \mathop{\mathbf{Pr}}_{(\mathbf{x}_s,y_s)\sim P_S} \left(\mathop{\mathbf{E}}_{h\sim\rho} y_s h(\mathbf{x}_s) \leq 0 \right).$$

Knowing this, Lacasse et al. (2007) and Laviolette et al. (2011) have proven the following C-bound over $\mathbf{R}_P(B_\rho)$ by making use of the Cantelli-Chebitchev inequality.

Theorem 2 (The C-bound as expressed in Laviolette et al. (2011))**.** *For all distribution ρ over $\mathcal{H}$, for all domain P_S over $X \times Y$ of marginal (over X) D_S, if $\mathop{\mathbf{E}}_{h\sim\rho} \mathop{\mathbf{E}}_{(\mathbf{x}_s,y_s)\sim P_S} y_s h(\mathbf{x}_s) > 0$, then:*

$$\mathbf{R}_{P_S}(B_\rho) \leq 1 - \frac{\left(\mathop{\mathbf{E}}_{h\sim\rho} \mathop{\mathbf{E}}_{(\mathbf{x}_s,y_s)\sim P_S} y_s h(\mathbf{x}_s) \right)^2}{\mathop{\mathbf{E}}_{(h,h')\sim\rho^2} \mathop{\mathbf{E}}_{\mathbf{x}_s\sim D_S} h(\mathbf{x}_s) h'(\mathbf{x}_s)}.$$

The numerator of this bound corresponds in fact to the first moment of the ρ-margin of B_ρ realized on P_S, which is related to the risk of the Gibbs classifier (Equation (1)). The denominator is the second moment of this ρ-margin, which can be seen as a measure of disagreement between the voters from $\mathcal{H}$ (the lowest this value is, the more the voters disagree) and can be related to the domain disagreement (Equation (4)).

In the supervised setting, Laviolette et al. (2011) have then proposed to minimize the empirical counterpart of the C-bound for learning a good majority vote over $\mathcal{H}$, justified by an elegant PAC-Bayesian generalization bound. Following this principle the authors have derived a quadratic program called MinCq and described in Algorithm 1. Concretely, MinCq learns a weighted majority vote by optimizing the empirical C-bound measured on the learning sample S: It minimizes the denominator, *i.e.* the disagreement (Equation (5)), given a fixed numerator *i.e.* a fixed risk of the Gibbs classifier (Equation (6)), under a particular regularization (Equation (7))[8]. Note that, MinCq has showed good performances on supervised classification tasks.

The key point here is that, through a DA point of view, the C-bound, and thus MinCq, focus on the trade-off suggested by Theorem 3. Indeed, the definition of the domain disagreement (Equation (4)) is related to the C-bound according to the following statement: If source and target risks of the Gibbs

[8] For more technical details on MinCq please see (Laviolette et al., 2011).

Algorithm 1 MinCq$(S, \mathcal{H}, \mu)$

input A m_s-sample $S = \{(\mathbf{x}_s, y_s)\}_{s=1}^{m_s} \sim (P_S)^{m_s}$, a set of n voters $\mathcal{H} = \{h_1, \ldots, h_n\}$, a desired margin $\mu > 0$

output $B_\rho(\cdot) = \text{sign}\left[\sum_{j=1}^{n} \left(2\rho_j - \frac{1}{n}\right) h_j(\cdot)\right]$

$$\textbf{Solve} \quad \underset{\boldsymbol{\rho}}{\text{argmin}} \quad \boldsymbol{\rho}^T \mathbf{M} \boldsymbol{\rho} - \mathbf{A}^T \boldsymbol{\rho}, \tag{5}$$

$$\textbf{s.t.} \quad \mathbf{m}^T \boldsymbol{\rho} = \frac{\mu}{2} + \frac{1}{2nm_s} \sum_{j=1}^{n} \sum_{s=1}^{m_s} y_s h_j(\mathbf{x}_s), \tag{6}$$

$$\forall j \in \{1, \ldots, n\}, \quad 0 \leq \rho_j \leq \tfrac{1}{n}, \tag{7}$$

where $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_n)^\top$ is a weight vector, and $\mathbf{M}$ is the $n \times n$ matrix formed by $\sum_{s=1}^{m_s} \frac{h_j(\mathbf{x}_s) h_{j'}(\mathbf{x}_s)}{m_s}$ for $(j, j') \in \{1, \ldots, n\}^2$,

$$\text{and } \mathbf{m} = \left(\tfrac{1}{m_s} \sum_{s=1}^{m_s} y_s h_1(\mathbf{x}_s), \ldots, \tfrac{1}{m_s} \sum_{s=1}^{m_s} y_s h_n(\mathbf{x}_s)\right)^\top,$$

$$\text{and } \mathbf{A} = \left(\sum_{j=1}^{n} \sum_{s=1}^{m_s} \frac{h_1(\mathbf{x}_s) h_j(\mathbf{x}_s)}{nm_s}, \ldots, \sum_{j=1}^{n} \sum_{s=1}^{m_s} \frac{h_n(\mathbf{x}_s) h_j(\mathbf{x}_s)}{nm_s}\right)^\top$$

classifier are similar, then the source and target risks of the majority vote are similar when the deviation between the source and target voters' disagreement tends to be low.

