Domain Adaptation Problems: A DASVM Classification Technique and a Circular Validation Strategy

Lorenzo Bruzzone, *Fellow*, *IEEE*, and Mattia Marconcini, *Member*, *IEEE*

Abstract—This paper addresses pattern classification in the framework of domain adaptation by considering methods that solve problems in which training data are assumed to be available only for a source domain different (even if related) from the target domain of (unlabeled) test data. Two main novel contributions are proposed: 1) a domain adaptation support vector machine (DASVM) technique which extends the formulation of support vector machines (SVMs) to the domain adaptation framework and 2) a circular indirect accuracy assessment strategy for validating the learning of domain adaptation classifiers when no true labels for the target-domain instances are available. Experimental results, obtained on a series of two-dimensional toy problems and on two real data sets related to brain computer interface and remote sensing applications, confirmed the effectiveness and the reliability of both the DASVM technique and the proposed circular validation strategy.

Index Terms—Domain adaptation, transfer learning, semi-supervised learning, support vector machines, accuracy assessment, validation strategy.

1 Introduction

The complexity of pattern classification problems depends on both the investigated application and the available prior information. Two main families of learning methods can be used for training a classifier: *supervised learning* methods (when labeled training samples are given) or *unsupervised learning* methods (when labeled training samples are not available). Let us define a domain D as a distribution $P(\mathbf{x}, y)$, $\mathbf{x} \in \tilde{\mathcal{X}}$, $y \in \Omega$, which governs the classification problem under investigation, where $\tilde{\mathcal{X}}$ and Ω represent all possible instances and all possible information classes for the considered problem, respectively. In the supervised learning setting, classification algorithms are designed under the hypothesis that the distribution $\hat{P}(\mathbf{x}, y)$ estimated from available labeled training data $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_i$, $\mathbf{x}_i \in \mathcal{X} \subset \tilde{\mathcal{X}}$, $y_i \in \Omega$ drawn from D well approximates $P(\mathbf{x}, y)$. Hence, it is possible to obtain high classification accuracies over unseen test data drawn from the same domain. In the unsupervised learning setting, no training data are available. Thus, the problem can be addressed only through clustering methods. However, in many operational applications, there are hybrid situations where, even if prior information is available, it is not sufficient to define a training set representative of the distribution to which the trained model should be applied. These kinds of problems can be addressed according to *transfer learning* methods.

Transfer learning refers to the problem of retaining and applying the knowledge available for one or more tasks, domains, or distributions to efficiently develop an effective hypothesis for a new task, domain, or distribution. Instead of involving generalization across problem instances, transfer learning emphasizes the transfer of knowledge across tasks, domains, and distributions that are similar but not the same. When the objective is to transfer knowledge across different tasks, this results in the *multitask learning* subproblem. Multitask learning methods aim at improving the generalization capability by exploiting the information contained in training data available for the considered tasks (where the set of considered information classes is allowed to vary). In particular, what is learned for each task is used as a bias for other tasks in order to improve the classification performances [1], [2], [3]. In the single-task framework, the default assumption of supervised learning methods is that training and test data are drawn from the same distribution. When the two distributions do not match, two distinct transfer learning subproblems can be defined depending on whether training and test data refer to the same domain or not: 1) learning under *sample selection bias* and 2) learning under *domain adaptation*.

In the case of *sample selection bias*, unlabeled test data are drawn from the same domain D of training data, but the estimated distribution $\hat{P}(\mathbf{x}, y) = \hat{P}(\mathbf{x})\hat{P}(y \mid \mathbf{x})$ does not correctly model the true underlying distribution that governs D since the number (or the quality) of available training samples is not sufficient for an adequate learning of the classifier. The small amount of labeled data generally leads to a poor estimation $\hat{P}(\mathbf{x})$ of the prior distribution $P(\mathbf{x})$ (i.e., $\hat{P}(\mathbf{x}) \neq P(\mathbf{x})$). Moreover, if the few available training data do not represent the general target population and introduce a bias in the estimated class prior distribution (i.e., $\hat{P}(y) \neq P(y)$), this may cause a poor estimation of the

- *The authors are with the Department of Information Engineering and Computer Science, University of Trento, 38050, Povo, Trento, Italy. E-mail: lorenzo.bruzzone@ing.unitn.it, mattia.marconcini@gmail.com.*

Manuscript received 16 Jan. 2008; revised 19 Feb. 2009; accepted 2 Mar. 2009; published online 10 Mar. 2009.
Recommended for acceptance by A. Smola.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-01-0033.
Digital Object Identifier no. 10.1109/TPAMI.2009.57.

0162-8828/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

TABLE 1
Taxonomy of Learning Types and Problems
($\tilde{\mathcal{X}}$ and Ω Represent All Possible Instances and All Possible Information Classes, Respectively, for the Considered Problem)

TYPE OF LEARNING		HYPOTHESES	OBJECTIVE
Supervised Learning		o Labeled training data $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$, $\mathbf{x}_i\in\mathcal{X}\subset\tilde{\mathcal{X}}$, $y_i\in\Omega$, are drawn from domain D ; o Unlabeled test data $\mathcal{X}'=\{\mathbf{x}_i\}_i$, $\mathcal{X}'\subset\tilde{\mathcal{X}}$, are drawn from the same domain D of training data; o It is possible to estimate a distribution $\hat{P}(\mathbf{x},y)$ from $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$ that correctly models the true distribution $P(\mathbf{x},y)$ governing D .	Infer a good approximation $\hat{P}(\mathbf{x},y)$ for $P(\mathbf{x},y)$ by exploiting labeled training data $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$.
Transfer Learning	Multi-task Learning	o Labeled training data $\mathcal{T}=\mathcal{T}_1\cup\mathcal{T}_2\cup\ldots\cup\mathcal{T}_K$ refer to K related tasks characterizing K different domains, where the generic kth domain D_k is governed by the distribution $P^k(\mathbf{x},y)$, $\mathbf{x}\in\mathcal{X}_k\subset\tilde{\mathcal{X}}$, $y\in\Omega_k\subset\Omega$; o For the kth task, it is not possible to estimate a distribution $\hat{P}^k(\mathbf{x},y)$ from $\mathcal{T}_k=\{(\mathbf{x}_i^k,y_i^k)\}_i$, $\mathbf{x}_i^k\in\mathcal{X}_k\subset\tilde{\mathcal{X}}$, $y_i^k\in\Omega_k$ that correctly models the true distribution $P^k(\mathbf{x},y)$ governing D_k .	Infer a good approximation $\hat{P}^k(\mathbf{x},y)$ for $P^k(\mathbf{x},y)$ by jointly exploiting all labeled training data $\mathcal{T}=\mathcal{T}_1\cup\mathcal{T}_2\cup\ldots\cup\mathcal{T}_K$.
	Learning Under Sample Selection Bias / Covariate Shift	o Labeled training data $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$, $\mathbf{x}_i\in\mathcal{X}\subset\tilde{\mathcal{X}}$, $y_i\in\Omega$, are drawn from domain D ; o Unlabeled test data $\mathcal{X}'=\{\mathbf{x}_i\}_i$, $\mathcal{X}'\subset\tilde{\mathcal{X}}$, are drawn from the same domain D of training data; o It is not possible to estimate a distribution $\hat{P}(\mathbf{x},y)$ from $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$ that correctly models the true distribution $P(\mathbf{x},y)$ governing D . There are two possible cases: - If $\hat{P}(\mathbf{x})\neq P(\mathbf{x})$ and $\hat{P}(y\vert\mathbf{x})\neq P(y\vert\mathbf{x})$ the problem is referred to as *sample selection bias*; - If $\hat{P}(\mathbf{x})\neq P(\mathbf{x})$ and $\hat{P}(y\vert\mathbf{x})\approx P(y\vert\mathbf{x})$ the problem is referred to as *covariate shift*.	Infer a good approximation $\hat{P}(\mathbf{x},y)$ for $P(\mathbf{x},y)$ by jointly exploiting labeled training data $\mathcal{T}=\{(\mathbf{x}_i,y_i)\}_i$ and unlabeled test data $\mathcal{X}'=\{\mathbf{x}_i\}_i$.
	Learning Under Domain Adaptation	o Labeled training data $\mathcal{T}_s=\{(\mathbf{x}_i^s,y_i^s)\}_i$, $\mathbf{x}_i^s\in\mathcal{X}_s\subset\tilde{\mathcal{X}}$, $y_i^s\in\Omega$, are drawn from source domain D_s ; o Unlabeled test data $\mathcal{X}_t=\{\mathbf{x}_i^t\}_i$, $\mathcal{X}_t\subset\tilde{\mathcal{X}}$, are drawn from target domain $\mathrm{D}_t\neq\mathrm{D}_s$; o Distribution $P^s(\mathbf{x},y)=P^s(y\vert\mathbf{x})\cdot P^s(\mathbf{x})$ governing D_s is different yet *correlated*[1] to distribution $P^t(\mathbf{x},y)=P^t(y\vert\mathbf{x})\cdot P^t(\mathbf{x})$ governing D_t .	Infer a good approximation $\hat{P}^t(\mathbf{x},y)$ for $P^t(\mathbf{x},y)$ by jointly exploiting labeled source-domain training data $\mathcal{T}_s=\{(\mathbf{x}_i^s,y_i^s)\}_i$ and unlabeled target-domain test data $\mathcal{X}_t=\{\mathbf{x}_i^t\}_i$.
Unsupervised Learning		o Only unlabeled data $\mathcal{X}'=\{\mathbf{x}_i\}_i$, $\mathcal{X}'\subset\tilde{\mathcal{X}}$, drawn from domain D governed by distribution $P(\mathbf{x},y)$ are available.	Infer a good approximation $\hat{P}(\mathbf{x},y)$ for $P(\mathbf{x},y)$ by exploiting unlabeled data $\mathcal{X}'=\{\mathbf{x}_i\}_i$.

conditional distribution (i.e., $\hat{P}(y \mid \mathbf{x}) \neq P(y \mid \mathbf{x})$). On one hand, if both $\hat{P}(\mathbf{x}) \neq P(\mathbf{x})$ and $\hat{P}(y \mid \mathbf{x}) \neq P(y \mid \mathbf{x})$, the problem is referred to as *sample selection bias* [4], [5], [6]. On the other hand, the particular case where the true and estimated distributions are assumed to differ only via $\hat{P}(\mathbf{x}) \neq P(\mathbf{x})$, but $\hat{P}(y \mid \mathbf{x}) \approx P(y \mid \mathbf{x})$ is denoted by *covariate shift* [7], [8].

In the case of *domain adaptation*, unlabeled test patterns $\mathcal{X}_t = \{\mathbf{x}_i^t\}_i$, $\mathcal{X}_t \subset \tilde{\mathcal{X}}$, are drawn from a target domain D_t different from the source domain D_s of training samples $\mathcal{T}_s = \{(\mathbf{x}_i^s, y_i^s)\}_i, \mathbf{x}_i^s \in \mathcal{X}_s \subset \tilde{\mathcal{X}}, y_i^s \in \Omega$. This may happen when the available labeled data are out of date, whereas the test data are obtained from fast evolving information sources, or when series of data acquired at different times should be classified, but training samples collected only at one time are available. In this context, let $P^s(\mathbf{x}, y) = P^s(y \mid \mathbf{x}) \cdot P^s(\mathbf{x})$ and $P^t(\mathbf{x}, y) = P^t(y \mid \mathbf{x}) \cdot P^t(\mathbf{x})$ be the true underlying distributions for the source and target domains, respectively. The key idea is to infer a good approximation of $P^t(\mathbf{x}, y)$ by exploiting $P^s(\mathbf{x}, y)$. If $P^t(y \mid \mathbf{x})$ deviates from $P^s(y \mid \mathbf{x})$ to a given extent, domain adaptation is necessary. In the framework of domain adaptation, most of the learning methods are inspired by the idea that, although different, the two considered domains are correlated.[1] In particular, it is intuitive to observe that considering the (unlabeled) data of the target domain in the training phase could improve the performances with respect to ignore this information source.[2] Table 1 summarizes the main characteristics of all the aforementioned learning methods.

In the last few years, transfer learning has been recognized as an important topic in the machine learning and pattern recognition community. However, the attention has been focused mainly on developing methodologies for addressing multitask learning or learning under sample selection bias, whereas less attention has been devoted to

[1] The correlation between probability distributions (which allows estimating quantitatively how similar they are) can be empirically evaluated according to some similarity metrics. Hence, two domains are considered *correlated* if the distance between the corresponding underlying distributions is relatively small according to proper metrics (e.g., [9], [10], [11]).

[2] A simple toy example for domain adaptation problems is described and investigated in Section 6, where source-domain samples are distributed according to two intertwining moons associated with two specific information classes, while target-domain samples are obtained by a free rotation of the original source-domain patterns (i.e., due to rotation, source- and target-domain data exhibit different distributions: $P^s(\mathbf{x}) \neq P^t(\mathbf{x})$ and $P^s(y \mid \mathbf{x}) \neq P^t(y \mid \mathbf{x})$).

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

domain adaptation problems. This is due to two main motivations: 1) Domain adaptation is a more critical and challenging problem with respect to the other transfer learning subproblems, as training data are assumed to be available only for a source domain different (even if related) from the target domain of the (unlabeled) test samples, and 2) unlike other problems, in practice, there are no strategies to assess the effectiveness of the classification results using standard statistical validation methods as no labeled samples are assumed to be available for the target domain. Considering the complexity of the problem, the lack of procedures for the accuracy assessment is crucial, and at present, seems to be a major limitation for the development of operational domain adaptation learning methods.

In this paper, we address domain adaptation problems by introducing two main novel contributions: 1) a domain adaptation support vector machine (DASVM) technique that extends support vector machines (SVMs) to the domain adaptation framework by exploiting labeled source-domain data and unlabeled target domain data in the training phase of the algorithm and 2) a circular indirect accuracy assessment strategy for the domain adaptation learning that permits to automatically identify reliable solutions for the target-domain classification problem by only exploiting source-domain labeled samples.

The rationale for developing a domain adaptation technique in the framework of SVMs [12], [13] is due to the effectiveness of this classification methodology that attempts to separate samples belonging to different classes by defining maximum margin hyperplanes [14], [15], [16]. The relevance of SVMs is mainly related to their desirable properties that can be summarized as follows:

1. Empirical effectiveness with respect to other traditional classifiers, which results in relatively high classification accuracies and very good generalization capabilities.
2. Convexity of the objective function used in the learning of the classifier, which results in a unique solution (i.e., the system cannot fall into suboptimal solutions associated with local minima).
3. Possibility of representing the optimization problem in a dual formulation, where only nonzero Lagrange multipliers are necessary for defining the separation hyperplane (sparsity of the solution).
4. Capability of addressing classification problems in which no explicit parametric models on the distribution of information classes are assumed (distribution-free classifier).
5. Possibility of defining nonlinear decision boundaries by implicitly mapping the available observations into a higher dimensional space (i.e., kernel trick).

In the literature, semi-supervised [17], [18], [19] and transductive [20], [21] techniques based on SVMs have been proposed for solving problems under sample selection bias characterized by a large amount of unlabeled data but a reduced number of labeled data.[3] In particular, they try to recover information from the distribution of unlabeled data in the input space in order to improve the final classification performances. Nevertheless, these techniques are designed for handling problems where labeled and unlabeled data come from the same domain; thus, they are ineffective on domain adaptation problems, especially when the source- and target-domain distributions are significantly different. In order to overcome such a drawback, the proposed DASVM technique exploits and extends to domain adaptation problems principles of both transductive SVMs (TSVMs) [20] and progressive transductive SVMs (PTSVMs) [21]. From a general perspective, available labeled data from the source domain D_s are used for determining an initial unreliable solution for the target-domain problem; then, unlabeled samples of the target domain D_t are exploited for properly adjusting the decision function, while labeled samples of D_s are gradually erased. The final classification function is determined only on the basis of *semilabeled* samples, i.e., originally unlabeled target-domain instances that obtain labels during the learning process.

In order to estimate the correctness of the solutions for domain adaptation problems (where no prior information for D_t is available), we propose a novel validation strategy developed under the assumption that there exists an intrinsic structure intimately relating D_s and D_t. Under the hypothesis that data in the two domains do not follow uncorrelated distributions, we assume that it is possible to obtain an indirect evaluation of the reliability of the solution to the investigated target problem. The effectiveness of the solution for the target-domain samples can be inferred at the end of a circular procedure by exploiting available labeled samples (i.e., prior information) related to the source domain D_s.

Experimental results obtained on a series of simulated domain adaptation toy problems and on two real domain adaptation problems defined in the framework of brain computer interface and remote sensing point out the effectiveness and the reliability of both the presented DASVM and the proposed circular validation strategy. It is worth noting that the circular validation strategy is general and can be used with any classification technique applied to domain adaptation problems.

The paper is organized into seven sections. In Section 2, a survey on domain adaptation methods is presented. Section 3 introduces the notation and the assumptions considered in this work. Section 4 presents the proposed DASVM technique. Section 5 describes the circular validation strategy devised for assessing the accuracy of domain adaptation learning algorithms. In Section 6, experimental results are reported and discussed. Finally, Section 7 draws the conclusions of this paper.

2 Related Work

In the last few years, the scientific community has devoted a growing interest to the definition of classification techniques for addressing domain adaptation problems. It is worth mentioning that a series of preliminary algorithms has been developed under the assumption that a small amount of target-domain labeled samples are available in the learning phase, thus violating one of the key hypotheses for domain adaptation. However, the role of these studies

[3] It is worth noting that the objective functions used in the learning of semi-supervised and transductive SVMs are often not convex.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

has been particularly important for later development of domain adaptation classifiers. In the following, we first briefly review some of the most relevant algorithms developed in the aforementioned framework; then, we focus the attention on current state-of-the-art domain adaptation techniques.

2.1 Algorithms Assuming Labeled Data Available for the Target Domain

Most of the techniques presented in this framework have been developed for solving text classification problems. A common approach is to treat source-domain data as prior knowledge and to estimate the target-domain model parameters under such prior distribution. Hwa [22] and Gildea [23] proved that simple techniques based on using adequately selected subsets of source-domain data and parameter pruning can improve the performance on unlabeled target data. In [24], Roark and Bacchiani used source-domain data to construct a Dirichlet prior for MAP estimation of the target domain. In [25], Li and Bilmes proposed an accuracy-regularization objective function, which minimizes the empirical risk on target data while maximizing a Bayesian divergence prior determined on the source-domain data distribution. Another approach proposed in [26] by Chelba and Acero is to use the parameters of the maximum entropy model learned from the source domain as the means of a Gaussian prior when training a new model on target data. A different technique based on the Conditional Expectation Maximization (CEM) algorithm developed in the maximum entropy framework has been presented by Daumè and Marcu in [27]. Unlike the aforementioned techniques that do not consider unlabeled samples of the target domain in the learning phase, in [28], a domain adaptation method that can exploit information intrinsic in unlabeled target-domain data has been presented. Jiang and Zhai proposed a general instance weighting framework that implements several adaptation heuristics: removing misleading training samples in the source domain, assigning more weights to labeled target patterns than labeled source patterns, and augmenting training samples with target samples with predicted labels. Other techniques aim at bridging the gap between source and target distributions by changing data representation. As an example, in [29], Florian et al. developed an algorithm that builds a source-domain model and considers its predictions as features for the target domain. In this context, another interesting approach has been recently proposed in [30], where Daumè presented an algorithm based on the idea of transforming domain adaptation problems into standard supervised learning problems (to which any standard algorithm may be applied) by augmenting the size of the feature space of both source and target data.

2.2 Domain Adaptation Algorithms

At present, several domain adaptation algorithms rely on defining new features for capturing the correspondence between source and target domains [31], [32]. In this way, the two domains appear to have similar distributions, thus enabling effective domain adaptation. Moreover, as often features are correlated, careful feature subsetting could lead to significant accuracy gains [33]. In [31], Blitzer et al. describe a heuristic method for domain adaptation, which exploits unlabeled data from both domains to induce correspondences among features in the two domains. The unlabeled target samples are exploited for inferring a good feature representation, which can be regarded as weighting the features. In [32], rather than choosing a common feature representation heuristically, Ben-David et al. try to directly learn a new representation which minimizes a bound on the target generalization error. The bound is determined both using source-domain labeled samples and source and target-domain unlabeled samples, and it is stated in terms of a representation function designed to minimize domain divergence, as well as classification error. The algorithm aims at jointly minimizing a trade-off between source-target similarity and source-domain training error. In [33], Satpal and Sarawagi present a method for addressing domain adaptation problems that selects a subset of features for which the distance (evaluated in terms of a particular distortion metric) between the source and target distributions is minimized, while maximizing the likelihood of labeled training data.

Other interesting approaches for domain adaptation have been presented by Dai et al. [34] and [35]. In [34], they introduced a naive Bayes algorithm for addressing domain adaptation in the context of text categorization, where the EM algorithm is used to find a locally optimal posterior hypothesis under the target distribution. An initial model based on the source training data is first estimated. Such a model is treated as a poor estimation of the target distribution. The EM algorithm is applied to find a local optimal in the hypothesis space, where the estimation should gradually approach the target distribution. In [35], a co-clustering-based classification algorithm is presented, where co-clustering is used as a bridge to propagate the class structure and knowledge from the source domain to the target domain.

Domain adaptation without labeled target-domain data has also been previously analyzed by the authors in the context of remote sensing image classification [36], [37], [38], [39] for addressing automatic updating of land-cover maps. In [36], a domain adaptation approach is proposed that is able to update the parameters of an already trained parametric maximum-likelihood (ML) classifier on the basis of the distribution of a new image for which no labeled samples are available. In [37], in order to take into account the temporal correlation between images acquired on the same area at different times, the ML-based domain adaptation approach is reformulated in the framework of the Bayesian rule for cascade classification (i.e., the classification process is performed by jointly considering information contained in the source and target domains). The basic idea in both approaches is modeling the observed spaces by a mixture of distributions whose components are estimated by the use of unlabeled target data and according to a proper inference applied to source samples of the reference image. This is achieved by using a specific version of the EM algorithm with finite Gaussian Mixture Models [40]. In [38], domain adaptation approaches based on a multiple-classifier system and a multiple-cascade-classifier system (MCCS) have been defined, respectively. In particular, in [39], the proposed MCCS architecture is composed of an ensemble of classifiers developed in the framework of cascade classification, which is integrated in a multiple-classifier architecture. Both a parametric ML classification approach and a nonparametric radial basis function neural network (RBF-NN) classification technique are used as basic classifiers. In addition, in order to increase both the effectiveness and robustness of the

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

ensemble, hybrid ML and RBF-NN cascade classifiers are defined.

3 Problem Formulation and Assumptions

Given an input space $\tilde{\mathcal{X}}$ and a set of information classes Ω, a classifier is any function $g(\mathbf{x}) : \tilde{\mathcal{X}} \rightarrow \Omega$ that maps instances $\mathbf{x} \in \tilde{\mathcal{X}}$ to information classes. In supervised learning problems, training samples $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_i$, $\mathbf{x}_i \in \mathcal{X} \subset \tilde{\mathcal{X}}$, $y_i \in \Omega$, drawn from the probability distribution $P(\mathbf{x}, y) = P(y \mid \mathbf{x}) \cdot P(\mathbf{x})$ are assumed to be available. Accordingly, the learning problem is to determine a supervised classifier $g(\mathbf{x} \mid \mathcal{T}, \boldsymbol{\theta})$[4] that permits to obtain high predictive accuracy for unlabeled test samples drawn from the same distribution $P(\mathbf{x}, y)$ by exploiting the available training set $\mathcal{T}$. The discrimination capability depends on the classifier model, which is described by a vector of parameters $\boldsymbol{\theta}$ that is specific for each family of classifiers.

In the framework of domain adaptation, the problem is more complex as test patterns are drawn from a target-domain distribution $P^t(\mathbf{x}, y) = P^t(y \mid \mathbf{x}) \cdot P^t(\mathbf{x})$ different from the source-domain distribution of training samples $P^s(\mathbf{x}, y) = P^s(y \mid \mathbf{x}) \cdot P^s(\mathbf{x})$. Obtaining a good adaptation requires an adequate modeling of the relationship between source and target domains D_s and D_t. There are two extreme cases for domain adaptation problems: 1) If $P^s(\mathbf{x}, y) \equiv P^t(\mathbf{x}, y)$, adaptation is not necessary and standard supervised learning algorithms can be employed and 2) if $P^s(\mathbf{x}, y)$ and $P^t(\mathbf{x}, y)$ are uncorrelated, then source-domain data are useless for building a model for D_t. Nevertheless, in real applications, D_s and D_t are generally neither identical nor uncorrelated. In these situations, it is reasonable to assume the existence of an intrinsic relationship between the two domains that makes it possible adaptation. We expect that the probability to succeed in the adaptation process is associated with the complexity of the problem, which depends on the correlation between $P^s(\mathbf{x}, y)$ and $P^t(\mathbf{x}, y)$.

In this context, let us consider two sets $\mathcal{X}_s = \{\mathbf{x}_l^s\}_{l=1}^N$ and $\mathcal{X}_t = \{\mathbf{x}_u^t\}_{u=1}^M$ composed of N source-domain and M target-domain patterns, respectively. Let $\mathbf{x}^s$ and $\mathbf{x}^t$ be the d-dimensional feature vectors related to D_s and D_t, respectively (d represents the dimensionality of the input space). The proposed techniques are formulated under the following assumptions:

- The same set of L classes $\Omega = \{\omega_i\}_{i=1}^L$ characterizes D_s and D_t.
- A set of true labels $\mathcal{Y}_s = \{y_l^s\}_{l=1}^N$ for $\mathcal{X}_s$ is available, thus, it is possible to define a training set $\mathcal{T}_s = \{\mathcal{X}_s, \mathcal{Y}_s\} = \{(\mathbf{x}_l^s, y_l^s)\}_{l=1}^N$ for D_s.
- A set of true labels $\mathcal{Y}_t = \{y_u^t\}_{u=1}^M$ for $\mathcal{X}_t$ is not available, thus, it is not possible to define a training set for D_t.

Under such a hypothesis, our goals are:

1. to define a domain adaptation classifier $g(\mathbf{x} \mid \mathcal{T}_s, \mathcal{X}_t, \psi)$ based on SVMs which permits us to obtain an accurate classification for target-domain samples by exploiting labeled training samples $\mathcal{T}_s$ from D_s and unlabeled samples $\mathcal{X}_t$ from D_t (as for supervised classifiers, the model adopted for classification is described by a vector of parameters ψ, which is specific for each family of domain adaptation classifiers);
2. to develop a strategy for validating the learning of the domain adaptation classifier without labeled target-domain data.