We thus now propose to make use of the C-bound and MinCq for designing an original and general framework for learning a majority vote over a set of real-valued voters in a DA scenario.

3 An adaptive MinCq

In this section, we introduce our new DA framework for learning a weighted majority vote over a set of real-valued voters. In order to take advantage of the algorithm MinCq, we first extend the C-bound to the DA setting.

3.1 A C-bound suitable for DA with self-labeling

Given a labeling function $l : X \to Y$, which associates a label $y \in Y$ to an unlabeled (target) example $\mathbf{x}_t \sim D_T$, we propose to rewrite the C-bound as follows.

Corollary 3. *For all distribution ρ over $\mathcal{H}$, for all domain P_T over $X \times Y$ of marginal (over X) D_T, for all labeling functions $l : X \to Y$ such that $\mathop{\mathbf{E}}_{h\sim\rho} \mathop{\mathbf{E}}_{\mathbf{x}_t\sim D_T} l(\mathbf{x}_t) h(\mathbf{x}_t) > 0$, we have:*

$$\mathbf{R}_{P_T}(B_\rho) \leq 1 - \frac{\left(\mathop{\mathbf{E}}_{h\sim\rho} \mathop{\mathbf{E}}_{\mathbf{x}\sim D_T} l(\mathbf{x}_t) h(\mathbf{x}_t)\right)^2}{\mathop{\mathbf{E}}_{(h,h')\sim\rho^2} \mathop{\mathbf{E}}_{\mathbf{x}_t\sim D_T} h(\mathbf{x}_t) h'(\mathbf{x}_t)} + \frac{1}{2}\Big|\mathop{\mathbf{E}}_{(\mathbf{x}_t, y_t)\sim P_T} \big(y_t - l(\mathbf{x}_t)\big)\Big|.$$

Proof. The result comes directly from:

$$\left|\mathbf{R}_{P_T}(B_\rho) - \mathbf{R}_{\widehat{P_T}}(B_\rho)\right| = \tfrac{1}{2}\Big|\mathop{\mathbf{E}}_{(\mathbf{x}_t, y_t)\sim P_T} (y_t - l(\mathbf{x}_t))\Big|,$$

where: $\mathbf{R}_{\widehat{P_T}}(B_\rho) = \frac{1}{2}\Big(1 - \mathop{\mathbf{E}}_{\mathbf{x}_t\sim D_T} l(\mathbf{x}_t) B_\rho(\mathbf{x}_t)\Big)$. □

We can recognize the C-bound of Theorem 2 where the true label y_t of an example $\mathbf{x}_t$ is substituted by $l(\mathbf{x})$. The term $\frac{1}{2}\left|\mathbf{E}_{(\mathbf{x}_t, y_t)\sim P_T} (y_t - l(\mathbf{x}_t))\right|$ can be seen as a divergence between the true labeling and the one provided by l, since it computes the gap between the labeling function and the true labeling one: The more similar l and the true labeling functions, the tighter the bound is. Note that generalization bounds provided by Laviolette et al. (2011) are still valid.

With a DA point of view, it is important to note that only one domain appears in this bound. If we suppose this domain is the target one, it is required to compute a relevant labeling function by making use of the information carried by the source labeled sample S. To tackle the issue of defining this labeling function, that we called a self-labeling function, we follow the intuition that given a labeled source instance

Algorithm 2 $\widehat{PV}(S,T,\epsilon,d)$

input $S=\{\mathbf{x}_s\}_{s=1}^{m_s}$, $T=\{\mathbf{x}_t\}_{t=1}^{m_t}$ are unlabeled samples, a radius $\epsilon>0$, a distance measure $d:X\times X\to\mathbb{R}^+$
output $\widehat{PV}(S,T,\epsilon,d)$

1. $G\leftarrow\Big(V=(A,B),E\Big)$, $A=\{\mathbf{x}_s\in S\}$, $B=\{\mathbf{x}_t\in T\}$, $e_{st}\in E$ if $d(\mathbf{x}_s,\mathbf{x}_t)\leq\epsilon$
2. $M_{ST}\leftarrow$ Maximum matching on G
3. $(S_u,T_u)\leftarrow$ number of unmatched vertices in S, resp. in T
4. Return $\widehat{PV}(S,T,\epsilon,d)=\frac{1}{2}\Big(\frac{S_u}{m_s}+\frac{T_u}{m_t}\Big)$