4 Proposed DASVM Technique

In this section, for simplicity, we describe the proposed DASVM technique in the case of a two-class problem. Unlike transductive and semi-supervised SVMs, the DASVM algorithm takes into account that unlabeled target-domain samples are drawn from a distribution $P^t(\mathbf{x}, y)$ different from the one of source-domain training patterns $P^s(\mathbf{x}, y)$. Therefore, source-domain samples are only exploited for initializing the discriminant function for the target-domain problem, while they are successively gradually erased in order to obtain a final separation hyperplane defined only on the basis of target-domain samples. This represents an important conceptual difference with respect to both transductive and semi-supervised SVMs techniques, which recover information from unlabeled samples under the assumption that the labeled and unlabeled samples are drawn from the same domain. Thus, they cannot be used for solving domain adaptation problems. On the contrary, the DASVM technique, by iteratively deleting source-domain samples and adapting the discriminant function step by step to the target-domain instances, can recover useful information and properly seize the target-domain classification problem.

The proposed DASVM algorithm is made up of three main phases: 1) initialization (only $\mathcal{T}_s$ is used for initializing the discriminant function), 2) iterative domain adaptation ($\mathcal{T}_s$ and $\mathcal{X}_t$ are used for gradually adapting the discriminant function to D_t), and 3) convergence (only $\mathcal{X}_t$ is used for defining the final discriminant function). In the following, we will denote by $\mathcal{T}^{(i)}$ and $\mathcal{X}_t^{(i)}$ the training set and the unlabeled set (i.e., the set containing the target-domain samples that have not been inserted into the training set $\mathcal{T}^{(i)}$) at the generic iteration i, respectively. These phases are described in the following.

4.1 Phase 1: Initialization

In the first phase, an initial separation hyperplane is determined on the basis of source-domain training data alone. We have that $\mathcal{T}^{(0)} = \{\mathcal{X}_s, \mathcal{Y}_s\} = \{(\mathbf{x}_l^s, y_l^s)\}_{l=1}^N$ and $\mathcal{X}_t^{(0)} = \{\mathbf{x}_u^t\}_{u=1}^M$. As for standard supervised SVMs, the bound cost function to minimize is the following:

$$\begin{cases} \min\limits_{\mathbf{w},b,\xi}\left\{\dfrac{1}{2}\parallel \mathbf{w}^{(0)} \parallel^2 + C\sum\limits_l \xi_l^s\right\} \\ y_l^s\left(\mathbf{w}^{(0)} \cdot \mathbf{x}_l^s + b^{(0)}\right) \geq 1 - \xi_l^s \quad \forall l = 1, \ldots, N, \ \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{T}^{(0)}, \\ \xi_l^s \geq 0 \end{cases} \tag{1}$$

[4] This notation has been adopted for pointing out all the input variables (data and learning parameters) that affect the output of a classifier.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

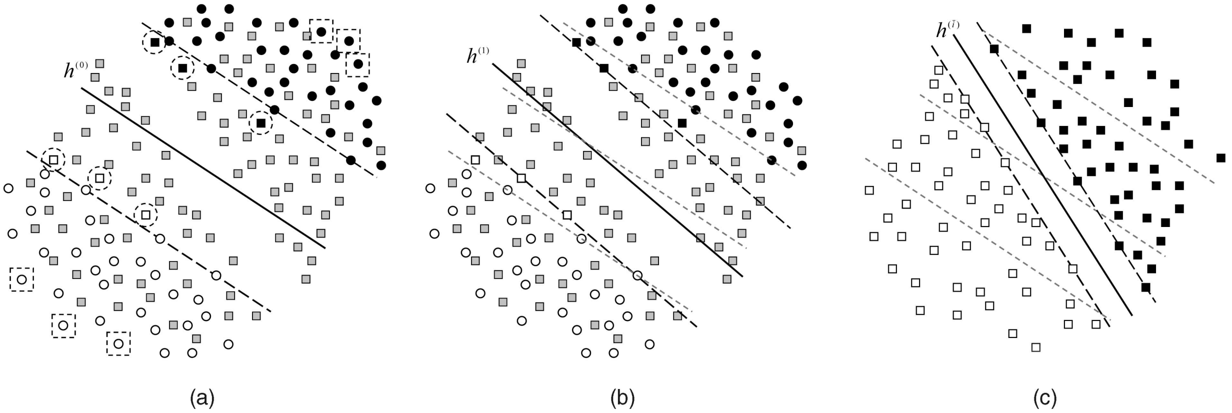


Fig. 1. Separation hyperplane (solid line) and margin bounds (dashed lines) at different stages of the DASVM algorithm for a toy data set. Labeled source-domain patterns are shown as white and black circles. Semilabeled target-domain patterns are shown as white and black squares, respectively. Unlabeled target-domain patterns are represented as gray squares. Feature space structure obtained: (a) at the first iteration (the dashed circles highlight the ρ semilabeled patterns selected from both sides of the margin; in the example $\rho = 3$); (b) at the second iteration and (c) at the last iteration, respectively, in an ideal situation (the dashed gray lines represent both the separation hyperplane and the margin bounds at the beginning of the learning process).

where $\mathbf{w}^{(0)}$ is a vector normal to the separation hyperplane $h^{(0)} : \mathbf{w}^{(0)} \cdot \mathbf{x} + b^{(0)} = 0$, b is a constant such that $b^{(0)}/\|\mathbf{w}^{(0)}\|^2$ represents the distance of the hyperplane from the origin, ξ_i are slack variables, and C is a *penalization parameter* (also called regularization parameter).

4.2 Phase 2: Iterative Domain Adaptation

At the generic iteration i, all the original unlabeled target-domain samples $\mathbf{x}_u^t \in \mathcal{X}_t^{(0)}$ are associated with an estimated label $\hat{y}_u^{t(i)} = \mathrm{sgn}[f^{(i)}(\mathbf{x}_u^t)]$, determined according to the current decision function $f^{(i)}(\mathbf{x}_u^t) = \mathbf{w}^{(i)} \cdot \mathbf{x}_u^t + b^{(i)}$. Then, a subset of the (remaining) unlabeled samples $\mathcal{X}_t^{(i)}$ is iteratively selected and moved (with the corresponding estimated labels) into the training set $\mathcal{T}^{(i+1)}$. On one hand, the higher the distance from the separation hyperplane $h^{(i)} : \mathbf{w}^{(i)} \cdot \mathbf{x} + b^{(i)} = 0$, the higher the probability for an unlabeled sample to be correctly classified. On the other hand, the current unlabeled samples falling into the margin band $\mathcal{M}^{(i)} = \{\mathbf{x} \mid -1 \leq f^{(i)}(\mathbf{x}) \leq 1\}$ are those with the highest probability to be associated with nonzero Lagrange multipliers (and thus, to affect the position of $h^{(i+1)}$) once inserted in $\mathcal{T}^{(i+1)}$ with their current estimated label (patterns falling outside the margin band are more likely to be associated with null multipliers). According to these two observations, at each iteration, we progressively take into account the unlabeled target-domain samples falling into $\mathcal{M}^{(i)}$ closest to the margin bounds. Let us define the following two subsets:

$$\begin{aligned}\mathcal{H}_{up}^{(i)} &= \left\{\left(\mathbf{x}_u^t, \hat{y}_u^{t(i)}\right) \mid \mathbf{x}_u^t \in \mathcal{X}_t^{(i)}, 1 \geq f^{(i)}\left(\mathbf{x}_u^t\right) \geq f^{(i)}\left(\mathbf{x}_{u+1}^t\right) \geq 0\right\},\\ \mathcal{H}_{\mathrm{low}}^{(i)} &= \left\{\left(\mathbf{x}_u^t, \hat{y}_u^{t(i)}\right) \mid \mathbf{x}_u^t \in \mathcal{X}_t^{(i)}, -1 \leq f^{(i)}\left(\mathbf{x}_u^t\right) \leq f^{(i)}\left(\mathbf{x}_{u+1}^t\right) < 0\right\},\end{aligned} \tag{2}$$

where $\mathcal{H}_{\mathrm{up}}^{(i)}$ and $\mathcal{H}_{\mathrm{low}}^{(i)}$ are made up of the patterns of the current unlabeled set $\mathcal{X}_t^{(i)}$ (considered with their corresponding estimated labels) lying in the upper and lower sides of the margin band $\mathcal{M}^{(i)}$, respectively. Samples of $\mathcal{H}_{\mathrm{up}}^{(i)}$ and $\mathcal{H}_{\mathrm{low}}^{(i)}$ are sorted in ascending order with respect to their distance from the upper and lower bound of the margin, respectively. The DASVM approach exploits a strategy inspired by the PTSVM algorithm [21] in which, at each iteration of the learning process, the unlabeled samples inside the margin band $\mathcal{M}^{(i)}$ with the maximum and minimum values of the decision function are moved into the training set. As two patterns may not be sufficiently representative for tuning the position of the hyperplane, in the proposed DASVM, at each iteration, the first ρ patterns (where the parameter $\rho \geq 1$ is defined a priori by the user) belonging to $\mathcal{H}_{\mathrm{up}}^{(i)}$ and to $\mathcal{H}_{\mathrm{low}}^{(i)}$, whose current estimated labels $\hat{y}_u^{t(i)}$ are "+1" and "−1," respectively, are selected and inserted into the training set $\mathcal{T}^{(i)}$ (see Figs. 1a and 1b). Such samples are defined as *semilabeled* patterns. As the cardinality of $\mathcal{H}_{\mathrm{up}}^{(i)}$ and $\mathcal{H}_{\mathrm{low}}^{(i)}$ may be lower than ρ, the subset of target-domain patterns selected at the generic iteration i becomes

$$\begin{aligned}\mathcal{H}^{(i)} = &\left\{\left(\mathbf{x}_u^t, \hat{y}_u^{t(i)}\right) \in \mathcal{H}_{\mathrm{up}}^{(i)} \mid 1 \leq u \leq \lambda^{(i)}\right\}\\ &\cup \left\{\left(\mathbf{x}_u^t, \hat{y}_u^{t(i)}\right) \in \mathcal{H}_{\mathrm{low}}^{(i)} \mid 1 \leq u \leq \delta^{(i)}\right\},\end{aligned} \tag{3}$$

where $\lambda^{(i)} = \min(\rho, |\mathcal{H}_{\mathrm{up}}^{(i)}|)$ and $\delta^{(i)} = \min(\rho, |\mathcal{H}_{\mathrm{low}}^{(i)}|)$. Patterns belonging to $\mathcal{H}^{(i)}$ are then merged with $\mathcal{T}^{(i)}$. A dynamical adjustment is necessary for taking into account that the position of the separation hyperplane $h^{(i)}$ changes at each iteration. Let

$$\mathcal{S}^{(i)} = \left\{(\mathbf{x}_u^t, \hat{y}_u^{t(i-1)}) \in \mathcal{T}^{(i)} \mid \hat{y}_u^{t(i)} \neq \hat{y}_u^{t(i-1)}\right\} \tag{4}$$

represent the set of semilabeled samples belonging to $\mathcal{T}^{(i)}$ whose labels at iteration i are different from those at iteration $i - 1$. If the label of a semilabeled pattern at iteration i is different from the one at iteration $i - 1$ (label inconsistency), such a label is erased and the semilabeled pattern is reset to the unlabeled state and moved to $\mathcal{X}_t^{(i+1)}$. In this way, it is possible to reconsider this pattern in the following iterations of the learning procedure.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

Let $\mathcal{J}^{(i)}$ represent the set containing all the semilabeled patterns at the ith iteration. $\mathcal{J}^{(i)}$ is partitioned into a finite number of subsets $\gamma \in \mathbb{N}_0$,

$$\mathcal{J}^{(i)} = \mathcal{J}_1^{(i)} \cup \mathcal{J}_2^{(i)} \cup \cdots \cup \mathcal{J}_\gamma^{(i)}$$
$$\begin{cases} \mathcal{J}_1^{(i)} = \mathcal{H}^{(i)} \\ \mathcal{J}_k^{(i)} = \mathcal{J}_{k-1}^{(i-1)} - \mathcal{S}^{(i)}, \quad \forall k = 2, \ldots, \gamma - 1 \\ \mathcal{J}_\gamma^{(i)} = \left(\mathcal{J}_\gamma^{(i-1)} \cup \mathcal{J}_{\gamma-1}^{(i-1)}\right) - \mathcal{S}^{(i)}, \end{cases} \tag{5}$$

where each kth subset includes all of the semilabeled samples that do not change their label after the tuning of the separation hyperplane at the ith iteration and that belonged to the subset with index $k-1$ at iteration $i-1$. As will be pointed out in the following, the DASVM algorithm aims at gradually increasing the regularization parameter for the semilabeled patterns according to a time-dependent criterion; accordingly, γ is defined as the maximum number of iterations for which the user allows the regularization parameter for semilabeled samples to increase.

As the main purpose of the proposed technique is to define and solve a bound minimization problem with respect only to the target-domain samples, at each iteration a subset $\mathcal{Q}^{(i)}$ of the original source-domain training patterns is deleted. The higher the distance from the separation hyperplane $h^{(i)}$, the lower the influence in affecting its position. Accordingly, it is reasonable to erase from $\mathcal{T}^{(i)}$ the source-domain samples lying farther from $h^{(i)}$ (see Fig. 1a). Let us define the following two subsets:

$$\begin{aligned} \mathcal{Q}_{\text{up}}^{(i)} &= \left\{ \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{T}^{(i)} \mid f^{(i)}\left(\mathbf{x}_l^s\right) \geq f^{(i)}\left(\mathbf{x}_{l+1}^s\right) \geq 0 \right\}, \\ \mathcal{Q}_{\text{low}}^{(i)} &= \left\{ \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{T}^{(i)} \mid f^{(i)}\left(\mathbf{x}_l^s\right) \leq f^{(i)}\left(\mathbf{x}_{l+1}^s\right) < 0 \right\}, \end{aligned} \tag{6}$$

where $\mathcal{Q}_{\text{up}}^{(i)}$ and $\mathcal{Q}_{\text{low}}^{(i)}$ contain the unlabeled target-domain patterns that lie above and under the separation hyperplane, respectively, sorted in descending order with respect to their distance from $h^{(i)}$. At the ith iteration, the number of patterns to erase from $\mathcal{Q}_{\text{up}}^{(i)}$ and $\mathcal{Q}_{\text{low}}^{(i)}$ is set equal to the number of semilabeled patterns selected from the upper and lower sides of the margin band (i.e., $\lambda^{(i)}$ and $\delta^{(i)}$), respectively. If none of the remaining unlabeled samples fall into the margin band ($\mathcal{H}^{(i)} = \emptyset$), the number of patterns to delete is set to ρ. As a consequence, we have:

$$\begin{aligned} \mathcal{Q}^{(i)} = &\left\{ \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{Q}_{\text{up}}^{(i)} \mid 1 \leq l \leq \nu^{(i)} \right\} \\ &\cup \left\{ \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{Q}_{\text{low}}^{(i)} | 1 \leq l \leq \kappa^{(i)} \right\}, \end{aligned} \tag{7}$$

where

$$\nu^{(i)} = \begin{cases} \min\left(\lambda^{(i)}, \left|\mathcal{Q}_{\text{up}}^{(i)}\right|\right) & \text{if } \mathcal{H}^{(i)} \neq \emptyset \\ \min\left(\rho, \left|\mathcal{Q}_{\text{up}}^{(i)}\right|\right) & \text{if } \mathcal{H}^{(i)} = \emptyset \end{cases} \quad \text{and}$$

$$\kappa^{(i)} = \begin{cases} \min\left(\delta^{(i)}, \left|\mathcal{Q}_{\text{low}}^{(i)}\right|\right) & \text{if } \mathcal{H}^{(i)} \neq \emptyset, \\ \min\left(\rho, \left|\mathcal{Q}_{\text{low}}^{(i)}\right|\right) & \text{if } \mathcal{H}^{(i)} = \emptyset. \end{cases}$$

Let $\mu^{(i)} = |\mathcal{T}^{(i)}| - |\mathcal{J}^{(i-1)}|$ and $\eta^{(i)} = |\mathcal{J}^{(i-1)}|$ represent the number of original source-domain and semilabeled samples belonging to the current training set $\mathcal{T}^{(i)}$, respectively. For $i \geq 1$, the bound minimization problem can be written as

$$\begin{cases} \min_{\mathbf{w}, b, \xi^s, \xi^t} \left\{ \frac{1}{2} \left\|\mathbf{w}^{(i)}\right\|^2 + C^{(i)} \sum_l \xi_l^s + \sum_u C_u^* \xi_u^t \right\} \\ y_l^s \cdot \left(\mathbf{w}^{(i)} \cdot \mathbf{x}_l^s + b^{(i)}\right) \geq 1 - \xi_l^s \\ \qquad \forall l = 1, \ldots, \mu^{(i)}, \left(\mathbf{x}_l^s, y_l^s\right) \in \mathcal{T}^{(i)} \\ \hat{y}_u^{t(i-1)} \cdot \left(\mathbf{w}^{(i)} \cdot \mathbf{x}_u^t + b^{(i)}\right) \geq 1 - \xi_u^t \\ \qquad \forall u = 1, \ldots, \eta^{(i)}, \left(\mathbf{x}_u^t, \hat{y}_u^{t(i-1)}\right) \in \mathcal{T}^{(i)} \\ \xi_l^s, \xi_u^t \geq 0. \end{cases} \tag{8}$$

The semilabeled samples $\left(\mathbf{x}_u^t, \hat{y}_u^{t(i-1)}\right) \in \mathcal{T}^{(i)}$ are associated with a regularization parameter $C_u^* = C_u^*(k) \in \mathbb{R}^+$ that depends on the kth subset $\mathcal{J}_k^{(i-1)}$ which they belong to at iteration $i-1$. The original source-domain patterns, instead, are associated with a regularization parameter $C^{(i)}$ that directly depends on the ith iteration. The purpose of $C^{(i)}$ and C_u^* is to control the number of misclassified samples of the current training set $\mathcal{T}^{(i)}$ drawn from D_s and D_t, respectively. On increasing their values, the penalty associated with errors increases. In other words, the larger the regularization parameter, the higher the influence of the associated samples on the selection of the separation hyperplane. As $P^t(\mathbf{x}, y)$ in general could be rather different compared to $P^s(\mathbf{x}, y)$, unlabeled samples should be considered gradually in the learning process in order to avoid instabilities. For this reason, the algorithm adopts a weighting strategy based on a temporal criterion. The regularization parameter for the semilabeled patterns increases in a quadratic way, depending on the number of iterations k they had last inside the set containing the semilabeled patterns $\mathcal{J}^{(i)}$ (see Fig. 2a):

$$\begin{aligned} &\forall u = 1, \ldots, \left|\mathcal{J}^{(i)}\right|, \\ &C_u^* = \frac{C^{*\max} - C^*}{(\gamma - 1)^2} (k-1)^2 + C^* \Leftrightarrow \left(\mathbf{x}_u^t, \hat{y}_u^{t(i)}\right) \in \mathcal{J}_k^{(i)}, \\ &k = 1, \ldots, \gamma, \end{aligned} \tag{9}$$

where C^* is the initial regularization value for semilabeled samples (this is a user-defined parameter), and $C^{*\max}$ is the maximum cost value of semilabeled samples and is related to that of training patterns (i.e., $C^{*\max} = \tau \cdot C$, $0 < \tau \leq 1$ being a constant; a reasonable choice has proved to be $\tau = 0.5$). The greater k is, the higher the reliability of a semilabeled sample is expected to be.

Likewise, the algorithm makes the cost factor for the original source-domain labeled samples $C^{(i)}$ to decrease in a quadratic way (see Fig. 2b). At the beginning, the position of the separation hyperplane strongly depends on source-domain patterns $(\mathbf{x}_l^s, y_l^s)$, but their influence always gets lower as the number of iterations increases (until $i = \gamma$):

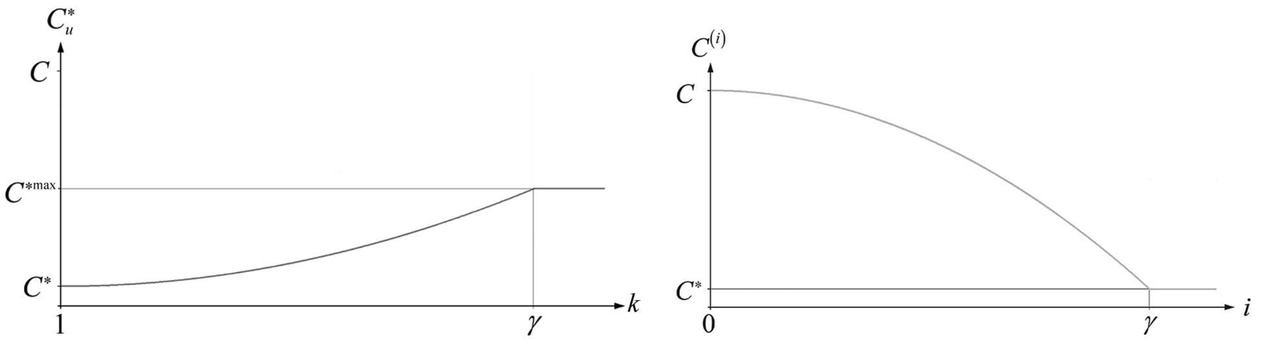
$$C^{(i)} = \max\left(\frac{C^* - C}{\gamma^2} i^2 + C, C^*\right). \tag{10}$$

The second phase ends when the convergence criterion described below is satisfied.

4.3 Phase 3: Convergence

From a theoretical viewpoint, it can be assumed that convergence is reached if none of the remaining target-domain samples lies into the margin band, $\mathcal{H}^{(i)} = \emptyset$, after all of the source-domain labeled samples have been erased, $\mathcal{Q}^{(i)} = \emptyset$ (see Fig. 1c). Nevertheless, such a choice might result in a high computational load. Moreover, it may

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 2. (a) Behavior of C_u^*, regularization parameter for the semilabeled patterns belonging to $\mathcal{J}^{(i)} = \mathcal{J}_1^{(i)} \cup \mathcal{J}_2^{(i)} \cup \ldots \cup \mathcal{J}_\gamma^{(i)}$, versus k (index corresponding to the subset $\mathcal{J}_k^{(i)}$, which is related to the number of iterations in which a semilabeled pattern is associated with the same label). (b) Behavior of the regularization parameter for the original source-domain labeled patterns, $C^{(i)}$, versus the number of iterations i.

happen that even if the margin band is empty, the number of inconsistent patterns is not negligible. For these reasons, the following empirical stopping criterion has been defined:

$$\begin{cases} \mathcal{Q}^{(i)} = \emptyset, \\ |\mathcal{H}^{(i)}| \leq \lceil \beta \cdot M \rceil, \\ |\mathcal{S}^{(i)} \leq \lceil \beta \cdot M \rceil \end{cases} \tag{11}$$

where M is the number of target-domain samples and β is a constant fixed a priori that tunes the sensitivity of the learning process. This means that convergence is reached if both the number of mislabeled and remaining unlabeled patterns lying in the margin band at the current iteration is lower than or equal to $\lceil \beta \cdot M \rceil$. The final minimization problem at the last iteration $\bar{i}$ becomes

$$\begin{cases} \min_{\mathbf{w},b,\xi} \left\{ \frac{1}{2} \parallel \mathbf{w} \parallel^2 + \sum_u C_u^* \xi_u \right\} \\ \hat{y}_u^{t(\bar{i}-1)} \cdot \left(\mathbf{w} \cdot \mathbf{x}_u^t + b \right) \geq 1 - \xi_u \\ \qquad \forall u = 1, \ldots, \left| \mathcal{T}^{(\bar{i})} \right|, \left(\mathbf{x}_u^t, \hat{y}_u^{t(\bar{i}-1)} \right) \in \mathcal{T}^{(\bar{i})} \\ \xi_u \geq 0. \end{cases} \tag{12}$$

At the end, all of the target-domain patterns $\mathbf{x}_u^t \in \mathcal{X}_t$ are labeled according to the resulting separation hyperplane, i.e., $\hat{\mathcal{Y}}_t = \{\hat{y}_u^t = \text{sgn}[\mathbf{w} \cdot \mathbf{x}_u^t + b]\}_{u=1}^M$.

The above-described algorithm is defined for two-class problems. When a multiclass problem has to be investigated, a One-Against-All (OAA) strategy [41] can be employed.

5 Proposed Circular Validation Strategy

In this section, we present a novel general empirical strategy for validating the solutions obtained with a domain adaptation classifier when no labeled data related to the target domain are available. This method can also be used with the DASVM presented in the previous section.

5.1 Background and Rationale of the Proposed Strategy

The proposed strategy is based on the two following assumptions.

1. *Assumption 1*: Under the assumption that $P^s(\mathbf{x}, y)$ and $P^t(\mathbf{x}, y)$ are neither uncorrelated nor identical, it is reasonable to assume the existence of an intrinsic relationship between solutions that are satisfactory for the two domains. This relationship is associated with the intrinsic structure of the considered problem (which is related to the correlation between $P^s(\mathbf{x}, y)$ and $P^t(\mathbf{x}, y)$). We expect that an adequately trained domain adaptation algorithm seizes the structure of the problem and can move from modeling the source-domain problem to modeling the target-domain problem, and vice versa. On the contrary, if the learning process fails, the considered domain adaptation algorithm completely loses the structure of the problem and results in an unpredictable behavior that involves a random solution. This solution has no relation to the considered problem. In this condition, it is no more possible to recover the intrinsic structure of the problem, and thus, moving from modeling the target-domain problem to modeling the source-domain problem.
2. *Assumption 2:* The only labeled samples available are those related to the source domain D_s. Thus, for validating the learning for D_t, we should devise an indirect procedure based on the training set $\mathcal{T}_s$.

On the basis of these observations, the proposed strategy relies on the following rationale: Let us consider that, starting from a reliable estimated distribution $\hat{P}_n^s(\mathbf{x}, y)$ for D_s (and thus, from an acceptable classification accuracy on source-domain patterns), the nth generic domain adaptation classifier results in an accurate estimate $\hat{P}_n^t(\mathbf{x}, y)$ for $P^t(\mathbf{x}, y)$ (and hence, in a satisfactory classification accuracy for the instances related to D_t). In such a case, the domain adaptation classifier seizes the structure of the target-domain problem. In this condition, we assume that, by again applying the same learning algorithm in the reverse sense (using the classification labels in place of the missing prior knowledge for target-domain patterns $\mathcal{X}_t$, keeping the same learning parameters, and considering the problem of classifying source-domain patterns $\mathcal{X}_s$), it is possible to infer an accurate estimate for $P^s(\mathbf{x}, y)$ (thus obtaining a good discrimination capability also for the source-domain problem). On the contrary, if the domain adaptation algorithm does not identify an acceptable solution for D_t, this means that it does not capture the relationship between the two domains but converges to a solution which is not related to the investigated problem (i.e., the resulting $\hat{P}_n^t(\mathbf{x}, y)$ does not

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

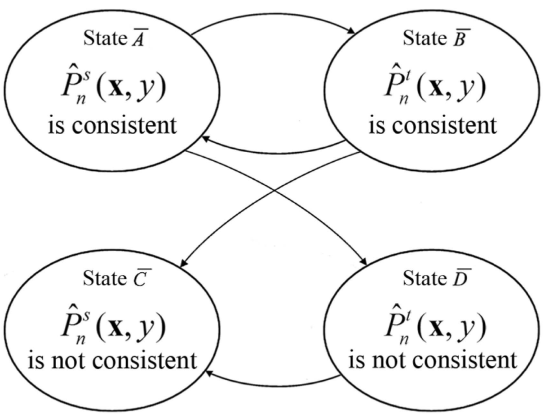


Fig. 3. Diagram of all the possible state transitions exploited from the proposed circular validation strategy.

represent an adequate estimation of the real distribution $P^t(\mathbf{x}, y)$). In this condition, by again applying the algorithm in the reverse sense (from target to source), it seems impossible to recover a reliable solution for the source-domain problem, but rather, it is reasonable to expect a poor estimate for $P^s(\mathbf{x}, y)$. This reasoning has some analogies with the definition of specific trajectories that model transitions between different states in chaotic systems. On the basis of these expected properties, we can use the accuracy evaluated on the original training samples from D_s for validating the solution obtained for target-domain instances after a circular (forward and backward) application of the considered domain adaptation algorithm.