Algorithm 3 PV-MinCq$(S,T,\mathcal{H},\mu,\epsilon,d)$

input $S=\{(\mathbf{x}_s,y_s)\}_{s=1}^{m_s}$ a source sample, $T=\{\mathbf{x}_t\}_{t=1}^{m_t}$ a target sample, a set of voters $\mathcal{H}$, a margin $\mu>0$, a radius $\epsilon>0$, a distance $d:X\times X\to\mathbb{R}^+$
output $B_\rho(\cdot)$

1. $M_{ST}\leftarrow$ Step 1. and 2. $\widehat{PV}(S,T,\epsilon,d)$
2. $\widehat{T}\leftarrow\{(\mathbf{x}_t,y_s):(\mathbf{x}_t,\mathbf{x}_s)\in M_{ST},\mathbf{x}_t\in T,(\mathbf{x}_s,y_s)\in S\}$
3. return MinCq$(\widehat{T},\mathcal{H},\mu)$

$(\mathbf{x}_s,y_s)\in S$, we want to transfer its label y_s to an unlabeled target point $\mathbf{x}_t$ close to $\mathbf{x}_s$. We thus propose to investigate the perturbed variation (PV) (Harel and Mannor, 2012), a recent measure of divergence between distributions based on this intuition. This gives rise, in the following, to a PV-based self-labeling function, then to a self-labeled target sample on which we can apply MinCq (justified by Corollary 3).

3.2 Adaptive MinCq via PV-based self-labeling

Before designing our self-labeling, we recall the definition of the PV proposed by Harel and Mannor (2012).

Definition 3 (Harel and Mannor (2012))**.** *Let D_S and D_T be two marginal distributions over X and $M(D_S,D_T)$ be the set of all joint distributions over $X\times X$ with marginals D_S and D_T. The PV w.r.t. a distance $d:X\times X\to\mathbb{R}^+$ and $\epsilon>0$ is:*

$$PV(D_S,D_T,\epsilon,d)=\inf_{\nu\in M(D_S,D_T)}\mathop{\mathbf{Pr}}_{\nu}\left[d(\mathcal{X},\mathcal{X}')>\epsilon\right],$$

over all pairs $(D_S,D_T)\sim\nu$, such that the marginal of $\mathcal{X}$ (resp. $\mathcal{X}'$) is D_S (resp. D_T).

In other words, two samples are similar if every target instance is close to a source instance. Note that this measure is consistent and its empirical counterpart $\widehat{PV}(S,T,\epsilon,d)$ can be efficiently computed by a maximum graph matching procedure described in Algorithm 2 (Harel and Mannor, 2012).

In our self-labeling goal, we make use of the maximum graph matching M_{ST} computed at step 2 of Algorithm 2. Concretely, we label the unlabeled target examples from T thanks to M_{ST}, with the intuition that if $\mathbf{x}_t\in T$ belongs to a pair $(\mathbf{x}_t,\mathbf{x}_s)\in M_{ST}$, then $\mathbf{x}_t$ is affected by the true label y_s of $\mathbf{x}_s$. Else, we remove $\mathbf{x}_t$ from T. The self-labeled sample $\widehat{T}$ constructed is:

$$\widehat{T}=\{(\mathbf{x}_t,y_s):(\mathbf{x}_t,\mathbf{x}_s)\in M_{ST},\mathbf{x}_t\in T,(\mathbf{x}_s,y_s)\in S\}.$$

Actually, we restrict the adaptation to region where the source and target marginals coincide under d. Then we provide $\widehat{T}$ to MinCq. Our PV-based self-labeling is illustrated on Figure 1, our framework, called PV-MinCq, is presented in Algorithm 3.

3.3 Analysis of the PV-based self-labeling

In this section, we discuss the impact of our PV-based self-labeling and the choice of the distance d. Given a DA task, we first define the notion of a good distance.

Definition 4. *Given a set of voters $\mathcal{H}$ and $\epsilon>0$, a distance $d:X\times X\to\mathbb{R}^+$ is $\epsilon(\mathcal{H})$-good for the DA task from P_S to P_T, if there exists $\epsilon(\mathcal{H})\geq 0$ such that:*

$$\epsilon(\mathcal{H})=\max_{h\in\mathcal{H},\ (\mathbf{x}_t,\mathbf{x}_s)\sim D_S\times D_T,\ d(\mathbf{x}_t,\mathbf{x}_s)\leq\epsilon}\left|h(\mathbf{x}_s)-h(\mathbf{x}_t)\right|.$$