5.2 Formulation of the Proposed Circular Validation Strategy

Given a classification accuracy measure $\Lambda(\mathcal{Y}_j, \hat{\mathcal{Y}}_{jn})$ that evaluates the similarity between a set of labels $\hat{\mathcal{Y}}_{jn}$ (i.e., a solution) predicted by the generic classifier $\mathbf{g}_n(\mathbf{x})$ and the corresponding set of true labels $\mathcal{Y}_j$, and given a threshold Λ_{th} for Λ, let us define the following four sets of classifiers (see Fig. 3):

$$\mathcal{A} = \{g_n(\mathbf{x}) \mid \Lambda(\mathcal{Y}_s, \hat{\mathcal{Y}}_{sn}) \geq \Lambda_{\mathrm{th}}\}, \tag{13}$$

$$\mathcal{B} = \{g_n(\mathbf{x}) \mid \Lambda(\mathcal{Y}_t, \hat{\mathcal{Y}}_{tn}) \geq \Lambda_{\mathrm{th}}\}, \tag{14}$$

$$\mathcal{C} = \{g_n(\mathbf{x}) \mid \Lambda(\mathcal{Y}_s, \hat{\mathcal{Y}}_{sn}) < \Lambda_{\mathrm{th}}\}, \tag{15}$$

$$\mathcal{D} = \{g_n(\mathbf{x}) \mid \Lambda(\mathcal{Y}_t, \hat{\mathcal{Y}}_{tn}) < \Lambda_{\mathrm{th}}\}, \tag{16}$$

where $\hat{\mathcal{Y}}_{sn} = \{\hat{y}^s_{in} = g_n(\mathbf{x}^s_i)\}$ and $\hat{\mathcal{Y}}_{tn} = \{\hat{y}^t_{in} = g_n(\mathbf{x}^t_i)\}$ are the labels predicted by $g_n(\mathbf{x})$ for source and target-domain samples, respectively. It is worth noting that the subscript n points out a classification model obtained starting from the same classification technique using different values of parameters in the training phase (e.g., a DASVM classifier with different values of learning parameters). On one hand, $\mathcal{A}$ and $\mathcal{B}$ contain all of the classifiers that permit to obtain solutions whose accuracy is higher than or equal to Λ_{th} for source and target-domain samples (*consistent* solutions), respectively. On the other hand, $\mathcal{C}$ and $\mathcal{D}$ contain all of the classifiers that provide solutions whose accuracy is lower than Λ_{th} for source and target-domain samples (*nonconsistent* solutions), respectively. Λ_{th} represents the smallest value for Λ such that a solution can be considered acceptable for the problem under investigation.

Each classifier $g_n(\mathbf{x})$ is associated with an estimate of the joint probability distribution $\hat{P}_n(\mathbf{x}, y) = \hat{P}_n(y \mid \mathbf{x}) \cdot \hat{P}(\mathbf{x})$ for the considered domain. Indeed, while $\hat{P}(\mathbf{x})$ directly depends on the instances available for the considered domain, the estimated conditional posterior distribution $\hat{P}_n(y \mid \mathbf{x})$ is related to the information class associated by the classifier to the generic sample $\mathbf{x}$, i.e., $\hat{P}_n(y \mid \mathbf{x}) = \hat{P}(g_n(\mathbf{x}) \mid \mathbf{x})$. Accordingly, both the source and target-domain estimated joint distributions can be written as

$$\hat{P}^s_n(\mathbf{x}, y) = \hat{P}^s_n(y \mid \mathbf{x}) \cdot \hat{P}^s(\mathbf{x}) = \hat{P}^s(g_n(\mathbf{x}) \mid \mathbf{x}) \cdot \hat{P}^s(\mathbf{x}), \tag{17}$$

$$\hat{P}^t_n(\mathbf{x}, y) = \hat{P}^t_n(y \mid \mathbf{x}) \cdot \hat{P}^t(\mathbf{x}) = \hat{P}^t(g_n(\mathbf{x}) \mid \mathbf{x}) \cdot \hat{P}^t(\mathbf{x}). \tag{18}$$

Therefore, it is possible to relate the quality of these estimated distributions with the four sets of classifiers described above:

- If $g_n(\mathbf{x}) \in \mathcal{A}$, we assume that $\hat{P}^s_n(\mathbf{x}, y)$ is consistent with $P^s(\mathbf{x}, y)$ (the system is in state $\bar{A}$).
- If $g_n(\mathbf{x}) \in \mathcal{B}$, we assume that $\hat{P}^t_n(\mathbf{x}, y)$ is consistent with $P^t(\mathbf{x}, y)$ (the system is in state $\bar{B}$).
- If $g_n(\mathbf{x}) \in \mathcal{C}$, we assume that $\hat{P}^s_n(\mathbf{x}, y)$ is not consistent with $P^s(\mathbf{x}, y)$ (the system is in state $\bar{C}$).
- If $g_n(\mathbf{x}) \in \mathcal{D}$, we assume that $\hat{P}^t_n(\mathbf{x}, y)$ is not consistent with $P^t(\mathbf{x}, y)$ (the system is in state $\bar{D}$).

Starting from state $\bar{A}$, with a proper choice of the learning parameters, a domain adaptation classifier is expected to move to state $\bar{B}$ (thus belonging to $\mathcal{B}$). On the contrary, if the choice of the parameters is not adequate (or $P^t(\mathbf{x}, y)$ is too different from $P^s(\mathbf{x}, y)$), the classifier moves to state $\bar{D}$ (thus belonging to $\mathcal{D}$). Let us now consider the solution obtained for the target-domain samples. We can address the reverse domain adaptation problem (from target to source) with the same classifier keeping the same learning parameters and jointly exploiting the classification labels $\hat{\mathcal{Y}}_{tn}$ (instead of the training set, thus defining $\hat{\mathcal{T}}_t = \{\mathcal{X}_t, \hat{\mathcal{Y}}_{tn}\}$) and source-domain samples $\mathcal{X}_s$ (considered without their labels $\mathcal{Y}_s$). As the true labels $\{y^s_i\}_{i=1}^N$ for source-domain instances are known, we can compute the value for Λ associated with the results obtained after the circular learning process. If $\Lambda < \Lambda_{\mathrm{th}}$, the classification accuracy for the source-domain problem is considered nonacceptable, then the backward classifier moves to state $\bar{C}$ (thus belonging to $\mathcal{C}$). On the contrary, if $\Lambda \geq \Lambda_{\mathrm{th}}$, the solution is consistent, thus the backward classifier belongs to $\mathcal{A}$ and the system moves back to state $\bar{A}$.

Our assumption is that, when the domain adaptation classifier starting from state $\bar{C}$ is able to return into state $\bar{A}$, the classification accuracy for target-domain data is satisfactory and $\hat{P}^t_n(\mathbf{x}, y)$ is a good approximation of $P^t(\mathbf{x}, y)$. This aspect is crucial because it means that, in such situations, we are able to assess that target-domain data are classified with a proper accuracy even if no prior knowledge is available. The two main hypotheses under which the proposed validation technique is effective are the following:

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

- *Starting from state $\bar{D}$ the system must never move back to state $\bar{A}$.* If the solution obtained in the forward sense (from source to target) for target-domain instances is not satisfactory (i.e., $\hat{P}_n^t(\mathbf{x}, y)$ is not consistent with $P^t(\mathbf{x}, y)$), by applying the considered algorithm in the backward sense (from target to source), it must never be possible to obtain an acceptable solution for source-domain instances (i.e., the resulting $\hat{P}_n^s(\mathbf{x}, y)$ is always not consistent with $P^s(\mathbf{x}, y)$).
- *Starting from state $\bar{B}$ the system can return to state $\bar{A}$.* If there exists a set of satisfactory solutions obtained in the forward sense (from source to target) for target-domain instances (i.e., the related $\hat{P}_n^t(\mathbf{x}, y)$ are consistent with $P^t(\mathbf{x}, y)$), by applying the domain adaptation classifier in the backward sense (from target to source), it must be possible to obtain for at least one of them an acceptable solution for source-domain samples (i.e., the related $\hat{P}_n^s(\mathbf{x}, y)$ is consistent with $P^s(\mathbf{x}, y)$).

It is worth noting that, under the aforementioned assumptions, the system may reject some solutions that are actually consistent with the target-domain problem as the learning parameters are not optimized for the backward process and we cannot assume a perfect symmetry between the domain adaptation problems from source to target, and vice versa. Nevertheless, the very important aspect is that the system never accepts and validates solutions that are nonconsistent, which is definitely a more critical aspect of validation in operational problems.

6 Experimental Results

In order to assess the effectiveness of both the proposed DASVM technique and the presented circular validation strategy, we carried out several experiments on different data sets. We analyzed three different domain adaptation problems: 1) a series of two-dimensional toy problems having different complexity, 2) a real problem in the framework of brain computer interface, and 3) a real problem in the context of remote sensing. For all of the data sets, true labels were available for both source and target-domain instances. However, prior information related to the target domain D_t was considered only for an objective and quantitative assessment of the performances of the proposed techniques. In the following, we will describe the common procedure adopted for analyzing the considered data sets; then, in the next sections, we will present in detail the results obtained for each of them.

In all of the trials, we employed Gaussian kernel functions (ruled by the free parameter σ) as they proved effective in addressing different kinds of problems. We chose the percentage overall accuracy *OA%* (i.e., the percentage of correctly labeled samples over the whole number of considered samples) as reference classification accuracy measure Λ and fixed $\Lambda_{th} = OA\%_{\mathrm{th}} = 85$ in the validation strategy. This means that, both for the source- and target-domain classification problems, we assumed that a solution was consistent if $OA\% \geq 85$.

In order to estimate the complexity of the investigated domain adaptation problems, we first analyzed the "distance" between source and target-domain distributions. To this aim, a common choice in the literature is to compute the Kullback-Leibler (KL) [10] divergence, which is defined as

$$D[P(\mathbf{x}) \parallel Q(\mathbf{x})] = \sum_n p_n \log \frac{p_n}{q_n}, \tag{19}$$

where p_n and q_n are point probabilities of the two considered source and target distributions P and Q, respectively [11]. Note that $D[P(\mathbf{x}) \| Q(\mathbf{x})] \in [0; \infty)$. Even if KL divergence is generally considered as a kind of distance between two distributions, it is not symmetric (i.e., $D[P(\mathbf{x}) \| Q(\mathbf{x})] \neq D[Q(\mathbf{x}) \| P(\mathbf{x})]$). Therefore, in our experiments, we considered the Jensen-Shannon divergence (D_{JS}), which is a symmetrized and smoothed version of the KL divergence [11]. D_{JS} is defined as

$$\begin{aligned} D_{JS}[P(\mathbf{x}), Q(\mathbf{x})] = \alpha \cdot D[P(\mathbf{x}) \parallel M(\mathbf{x})] \\ + \beta \cdot D[Q(\mathbf{x}) \parallel M(\mathbf{x})], \end{aligned} \tag{20}$$

where $M(\mathbf{x}) = \alpha \cdot P(\mathbf{x}) + \beta \cdot Q(\mathbf{x})$. In particular, we considered the case where $\alpha = \beta = 0.5$, (referred in the literature as *specific* D_{JS}) for which it holds that $D_{JS}[P(\mathbf{x}), Q(\mathbf{x})] \in [0; \log 2]$. The existence of both a lower and an upper bound for D_{JS} is particularly important as it permits us to understand how different the two considered distributions are. If $D_{JS}[P(\mathbf{x}), Q(\mathbf{x})] = 0$, then P and Q are considered identical, whereas if $D_{JS} = \log 2 \simeq 0.693$, P and Q are considered uncorrelated. Besides the distance $D_{JS}[P^s(\mathbf{x}), P^t(\mathbf{x})]$ evaluated between general distributions, we also analyzed the distance $D_{JS}[P^s(\mathbf{x}|\omega_i), P^t(\mathbf{x}|\omega_i)]$ between class conditional distributions.

In all experiments, we trained several supervised SVMs on the labeled source-domain samples $\mathcal{T}_s = \{\mathcal{X}_s, \mathcal{Y}_s\}$, in order to identify the models (and thus, the values of the supervised parameters σ and C) that resulted in accurate (consistent) solutions for source-domain samples. Successively, we trained a number of DASVMs using $\mathcal{T}_s$ as labeled set and target-domain instances $\mathcal{X}_t$ as unlabeled set. In the learning phase, we assigned to σ and C pairs of values associated with solutions consistent with D_s. On the remaining learning parameters (i.e., C^*, ρ, and γ), we applied a grid search. Note that, for the sake of comparison, we also evaluated the performances obtained by other two state-of-the-art domain adaptation techniques: 1) the retraining technique for maximum likelihood classifiers presented in [36] (denoted by $\mathrm{ML}_{retrain}$) and 2) the maximum likelihood cascade classifier proposed in [37] (denoted by $\mathrm{ML}_{cascade}$).