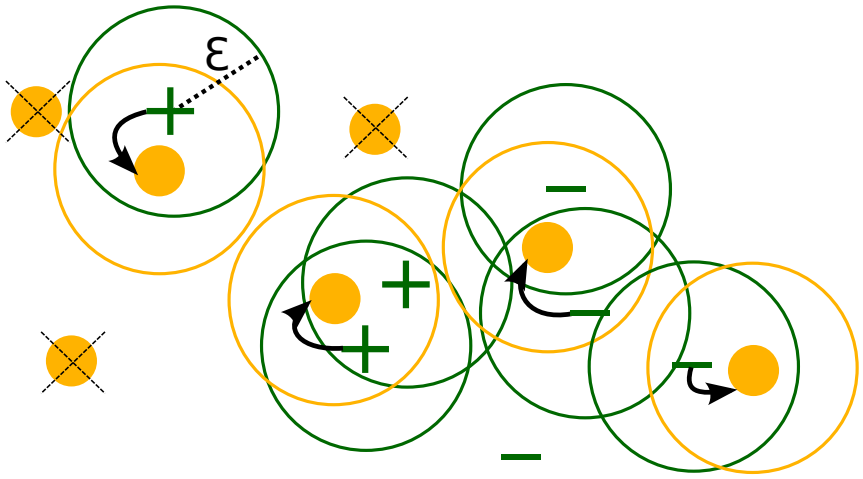

Figure 1: Illustration of the PV-based self-labeling. The labeled source examples are in (dark) green, the unlabeled target examples are in (light) orange. The circles are the candidates for the matching. The arrows correspond to the matched points M_{ST}, and thus to the label transfer: The unmatched target examples are removed from the target sample. Note that the unmatched source and target samples indicate the PV.

Put into words, we want the following natural property: If $\mathbf{x}_t$ and $\mathbf{x}_s$ are close under d, then for every voters in $\mathcal{H}$ the deviation between the returned values $h(\mathbf{x}_s)$ and $h(\mathbf{x}_t)$ is low.

Given a set of voters $\mathcal{H}$, a fixed $\epsilon > 0$ and a $\epsilon(\mathcal{H})$-good distance $d : X \times X \to \mathbb{R}^+$, we consider the matching M_{ST} computed at step 2 of Algorithm 2. By definition, for every $(\mathbf{x}_t, \mathbf{x}_s) \in M_{ST}$, $\mathbf{x}_t$ and $\mathbf{x}_s$ share the same label y_s and we have $d(\mathbf{x}_t, \mathbf{x}_s) \leq \epsilon$. We now study the influence of d and $\epsilon(\mathcal{H})$ on the PAC-Bayesian DA bound of Theorem 1 restricted to M_{ST}. We need of the following notations. The source and the target subsamples associated to M_{ST} are respectively:

$$\widehat{S} = \{(\mathbf{x}_s, y_s) : (\mathbf{x}_t, \mathbf{x}_s) \in M_{ST}, \mathbf{x}_t \in T, (\mathbf{x}_s, y_s) \in S\},$$
$$\widehat{T} = \{(\mathbf{x}_t, y_s) : (\mathbf{x}_t, \mathbf{x}_s) \in M_{ST}, \mathbf{x}_t \in T, (\mathbf{x}_s, y_s) \in S\}.$$

Firstly, we bound the deviation between the risks of G_ρ on $\widehat{S}$ and $\widehat{T}$. For all ρ on $\mathcal{H}$, for every pair $(\mathbf{x}_t, \mathbf{x}_s) \in M_{ST}$ we have:

$$\left|\tfrac{1}{2}\left(1 - y_s \mathop{\mathbf{E}}_{h\sim\rho} h(\mathbf{x}_t)\right) - \tfrac{1}{2}\left(1 - y_s \mathop{\mathbf{E}}_{h\sim\rho} h(\mathbf{x}_s)\right)\right| = \tfrac{1}{2}\left|\mathop{\mathbf{E}}_{h\sim\rho}(h(\mathbf{x}_t) - h(\mathbf{x}_s))\right|$$
$$\leq \tfrac{1}{2}\mathop{\mathbf{E}}_{h\sim\rho}|h(\mathbf{x}_t) - h(\mathbf{x}_s)| = \tfrac{1}{2}\mathop{\mathbf{E}}_{h\sim\rho}\epsilon(\mathcal{H}) = \tfrac{1}{2}\epsilon(\mathcal{H}).$$

Then, we have: $\left|\mathbf{R}_{\widehat{T}}(G_\rho) - \mathbf{R}_{\widehat{S}}(G_\rho)\right| \leq \frac{1}{2}\epsilon(\mathcal{H})$.