For validating the potentialities of the proposed domain adaptation technique, we identified the DASVM that provided the highest overall accuracy on the basis of the available target-domain true labels (which were not taken into account in the learning phase). This accuracy value represents an upper bound for the performances of the presented method. In order to assess the effectiveness of the empirical circular validation strategy introduced in Section 4, for all of the considered DASVMs, we applied the proposed domain adaptation algorithm in the reverse sense. Starting from the corresponding set of target-domain predicted labels, $\hat{\mathcal{Y}}_{tn}$, for the generic nth DASVM, we defined an estimated training set $\hat{\mathcal{T}}_{tn} = \{\mathcal{X}_t, \hat{\mathcal{Y}}_{tn}\}$ for D_t and trained the correspondent nth "backward" DASVM using $\hat{\mathcal{T}}_{tn}$ as labeled

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

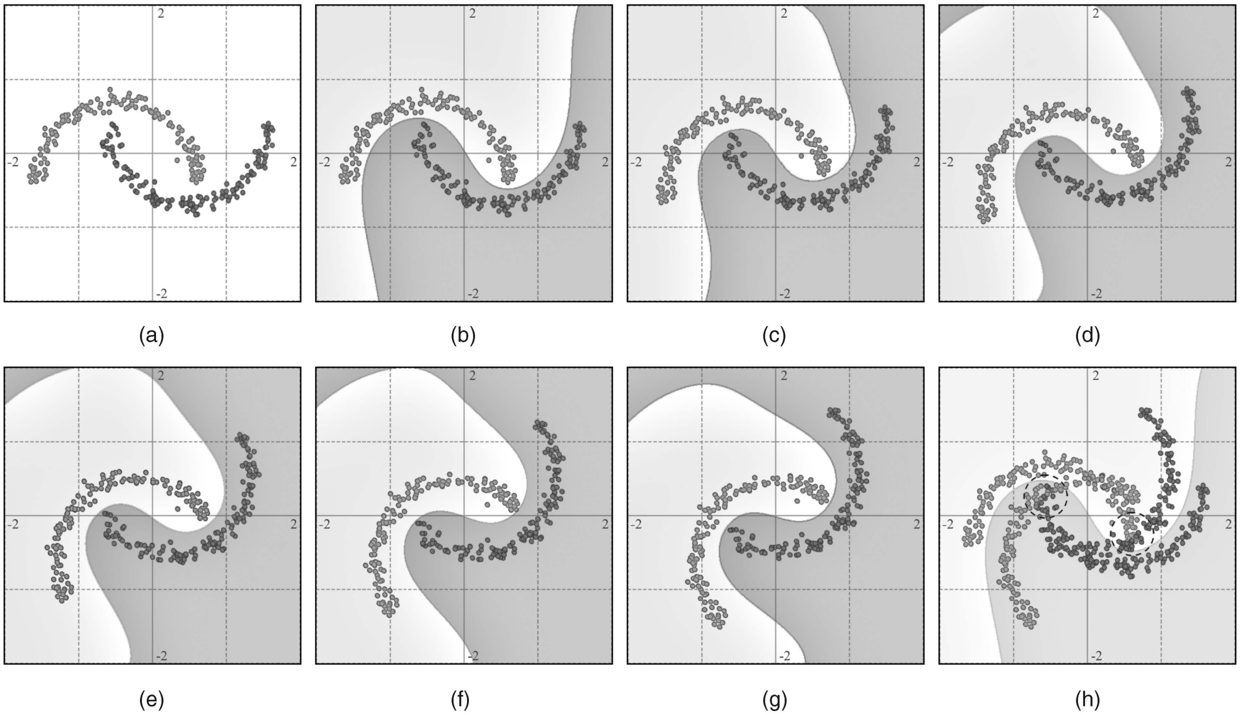


Fig. 4. Problem I: (a) original source-domain data; (b) decision regions obtained for the source-domain problem by a supervised SVM trained according to a 10-fold CV strategy. Decision regions obtained for the target-domain problem by the proposed DASVM technique with optimal selection of learning parameters when (c) $\phi = 10°$, (d) $\phi = 20°$, (e) $\phi = 30°$, (f) $\phi = 40°$, and (g) $\phi = 50°$. (h) Superimposition of source data and target data for $\phi = 50°$ (the dashed circles point out regions, where target data that belong to class ω_1 overlap source data that belong to class ω_2, and vice versa) and decision regions obtained for the source-domain problem by a supervised SVM trained according to a 10-fold CV strategy.

set and source-domain samples $\mathcal{X}_s$ (considered without their labels) as unlabeled set (the same learning parameters employed in the forward learning were used). By exploiting available prior information for D_s, we determined whether the final solution $\hat{\mathcal{Y}}_{sn}$ was consistent or not and, accordingly, inferred the correctness of the related solution to the target-domain problem $\hat{\mathcal{Y}}_{tn}$. By using the available target-domain true labels, we could compute the average percentage overall accuracy of the solutions correctly identified as consistent with D_t by the circular validation strategy. This value is very important as it represents an average measure for the quality of the solutions consistent with D_t identified by the proposed validation strategy without any prior information on $\mathcal{X}_t$. Note that, in order to obtain significant estimations, for all of the considered problems, we trained 350 backward DASVMs both starting from consistent and nonconsistent solutions with the target-domain problem. For the sake of comparison, we also evaluated the accuracies exhibited on target-domain samples by a supervised SVM trained on source-domain data according to a 10-fold cross validation (CV).

In all experiments, we employed the Sequential Minimal Optimization algorithm [42] for training both the supervised SVMs and, with proper modifications, the proposed DASVMs. As pointed out in Section 4, we fixed $\tau = 0.5$. Concerning the convergence criterion, a reasonable empirical choice has proven to be $\beta = 3{\cdot}10^{-2}$.

6.1 Problem I: Synthetic Data Set

The first set of experiments was aimed at characterizing the behavior of both the DASVM algorithm and the circular validation strategy when addressing a well-defined problem in a controlled environment at different levels of complexity. This analysis is particularly important for empirically understanding the operational conditions for which we can expect to obtain satisfactory performances with the proposed methods. We considered as source-domain data a toy data set made up of 300 samples generated according to a bidimensional pattern of two intertwining moons associated with two specific information classes (150 samples each), as shown in Fig. 4a. Target data were generated by rotating anticlockwise the original source data set 11 times by 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, and 60 degrees, respectively. Due to rotation, source and target-domain data exhibit different distributions (i.e., $P^s(\mathbf{x}) \neq P^t(\mathbf{x})$ and $P^s(y \mid \mathbf{x}) \neq P^t(y \mid \mathbf{x})$). In particular, the greater the rotation angle (ϕ), the more complex the resulting domain adaptation problem, as confirmed by the values for D_{JS} reported in Table 2.

The proposed DASVM algorithm proved to be particularly effective for solving this kind of problem and involved very high accuracies even in very critical conditions. From Table 3, one can observe that, for an optimal selection of learning parameters (DASVM^{best}), we could obtain perfect separation between information classes when $\phi \in [10°; 50°]$ (see Figs. 4c, 4d, 4e, 4f, and 4g). The accuracies are always higher than those exhibited by the supervised SVM trained according to a 10-fold CV on source-domain data (the $OA\%$ grows almost quadratically with respect to the rotation angle, i.e., from $+0.33$ for $\phi = 10°$ to $+67.33$ for $\phi = 50°$). Only for greater values of ϕ (i.e., $55°$ and $60°$) the DASVM

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
Problem I: Jensen-Shannon Divergence Values for Different Rotation Angles

	Rotation Angle										
	10°	15°	20°	25°	30°	35°	40°	45°	50°	55°	60°
$D_{JS}[P^s(\mathbf{x}),P^t(\mathbf{x})]$	0.040	0.091	0.150	0.211	0.260	0.291	0.378	0.410	0.435	0.442	0.457
$D_{JS}[P^s(\mathbf{x}\|\omega_1),P^t(\mathbf{x}\|\omega_1)]$	0.061	0.159	0.225	0.335	0.386	0.437	0.536	0.554	0.559	0.574	0.576
$D_{JS}[P^s(\mathbf{x}\|\omega_2),P^t(\mathbf{x}\|\omega_2)]$	0.071	0.170	0.203	0.304	0.371	0.438	0.468	0.486	0.538	0.572	0.573

was not able to find a solution consistent with D_t. Nevertheless, this behavior seems reasonable due to the complexity of the corresponding domain adaptation problems. In these cases, the initial separation hyperplane determined according to source-domain samples resulted in an average $OA\%$ on target-domain data smaller than 30. Accordingly, it was not possible to recover correct classification labels as more than 70 percent of target-domain samples were misclassified at the first iteration of the DASVM algorithm. The complexity of this problem is also confirmed from the high values of D_{JS}.

It should be pointed out that, as soon as the rotation angle becomes greater than $35°$, target-domain data that actually belong to the class ω_1 overlap source-domain data belonging to the class ω_2, and vice versa (see, for instance, Fig. 4h). Note that, due to rotation, there are target-domain instances that coincide with source-domain instances but with different true labels. This represents a strong limitation for other state-of-the-art domain adaptation techniques reported in Section 2. The proposed DASVM technique, due do to the fact that iteratively erases original training samples in the iterative learning phase, does not suffer from this drawback and is able to obtain satisfactory performances also in such critical cases. It is worth noting that DASVMs outperformed both $\mathrm{ML}_{retrain}$ and $\mathrm{ML}_{cascade}$: the greater the rotation angle ϕ (and thus, the greater the problem complexity), the higher the gap in terms of classification accuracy (see Table 3). In particular, for $\phi = 50°$, the increase in $OA\%$ is around 30, thus further confirming the effectiveness of the proposed technique in addressing also very critical situations.

With regard to the presented circular validation strategy, we obtained very promising results. In particular, for all of the considered cases, the proposed strategy was always able to correctly reject solutions that were not consistent with D_t, thus satisfying the most critical assumption for the operational employment of the proposed technique. As expected, when the DASVM started from a solution that did not adequately model the source-domain classification problem, the system could not recover a solution consistent with D_s, thus $P(\bar{A} \mid \bar{D}) = 0$. Table 3 also reports the probability of correct validation of solutions consistent with D_t, $P(\bar{A} \mid \bar{B})$, for the cases in which it was possible to obtain at least one of them in the forward learning phase (i.e., $\phi \in [10°; 50°]$). It is possible to notice that, for small values of ϕ, the proposed circular strategy identified more than one half of correct solutions, while, by increasing ϕ values, the number of correct solutions properly recognized gradually decreased. This is a reasonable behavior if we consider that the distance between distributions sharply increases and learning parameters are not optimized for the backward process.

Fig. 5 shows some examples of the empirical estimated probability distribution $\hat{P}(OA\%)$ of the $OA\%$ for the solutions obtained for source-domain data at the end of the backward learning process when the system starts from both state $\bar{B}$ (black line) and state $\bar{D}$ (gray line). If we consider as

TABLE 3
Problem I: Percentage Overall Accuracy Exhibited on Target-Domain Instances for Different Rotation Angles

	Rotation Angle										
	10°	15°	20°	25°	30°	35°	40°	45°	50°	55°	60°
SVM^{CV}	99.67	96.67	89.33	77.67	62.00	52.33	43.67	37.33	32.67	29.33	28.67
DASVM^{best}	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	57.33	52.33
DASVM^{ave}	99.76	99.17	99.33	99.71	99.64	99.72	96.19	98.54	96.03	–	–
$P(\bar{A}\|\bar{B})$	0.65	0.58	0.51	0.50	0.48	0.45	0.33	0.22	0.15	–	–

(a)

	Rotation Angle										
	10°	15°	20°	25°	30°	35°	40°	45°	50°	55°	60°
$\mathrm{ML}_{retrain}$	99.67	98.67	96.33	94.33	92.00	89.33	84.67	80.00	65.33	54.00	51.33
$\mathrm{ML}_{cascade}$	99.67	98.33	97.33	94.67	93.33	89.67	86.00	78.33	67.67	54.67	51.00

(b)

(a) Percentage OA exhibited by: 1) A supervised SVM trained on source-domain samples according to a 10-fold CV strategy (SVM^{CV}); 2) the proposed DASVM technique with optimal selection of learning parameters ($DASVM^{best}$). The average accuracy associated with the consistent solutions obtained by the proposed DASVM technique and correctly identified by the circular validation strategy ($DASVM^{ave}$) and the probability $P(\bar{A} \mid \bar{B})$ of identifying consistent solutions with the proposed circular validation strategy are also given. (b) Percentage OA exhibited by: 1) The retraining technique for maximum likelihood classifier ($ML_{retrain}$); 2) the maximum likelihood cascade classifier ($ML_{cascade}$).