Thus the empirical risk of the Gibbs classifier on the source subsample $\widehat{S}$ and the one on the self-labeled target sample $\widehat{T}$ differ at most by $\frac{1}{2}\epsilon(\mathcal{H})$. Hence the lower $\epsilon(\mathcal{H})$, the closer the risks are, and minimizing $\mathbf{R}_{\widehat{T}}(G_\rho)$ is equivalent to minimize $\mathbf{R}_{\widehat{S}}(G_\rho)$.

Secondly, similarly to the risks, we can bound the deviation between the voters' disagreement on $\widehat{S}$ and $\widehat{T}$. For every ρ on $\mathcal{H}$ and for every $(\mathbf{x}_t, \mathbf{x}_s) \in M_{ST}$, we have:

$$\left|\mathop{\mathbf{E}}_{(h,h')\sim\rho^2}[h(\mathbf{x}_s)h'(\mathbf{x}_s) - h(\mathbf{x}_t)h'(\mathbf{x}_t)]\right|$$
$$\leq \left|\mathop{\mathbf{E}}_{(h,h')\sim\rho^2}\Big[(\epsilon(\mathcal{H}) + h(\mathbf{x}_t))(\epsilon(\mathcal{H}) + h'(\mathbf{x}_t)) - h(\mathbf{x}_t)h'(\mathbf{x}_t)\Big]\right|$$
$$= \left|\epsilon(\mathcal{H})^2 + 2\mathop{\mathbf{E}}_{h\sim\rho}\epsilon(\mathcal{H})h(\mathbf{x}_t)\right|$$

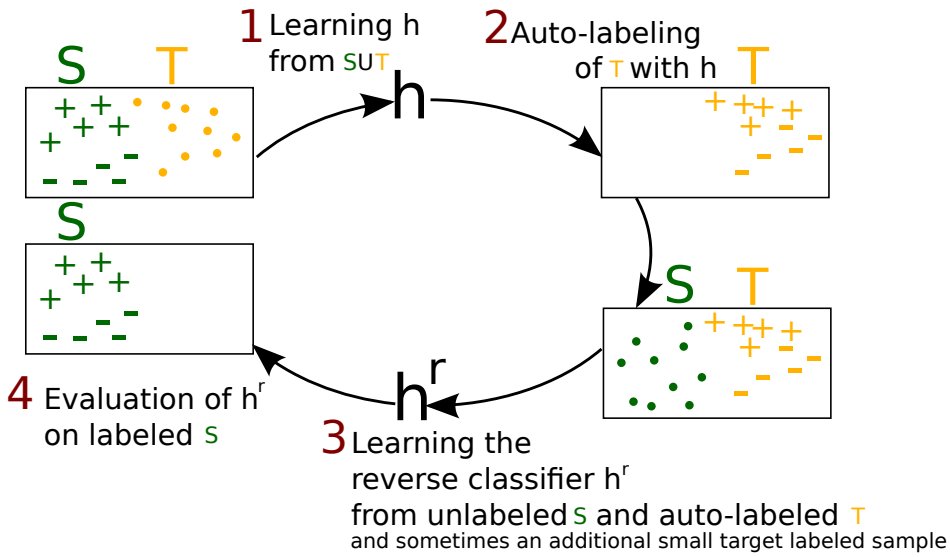

Figure 2: The principle of the reverse validation.

Then, the empirical domain disagreement between $\widehat{S}$ and $\widehat{T}$ can be rewritten by:

$$\begin{aligned}
\mathrm{dis}_\rho(\widehat{S},\widehat{T}) &= \left| \mathop{\mathbf{E}}_{(h,h')\sim\rho^2} \mathop{\mathbf{E}}_{(\mathbf{x}_t,\mathbf{x}_s)\in M_{ST}} [h(\mathbf{x}_s)h'(\mathbf{x}_s) - h(\mathbf{x}_t)h'(\mathbf{x}_t)] \right| \\
&\leq \mathop{\mathbf{E}}_{(\mathbf{x}_t,\mathbf{x}_s)\in M_{ST}} \left| \mathop{\mathbf{E}}_{(h,h')\sim\rho^2} [h(\mathbf{x}_s)h'(\mathbf{x}_s) - h(\mathbf{x}_t)h'(\mathbf{x}_t)] \right| \\
&\leq \mathop{\mathbf{E}}_{(\mathbf{x}_t)\in\widehat{T}} \left| \epsilon(\mathcal{H})^2 + 2 \mathop{\mathbf{E}}_{h\sim\rho} \epsilon(\mathcal{H}) h(\mathbf{x}_t) \right| \\
&\leq \epsilon(\mathcal{H}) \left(1 + 2 \mathop{\mathbf{E}}_{(\mathbf{x}_t)\in\widehat{T}} \left| \mathop{\mathbf{E}}_{h\sim\rho} h(\mathbf{x}_t) \right| \right)
\end{aligned}$$