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

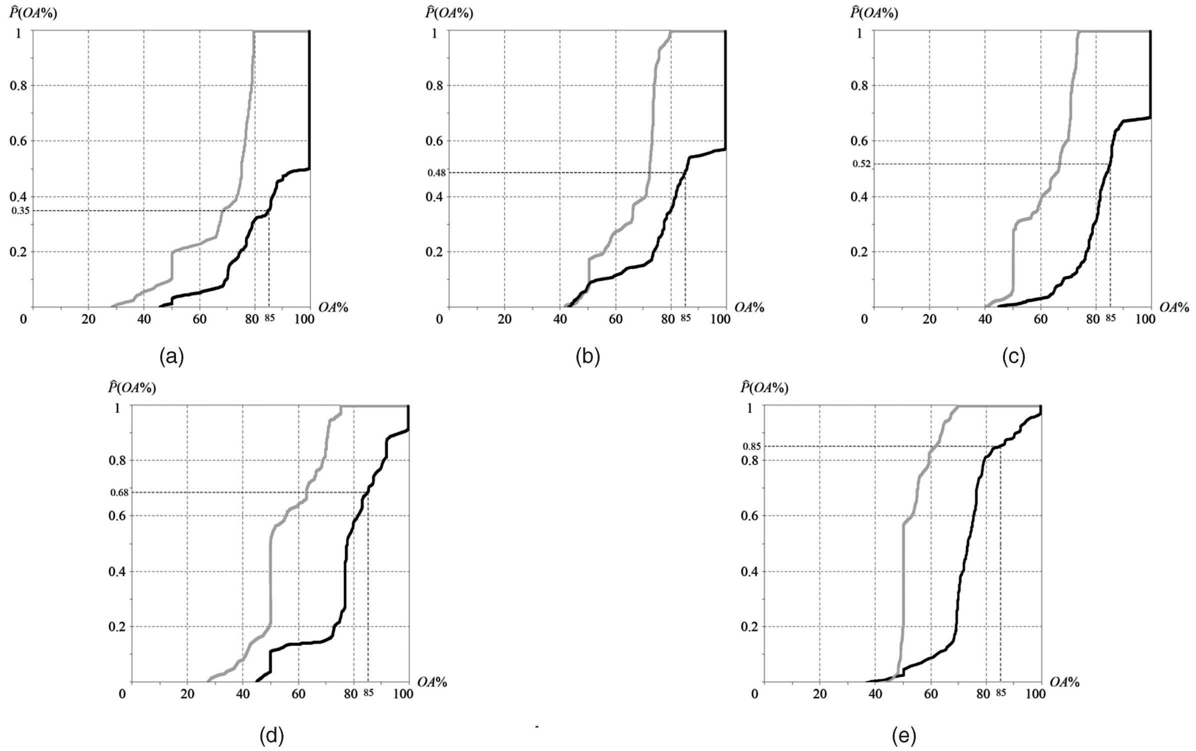


Fig. 5. Problem I: Empirical estimated probability distribution $\hat{P}(OA\%)$ of the percentage overall accuracy (*OA%*) for the solutions obtained on the source domain when the system starts from both state $\bar{B}$ (black line) and state $\bar{D}$ (gray line) for (a) $\phi = 10°$, (b) $\phi = 20°$, (c) $\phi = 30°$, (d) $\phi = 40°$, and (e) $\phi = 50°$. If $OA\% < 85$, the system moves to state $\bar{C}$; if $OA\% \geq 85$, the system moves to state $\bar{A}$.

an example Fig. 2b (which refers to the case $\phi = 20°$), we can see that $q_{0.85}(OA\%)$ (i.e., the quantile corresponding to $OA\%_{th} = 85$) is equal to 0.48. Accordingly, as the systems move to state $\bar{C}$ if $OA\% < OA\%_{th}$, we have that $P(\bar{C}|\bar{B}) = q_{0.85}(OA\%) = 0.48$; therefore, $P(\bar{A}|\bar{B}) = 1 - P(\bar{C}|\bar{B})$, which means that the classifier can go back to state $\bar{A}$ in the 52 percent of the cases. The very important conclusion of this analysis is that even in critical situations, the proposed circular validation strategy was able to identify correct solutions without considering any prior information on the labels of target-domain data. Moreover, if we consider the average $OA\%$ of solutions correctly identified as consistent with D_t, we can see that it is comparable with that obtained with optimal selection of leaning parameters, thus confirming the effectiveness of the proposed circular validation strategy.

6.2 Problem II: Brain Computer Interface Data Set

The second data set considered refers to a Brain Computer Interface (BCI) problem. A BCI is an assistive communication system, which helps people with severe disabilities to realize the control of motor neuroprotheses. The data set we considered was made up of electrocorticogram (ECoG) signals recorded from the same subject performing the same task in two different days (i.e., at times t_1 and t_2) with about one week in between. In the considered case, the subject had to perform imagined movements of either the left small finger or the tongue (associated with the classes ω_1 and ω_2, respectively) for at least 3 seconds. For greater details about this data set, the reader is referred to [43], [44]. It is worth noting that the design of a classifier for a BCI system is very challenging when a classifier trained on data acquired on a certain day should classify data recorded in other days without retraining. On one hand, the patient might be in a different state concerning motivation, fatigue, etc. Therefore, his brain will show different electrical activity. On the other hand, the recording system might have undergone slight changes concerning electrode positions and impedances.

Fig. 6a presents the system designed for extracting the features used in our experiments. Original ECoG signals were first low-pass (0-3 Hz) and band-pass (8-30 Hz) filtered in order to acquire movement-related potentials (MRD) and event-related desynchronization (ERD) signals, respectively, which are electrical physiological phenomena activated by limb movements or imagined movements [45], [46]. Then, we used the common spatial subspace decomposition (CSSD) technique, which allows us to extract signal components specific to one condition and eliminate background activities [47]. For both frequency intervals, we selected the two components that, according to the CSSD technique, are more related to the considered information classes. The final data set is obtained by merging together these two pairs of features. For the source domain at time t_1, the brain activity was monitored for 278 events (139 associated with each information class), whereas, for the target domain at time t_2, 100 events were considered (50 for each information class).

The resulting domain adaptation problem proved to be particularly challenging, as confirmed by the values for the D_{JS} reported in Table 4. Source and target-domain overall distributions were rather different (0.408), but the gap was even more relevant for the conditional class distributions (0.625 for the class ω_1 and 0.616 for the class ω_2). Even in such critical conditions, the DASVM algorithm proved effective

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

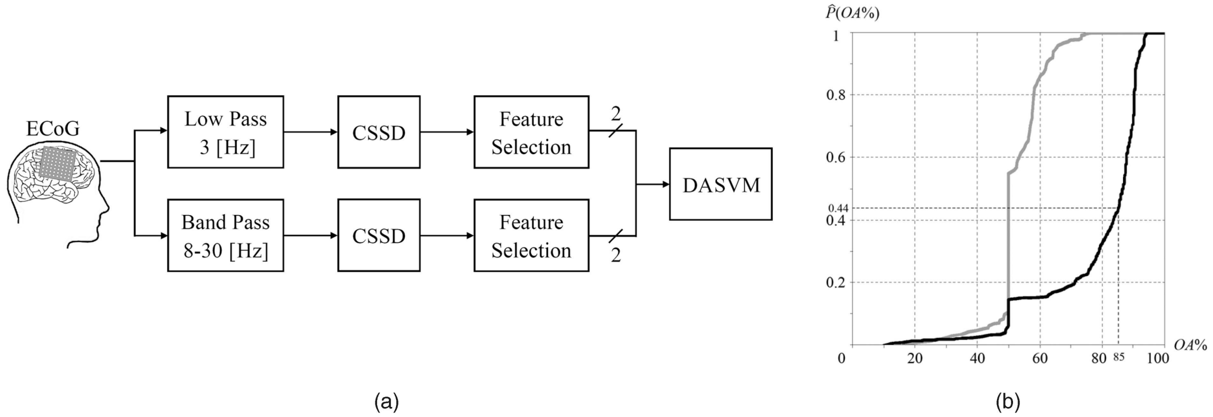


Fig. 6. Problem II: (a) Architecture of the system employed for extracting the features used in the experiments. (b) Empirical estimated probability distribution $\hat{P}(OA\%)$ of the percentage overall accuracy (*OA%*) for the solutions obtained on the source domain when the system starts from both state $\bar{B}$ (black line) and state $\bar{D}$ (gray line). If $OA\% < 85$, the system moves to state $\bar{C}$; if $OA\% \geq 85$, the system moves to state $\bar{A}$.

TABLE 4
Problem II: Jensen-Shannon Divergence Values

$D_{JS}[P^s(\mathbf{x}),P^t(\mathbf{x})]$	$D_{JS}[P^s(\mathbf{x}\mid\omega_1),P^t(\mathbf{x}\mid\omega_1)]$	$D_{JS}[P^s(\mathbf{x}\mid\omega_2),P^t(\mathbf{x}\mid\omega_2)]$
0.408	0.625	0.616

TABLE 5
Problem II: Percentage Overall Accuracy (*OA%*) Exhibited on Target-Domain Instances by:
1) a Supervised SVM Trained on Source-Domain Instances According to a 10-Fold CV Strategy (SVM^{CV});
2) the Proposed DASVM Technique with Optimal Selection of Learning Parameters (DASVM^{best});
3) the Maximum Likelihood Classifier ($\mathrm{ML}_{retrain}$); and 4) the Maximum Likelihood Cascade Classifier ($\mathrm{ML}_{cascade}$)

	SVM^{CV}	DASVM^{best}	DASVM^{ave}	$\mathrm{ML}_{retrain}$	$\mathrm{ML}_{cascade}$
OA%	79.00	93.00	91.65	88.00	87.00

The average accuracy associated with the consistent solutions obtained by the proposed DASVM technique and correctly identified by the circular validation strategy (DASVM^{ave}) is also given.

(see Table 5). In particular, in the best case, we were able to obtain an *OA%* equal to 93.00, which corresponds to an increase of $+14.00$ with respect to the *OA%* yielded with the standard supervised SVM trained according to the 10-fold CV strategy at time t_1. Moreover, DASVMs provided also higher *OA%* than both $\mathrm{ML}_{retrain}$ $(+5)$ and $\mathrm{ML}_{cascade}$ $(+6)$. The proposed technique exhibited a very good capability of seizing the structure of the investigated domain adaptation problem, as it is possible to infer from the behavior of $\hat{P}(OA\%)$ (see Fig. 6b). In particular, with the circular validation strategy, we were able to correctly identify 66 percent of solutions consistent at time t_2 (i.e., $q_{0.85}(OA\%) = 0.44$) with an average *OA%* equal to 91.65 $(+11.65$ with respect to the supervised SVM). This means that it was possible to recover a satisfactory accuracy for source-domain data only starting from high accuracies for target-domain data, thus confirming that solutions are correctly validated as consistent only if the target-domain problem is well modeled. Accordingly, also for this data set, the system never moved back from state $\bar{D}$ to state $\bar{A}$, thus exhibiting the crucial property to be always able to reject wrong solutions.

6.3 Problem III: Remote Sensing Data Set

The third set of experiments was carried out on a multiclass problem in the context of a remote sensing application. We investigated the task of automatic updating of land-cover maps by classification of remote sensing images acquired over the same geographical area at two different times t_1 and t_2. We considered two coregistered multispectral images acquired in September 1995 and July 1996 by the Thematic Mapper (TM) sensor of the Landsat 5 Satellite (see Figs. 7a and 7b). The selected test site was a section of about $11.7\ \mathrm{km} \times 10.8\ \mathrm{km}$ (i.e., 412×382 pixels) of a scene including Lake Mulargia on the Island of Sardinia, Italy. The five information classes that characterized the investigated site at both times were taken into account, i.e., forest, pasture, urban area, water body, and vineyard. For both source and target-domain problems, the same number of samples was considered for each information class (see Table 6). It is worth noting that, due to many differences at the two acquisition times (e.g., different acquisition system state, dissimilar illumination conditions, alterations in the phenologic state of vegetation, changes occurred on the ground, etc.), the distance between spectral distributions of the two images is considerable.

Among the seven available spectral bands, as commonly done in the literature, we did not take into account band 6, which corresponds to the low-resolution channel acquired in the thermal infrared. In order to characterize the texture properties of the considered classes and to exploit the distribution-free nature of SVMs, we extracted five texture features based on the gray-level co-occurrence matrix (i.e.,

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

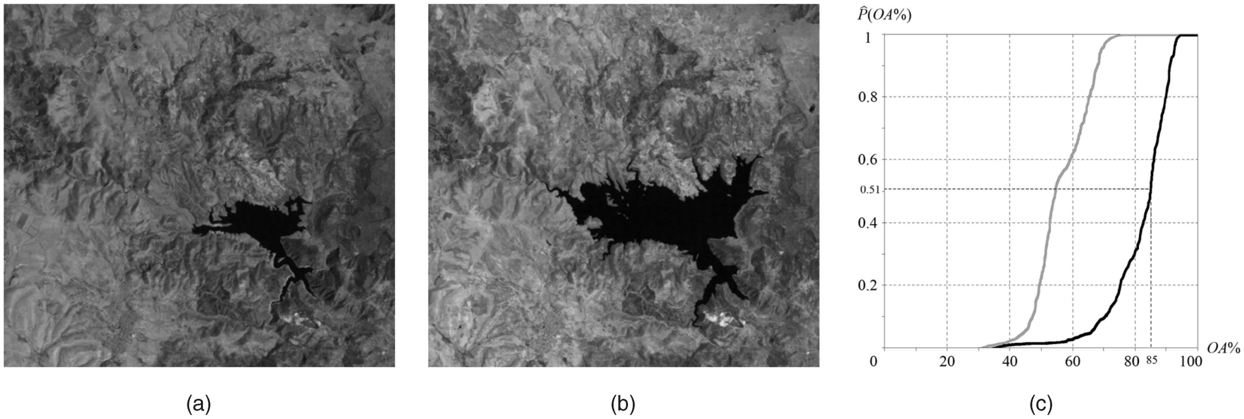


Fig. 7. Problem III: Spectral channel 5 of the multispectral Landsat-5 Thematic Mapper images used in the experiments. (a) Image acquired in September 1995. (b) Image acquired in July 1996. (c) Empirical estimated probability distribution $\hat{P}(OA\%)$ of the percentage overall accuracy (*OA%*) for the solutions obtained on the source domain when the system starts from both state $\bar{B}$ (black line) and state $\bar{D}$ (gray line). If $OA\% < 85$, the system moves to state $\bar{C}$; if $OA\% \geq 85$, the system moves to state $\bar{A}$.