In this situation, the divergence $\mathrm{dis}_\rho(\widehat{S},\widehat{T})$ between the two samples can be bounded by a term depending on the confidence of the majority vote over $\widehat{T}$ and on $\epsilon(\mathcal{H})$.
These results suggest that we have to minimize $\epsilon(\mathcal{H})$, while keeping good performances on $\widehat{T}$. This confirms the legitimacy of our framework which (i) transfers labels from the source sample to the target one in order to move closer the source and target risks of the Gibbs classifier, (ii) and then applies MinCq for optimizing the voters disagreement on the target sample (given a fixed Gibbs classifier risk on the self-labels). However, although we can choose ϵ (and thus $\epsilon(\mathcal{H})$) as small as we desire, a low ϵ implies a smaller matching M_{ST} and an higher empirical associated PV. In this case, the size of $\widehat{T}$ tends to decrease, and then the guarantees of the Gibbs classifier decreases. In order to avoid this behavior, we exploit this property in the next section for designing a way to tune the hyperparameters.

3.4 Validation of the hyperparameters

A last question concerns the selection of the hyperparameters μ and ϵ. Usually in DA, one can make use of a reverse/circular validation (Bruzzone and Marconcini, 2010; Zhong et al., 2010), with the idea that if the domains are close/related then a reverse classifier, learned from the target data labeled with the current classifier, has to perform well on the source data (see Figure 2 for the intuition). However, for PV-MinCq our first step is to transfer the source labels. As we have seen previously, our main goal is then to validate this transfer, and the reverse validation appears less relevant. We thus propose to deal with our analysis by making use of all the available information, *i.e* the original samples S and T. In this context, on the one hand, since we have shown that the domain disagreement can be upper-bounded by a term depending on the PV-based self-labeling (and the confidence of the majority vote on the target self-labels), the PV between D_S and D_T has to be controlled: The lower the PV, the more similar the samples are. However, minimizing the PV regarding to ϵ can be easy: It is possible to find a high[9] value

[9] *e.g.*, if ϵ equals to the highest distance between source and target example.

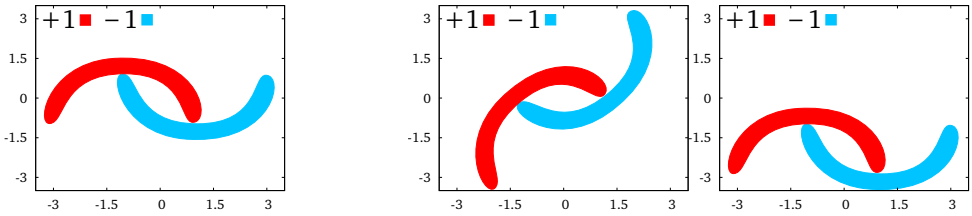

Figure 3: On the left: the source domain. On the right: a target domain with a 40° rotation angle, and the translated target domain.

for ϵ, leading to a small PV. On the other hand, to compensate this behavior, we thus have to control the performance. Indeed, the higher ϵ is, the higher the distance between source and target examples of the same pair (from M_{ST}) is. This implies that these points are less similar, which tends to increase the deviation between the source and target risks, and could imply a loss of performances on the original source sample. Therefore, a relevant PV-based self-labeling corresponds to the one enable to optimize the following trade-off:

$$\mathbf{R}_S(B_\rho) + \widehat{PV}(S, T, \epsilon, d),$$

where $\mathbf{R}_S(B_\rho)$ is the empirical risk on the source sample, and $\widehat{PV}(S, T, \epsilon, d)$ is the empirical PV between S and T. It is worth noting that this process can also be seen in connection with the philosophy of DA: We want to minimize the divergence between the domains while keeping good source performance.

Concretely, for every set of possible parameters (μ, ϵ) and given k-folds on the source sample ($S = \cup_{i=1}^{k} S_i$), PV-MinCq learns a majority vote B_ρ from the $k-1$ labeled folds of S (and T). Then, we evaluate B_ρ on the last k^{th} fold. Its empirical risk corresponds then to the mean of the error over the k-folds:

$$\mathbf{R}_S(B_\rho) = \frac{1}{k} \sum_{i=1}^{k} \mathbf{R}_{S_i}(B_\rho),$$

and $\widehat{PV}(S, T, \epsilon, d)$ is computed by Algorithm 2.

4 Experimental results

In this section, we evaluate our framework PV-MinCq for learning a vote over a set of Gaussian kernels defined from the learning sample. We compare it to the following methods:

- SVM only from the source sample, *i.e.* without adaptation;
- MinCq (Laviolette et al., 2011) only from the source sample;
- TSVM, the semi-supervised transductive-SVM[10], (Joachims, 1999) learns from the two domains;
- DASVM (Bruzzone and Marconcini, 2010), an iterative self-labeling DA algorithm;
- DASF (Morvant et al., 2012), a DA algorithm minimizing a trade-off between a divergence and a source risk based on the analysis of Ben-David et al. (2007);
- PBDA (Germain et al., 2013), the PAC-Bayesian DA algorithm for minimizing the bound of Theorem 1;
- PV-SVM, for which we compute the self-labeling of the target data as for PV-MinCq, and then we apply a classical SVM on these self-labeled data;
- NN-MinCq that uses a k-NN based self-labeling: We label a target point with a k-NN classifier of which the prototypes comes from the source sample (k is tuned).

To compute the PV-based self-labeling, we make use of the euclidean distance. Each parameter is selected with a grid search via a classical 5-folds cross-validation for SVM, MinCq and TSVM, a reverse 5-folds cross-validation for DASVM, DASF, PBDA and NN-MinCq, and the PV-based validation procedure described in Section 3.4 for PV-SVM and PV-MinCq.

We tackle the binary classification task called "inter-twinning moon". The source domain is problem where each moon corresponds to one label (see Figure 3). We consider seven different target domains by rotating anticlockwise the source one according to seven rotations angles from 20° to 80°. The higher the angle, the more difficult the adaptation is. We also consider one target domain as a translation of the source one. We randomly generate 150 positives examples and 150 negatives examples for each domain. To estimate the generalization error of our approach, each algorithm is evaluated on an independent test set of 1,500 target instances. Each DA task is repeated ten times. We report the average correct classification

[10] TSVM has not been proposed for DA scenario, but provides in general very interesting results in DA.

Table 1: Average accuracy results on ten runs for the seven rotations, and for the translation (*trans.*). NN-MinCq implies no result for the latter since, in this case, the self-label is the same for all target examples.

Rot. angle	20°	30°	40°	50°	60°	70°	80°	*trans.*
SVM	89.6	76	68.8	60	47.2	26.1	19.2	50.6
MinCq	92.1	78.2	69.8	61	50.1	40.7	32.7	50.7
TSVM	**100**	78.9	74.6	70.9	64.7	21.3	18.9	94.9
DASVM	**100**	78.4	71.6	66.6	61.6	25.3	21.1	50.1
PBDA	90.6	89.7	77.5	58.8	42.4	37.4	39.6	85.9
DASF	98.3	92.1	83.9	70.2	54.7	43	38.9	82.8
PV-SVM	94.2	82.5	75.1	67.7	55.2	43.6	30.3	97.1
NN-MinCq	97.7	83.7	77.7	69.2	58.1	47.9	42.1	∅
PV-MinCq	99.9	**99.7**	**99**	**91.6**	**75.3**	**66.2**	**58.9**	**97.4**

percentage on Table 1. We make the following remarks.

First, PV-MinCq outperforms on average the others, and appears more robust to change of density (NN-MinCq and MinCq appears also more robust). We also observe that SVM, respectively PV-SVM, provides lower performance than MinCq, respectively PV-MinCq. These observations confirm the necessity of taking into account the voters' disagreement. Second, the PV-based labeling implies better results than the NN one. For the translation task the labels affected by the NN-based self-labeling are the same for every target example. Unlike a NN-based labeling, using the matching implied by the PV appears to be a colloquial way to control the divergence between domains since it clearly focuses on the highest density region by removing the target points without matched source point, in other words on regions where the domains are close. These results confirm that the PV coupled with MinCq provides a nice solution to tackle DA for learning a target majority vote.

5 Discussion and future work

We design a general PAC-Bayesian domain adaptation (DA) framework—PV-MinCq—for learning a target weighted majority vote over a set of real-valued functions. To do so, PV-MinCq is based on MinCq, a quadratic program for minimizing the C-bound over the majority vote's risk by controlling the disagreement between voters known to be crucial in DA. The idea is to focus on regions where the marginals are closer in order to transfer the source labels to the unlabeled target examples (only in these regions). Then we apply MinCq on these self-labeled points, justified by a new version of the C-bound formulated to deal with self-labeling functions. We propose a the self-labeling process which has the originality to be defined thanks to the perturbed variation (PV) between the source and target marginals. Moreover, it has the clear advantage to be non-iterative, unlike usual self-labeling DA algorithms that are generally based on iterative procedures. As a consequence, PV-MinCq is easier to apply. Subsequently, we highlight the necessity of controlling the trade-off between low empirical PV and low source risk, that leads to an original hyperparameters selection. Finally, the empirical results are promising, and raise to exciting directions.

For instance, PV-MinCq could be useful for efficiently combining several data descriptions such as in multiview or multimodality learning[11]. Indeed, in such a situation one natural solution consists in (i) learning a classifier from each description, and (ii) learning[12] a majority vote over the learned classifiers. Thus, for adapting a majority vote from a source corpus to a target one, (ii) can be performed by PV-MinCq.

Given a DA task, another interesting direction is the design (or the learning) of a $\epsilon(\mathcal{H})$-good distance (or metric) d to provide a specific self-labeling, allowing a more accurate computation of the PV. Indeed, our analysis of the self-labeling suggests the requirement of a distance d implying a pertinent measure of closeness in the semantic space involved by the voters.

Lastly, our results raise the question of the usefulness of the PV to learn shared features or points across domains (*e.g.* as done in (Gong et al., 2013; Lin et al., 2013)) by identifying which source samples

[11] *e.g.* a document can be represented by different descriptors.
[12] Sometimes referred as stacking or classifier fusion.

are relevant for the target task.

References

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J., 2010. A theory of learning from different domains. Machine Learning Journal 79, 151–175.

Ben-David, S., Blitzer, J., Crammer, K., Pereira, F., 2007. Analysis of representations for domain adaptation., in: Proceedings of Conference on Neural Information Processing Systems, pp. 137–144.

Ben-David, S., Urner, R., 2012. On the hardness of domain adaptation and the utility of unlabeled target samples, in: Proceedings of Algorithmic Learning Theory, pp. 139–153.

Bruzzone, L., Marconcini, M., 2010. Domain adaptation problems: A DASVM classification technique and a circular validation strategy. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 770–787.

Catoni, O., 2007. PAC-Bayesian supervised classification: the thermodynamics of statistical learning. volume 56. Institute of Mathematical Statistic.

Germain, P., Habrard, A., Laviolette, F., Morvant, E., 2013. PAC-Bayesian domain adaptation bound with specialization to linear classifiers, in: Proceedings of International Conference on Machine Learning.

Germain, P., Lacasse, A., Laviolette, F., Marchand, M., 2009. PAC-Bayesian learning of linear classifiers, in: Proceedings of International Conference on Machine Learning.

Gong, B., Grauman, K., Sha, F., 2013. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation, in: Proceedings of International Conference on Machine Learning.

Harel, M., Mannor, S., 2012. The Perturbed Variation, in: Proceedings of Conference on Neural Information Processing Systems, pp. 1943–1951.

Huang, J., Smola, A.J., Gretton, A., Borgwardt, K.M., Scholkopf, B., 2007. Correcting sample selection bias by unlabeled data. Advances in neural information processing systems 19, 601.

Joachims, T., 1999. Transductive inference for text classification using support vector machines, in: Proceedings of International Conference on Machine Learning, pp. 200–209.

Lacasse, A., Laviolette, F., Marchand, M., Germain, P., Usunier, N., 2007. PAC-Bayes bounds for the risk of the majority vote and the variance of the Gibbs classifier, in: Proceedings of Conference on Neural Information Processing Systems.

Langford, J., 2005. Tutorial on practical prediction theory for classification. Journal of Machine Learning Research 6, 273–306.

Laviolette, F., Marchand, M., Roy, J.F., 2011. From PAC-Bayes bounds to quadratic programs for majority votes, in: Proceedings of International Conference on Machine Learning.

Lin, D., An, X., Zhang, J., 2013. Double-bootstrapping source data selection for instance-based transfer learning. Pattern Recognition Letters 34, 1279–1285.

Mansour, Y., Mohri, M., Rostamizadeh, A., 2008. Domain adaptation with multiple sources, in: Proceedings of Conference on Neural Information Processing Systems, pp. 1041–1048.

Margolis, A., 2011. A Literature Review of Domain Adaptation with Unlabeled Data. Technical Report. University of Washington.

McAllester, D.A., 1999. PAC-Bayesian model averaging, in: Proceedings of conference on Computational learning theory, pp. 164–170.

McAllester, D.A., 2003. Simplified PAC-Bayesian margin bounds, in: Proceedings of Conference on Computational Learning Theory, pp. 203–215.

Morvant, E., Habrard, A., Ayache, S., 2012. Parsimonious unsupervised and semi-supervised domain adaptation with good similarity functions. Knowledge and Information Systems 33, 309–349.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. Knowledge and Data Engineering, IEEE Transactions on 22, 1345–1359.

Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N., 2009. Dataset Shift in Machine Learning. MIT Press.

Seeger, M., 2002. PAC-Bayesian generalization error bounds for Gaussian process classification. Journal of Machine Learning Research 3, 233–269.

Zhong, E., Fan, W., Yang, Q., Verscheure, O., Ren, J., 2010. Cross validation framework to choose amongst models and datasets for transfer learning, in: Proceedings of European Conference on Machine Learning and Principles of Data Mining and Knowledge Discovery, Springer. pp. 547–562.