TABLE 6
Problem III: Number of Both Source-Domain (September 1995 Image) and Target-Domain (July 1996 Image) Patterns and Jensen-Shannon Divergence Values

	Pasture	Forest	Urban Area	Water Body	Vineyard	Overall
Number of Patterns	1143 (27.23%)	578 (13.77%)	826 (19.68%)	1355 (32.27%)	296 (7.05%)	4198
D_{JS}	0.517	0.278	0.391	0.423	0.567	0.391

TABLE 7
Problem III: Percentage Overall Accuracy (*OA%*), Producer's Accuracies (*PA%*), and User's Accuracies (*UA%*) Exhibited on Target-Domain Instances by: 1) a Supervised SVM Trained on Source-Domain Instances According to a 10-Fold CV Strategy (SVM^{CV}); 2) the Proposed DASVM Technique with Optimal Selection of Learning Parameters (DASVM^{best}); 3) the Retraining Technique for the Maximum Likelihood Classifier (ML$_{retrain}$); and 4) the Maximum Likelihood Cascade Classifier (ML$_{cascade}$)

	OA%	Pasture		Forest		Urban Area		Water Body		Vineyard	
		PA%	*UA%*	*PA%*	*UA%*	*PA%*	*UA%*	*PA%*	*UA%*	*PA%*	*UA%*
SVM^{CV}	75.73	26.86	85.28	98.10	79.30	87.17	99.72	100.00	96.85	77.70	22.95
DASVM^{best}	94.78	90.64	94.35	98.27	96.92	92.49	99.87	100.00	100.00	86.82	65.22
DASVM^{ave}	90.88	79.79	92.49	98.10	86.83	89.71	99.86	100.00	100.00	81.08	51.95
ML$_{retrain}$	92.76	94.06	88.50	87.22	88.19	93.06	97.49	100	100	64.10	73.53
ML$_{cascade}$	91.48	94.25	83.53	90.51	97.45	81.48	95.69	100	100	86.90	62.39

The average accuracies associated with the consistent solutions obtained by the proposed DASVM technique and correctly identified by the circular validation strategy (DASVM^{ave}) are also given.

correlation, sum average, sum variance, difference variance, and entropy) and added them to the six TM channels. As the images were acquired in different periods of the year, the resulting overall D_{JS} distance between source-domain distribution at time t_1 and target-domain distribution at time t_2 was considerable (i.e., 0.391). The complexity of the investigated domain adaptation problem is stressed up by the distances between conditional class distribution at the two dates, which assume high values, in particular, for the classes pasture (i.e., 0.517) and vineyard (i.e., 0.567). As pointed out in Section 3, one of the constraints imposed by the DASVM algorithm is the use of the OAA multiclass architecture; therefore, we adopted the same multiclass strategy also when dealing with supervised SVMs used for comparisons.

Table 7 shows the results obtained in terms of $OA\%$ and both percentage producer's and user's accuracies[5] (i.e., $PA\%$ and $UA\%$, respectively) for each information class. Taking into account the complexity of the problem under investigation, the obtained classification accuracies confirmed also in this case the good adaptation capabilities of the proposed DASVM technique, which was able to sharply increase the accuracy with respect to the supervised SVM trained according to a 10-fold CV on source-domain samples. In particular, in the best case, the $OA\%$

[5] For each specific information class: 1) Producer's accuracy is defined as the ratio between the number of samples correctly classified as belonging to that class and the total number of reference samples available for that class; 2) user's accuracy is defined as the ratio between the number of samples correctly classified as belonging to that class and the total number of samples classified as belonging to that class.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

increased up to 94.78 (i.e., $+19.05$). Moreover, the improvement in the $PA\%$ is huge for pasture (i.e., $+63.78$) and remarkable for vineyard ($+9.12$), whereas the increase in the $UA\%$ is noteworthy for vineyard (i.e., $+42.27$), forest ($+17.62$), and pasture ($+9.07$).

Also, in this case, DASVMs exhibited higher accuracies than $\mathrm{ML}_{retrain}$ and $\mathrm{ML}_{cascade}$ ($+2.02$ and $+3.03$ in terms of $OA\%$ for optimal selection of learning parameters), thus confirming their effectiveness also when addressing multiclass problems.

Fig. 7c points out the effectiveness of the proposed circular validation strategy also on this data set. This strategy allowed us to correctly reject all of the solutions that were not consistent with D_t. In addition, when the system started from state $\bar{B}$, we had $q_{0.85}(OA\%) = 0.51$, which corresponds to almost one-half (i.e., 49 percent) of consistent solutions properly validated without any prior information for the image at time t_2. It is worth noting that, in most part of the cases, the system moved back to state $\bar{A}$ when the classification accuracy at t_2 was particularly high. This is confirmed by the high average $OA\%$ obtained for target-domain instances for the solutions correctly identified as consistent by the validation strategy. The average $OA\%$ is equal to 90.88 ($+15.15$ with respect to the supervised SVM), whereas as concerns both $PA\%$ and $UA\%$, the behavior is similar to that obtained for the best case described above. In particular, the increase in pasture $PA\%$ and vineyard $UA\%$ is considerable (i.e., $+52.93$ and $+29.00$, respectively).

7 Discussion and Conclusion

In this paper, we have addressed domain adaptation problems by introducing two main novel contributions: 1) a domain adaptation classifier based on SVMs (DASVM) and 2) a circular strategy aimed at validating the results obtained with a domain adaptation classifier.

The proposed DASVM technique extends the principles of SVMs to the domain adaptation framework by taking into account that unlabeled test samples are drawn from a target domain D_t different from the source domain D_s of training samples. Labeled patterns drawn from D_s are used only for constraining the learning phase of the classifier, but the final solution only models the structure of D_t. In particular, labeled source-domain data are employed for determining an initial discriminant function for the target-domain problem; then, unlabeled samples from D_t are exploited for properly adjusting the decision function, while samples from D_s are gradually erased. The final classification function is determined only on the basis of *semilabeled* samples, i.e., originally unlabeled target-domain instances that obtain labels during the learning process. For better controlling the behavior and stability of the classifier, an adaptive weighting strategy for the regularization parameters based on a temporal criterion is adopted. This permits us to tune the influence of unlabeled patterns and, in general, prevents the system from converging to improper solutions.

It is worth noting that the proposed DASVM is designed for addressing a problem conceptually different from those faced by transductive and semi-supervised SVMs, which have been defined for handling problems where labeled and unlabeled data are drawn from the same domain. Thus, they are ineffective in domain adaptation, where training data are assumed to be available only for a source domain different (even if related) from the target domain of the (unlabeled) test samples.

As for transductive and semi-supervised SVMs, also for DASVMs it is not possible to guarantee for obtaining reliable solutions to the classification problem. In fact, if the accuracy after the initialization phase is particularly low (i.e., most of the unlabeled target-domain samples are incorrectly classified at the beginning of the learning process), it becomes difficult to obtain satisfactory performances. The convergence of the algorithm to a consistent solution depends on the intrinsic correlation between the two domains: The farther D_t is from D_s, the harder the resulting domain adaptation problem. This correlation can be measured by computing similarity metrics between D_s and D_t.

In the framework of domain adaptation, due to the absence of prior information for target-domain instances, standard statistical validation strategies proposed in the literature cannot be used for assessing the effectiveness of learning and the validity of the resulting classifier. In this context, we proposed an indirect circular validation strategy based on the idea that an intrinsic structure relates the solutions consistent with D_s and D_t. A solution for D_t (for which no prior information is available) is assumed to be consistent if the solution obtained by applying the same domain adaptation learning algorithm in the reverse sense (i.e., using the classification labels in place of missing prior knowledge for target-domain instances) to source-domain data (considered as unlabeled in the reverse domain adaptation learning) is associated with an acceptable accuracy (which can be evaluated due to the available true labels for source-domain samples).

From the analysis of experimental results obtained on a series of two-dimensional toy problems and on two real problems defined in the context of brain computer interface and remote sensing, we can conclude that the presented DASVM resulted in high and satisfactory classification accuracy, outperforming other state-of-the-art domain adaptation techniques. Several trials also confirmed the effectiveness of the proposed circular validation strategy, which always proved able to reject solutions that were not consistent with the investigated classification task and identified acceptable solutions for D_t even in very critical conditions. In this framework, the joint use of both the proposed DASVM classification technique and the circular validation strategy seems particularly suitable to define systems capable of solving real application problems.

As a final remark, it is worth noting that, from a computational viewpoint, each iteration of the DASVMs requires a time equivalent to that taken from the learning of a supervised SVM. In fact, as the number of target-domain semilabeled patterns increases, the number of source-domain labeled patterns decreases; therefore, the cardinality of the training set does not increase with the number of iterations. Accordingly, it is possible to state that the computational load grows linearly with the number of iterations. In general, the computational time is comparable to the one of semi-supervised methods and higher than the

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

one required by supervised SVMs (on average, a hundred of iterations for each binary DASVM are needed). Nevertheless, taking into account both the very promising results obtained and the complexity of the investigated problems, we can consider this cost reasonable.

Acknowledgments

The authors would like to thank the associate editor and the anonymous reviewers for their valuable comments.

References

[1] R. Caruana, "Multitask Learning," *Machine Learning J.,* vol. 28, no. 1, pp. 41-75, 1997.

[2] S. Thrun and L.Y. Pratt, *Learning to Learn.* Kluwer Academic Publishers, 1998.

[3] S. Ben-David and R. Schuller, "Exploiting Task Relatedness for Multiple Task Learning," *Proc. 16th Ann. Conf. Learning Theory,* 2003.

[4] B. Zadrozny, "Learning and Evaluating Classifiers under Sample Selection Bias," *Proc. 21st Int'l Conf. Machine Learning,* 2004.

[5] M. Dudik, R.E. Schapire, and J.S. Philips, "Correcting Sample Selection Bias in Maximum Entropy Density Estimation," *Advances in Neural Information Processing Systems 17,* MIT Press, 2005.

[6] J. Huang, A. Smola, A. Gretton, K.M. Borgwardt, and B. Schölkopf, "Correcting Sample Selection Bias by Unlabeled Data," *Advances in Neural Information Processing Systems 20,* MIT Press, 2007.

[7] H. Shimodaira, "Improving Predictive Inference under Covariate Shift by Weighting the Loglikelihood Function" *J. Statistical Planning and Inference,* vol. 90, pp. 227-244, 2000.

[8] M. Sugiyama and K.R. Müller, "Input-Dependent Estimation of Generalization Error under Covariate Shift," *Statistics and Decisions,* vol. 23, pp. 249-279, 2005.

[9] H. Jeffreys, "An Invariant Form for the Prior Probability in Estimation Problems," *Proc. Royal Soc. London,* vol. 186, pp. 453-461, 1946.

[10] S. Kullback and R. Leibler, "On Information and Sufficiency," *Annals of Math. Statistics,* vol. 22, pp. 79-86, 1951.

[11] J. Lin, "Divergence Measures Based on the Shannon Entropy," *IEEE Trans. Information Theory,* vol. 37, pp. 145-151, 1991.

[12] V.N. Vapnik, *Statistical Learning Theory.* John Wiley & Sons, Inc., 1998.

[13] V.N. Vapnik, *The Nature of Statistical Learning Theory,* second ed. Springer-Verlag, 1995.

[14] M. Pontil and A. Verri, "Support Vector Machines for 3D Object Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 6, pp. 637-646, June 1998.

[15] G. Ratsch, S. Mika, B. Schölkopf, and K.R. Muller, "Constructing Boosting Algorithms from SVMs: An Application to One-Class Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 9, pp. 1184-1199, Sept. 2002.

[16] K. In Kim, K. Jung, S.H. Park, and H.J. Kim, "Support Vector Machines for Texture Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 11, pp. 1542-1550, Nov. 2002.

[17] K.P. Bennett and A. Demiriz, "Semi-Supervised Support Vector Machines," *Advances in Neural Information Processing Systems,* vol. 10, pp. 368-374, MIT Press, 1998.

[18] G. Fung and O.L. Mangasarian, "Semi-Supervised Support Vector Machines for Unlabeled Data Classification," *Optimization Methods and Software,* vol. 15, no. 1, 2001.

[19] X. Zhu, "Semi-Supervised Learning Literature Survey," TR-1530, Computer Sciences, Univ. of Wisconsin-Madison, 2005.

[20] T. Joachims, "Transductive Inference for Text Classification Using Support Vector Machines," *Proc. 16th Int'l Conf. Machine Learning,* 1999.

[21] Y. Chen, G. Wang, and S. Dong, "Learning with Progressive Transductive Support Vector Machine," *Pattern Recognition Letters,* vol. 24, no. 12, pp. 1845-1855, 2003.

[22] R. Hwa, "Supervised Grammar Induction Using Training Data with Limited Constituent Information," *Proc. 37th Ann. Meeting of the Assoc. for Computational Linguistics,* 1999.

[23] D. Gildea, "Corpus Variation and Parser Performance," *Proc. 2001 Conf. Empirical Methods in Natural Language Processing,* 2001.

[24] B. Roark and M. Bacchiani, "Supervised and Unsupervised PCFG Adaptation to Novel Domains," *Proc. 2003 Conf. North Am. Chapter of the Assoc. for Computational Linguistics and Human Language Technology,* 2003.

[25] X. Li and J. Bilmes, "A Bayesian Divergence Prior for Classifier Adaptation," *Proc. 11th Int'l Conf. Artificial Intelligence and Statistics,* 2007.

[26] C. Chelba and A. Acero, "Adaptation of Maximum Entropy Capitalizer: Little Data Can Help a Lot," *Proc. 2004 Conf. Empirical Methods in Natural Language Processing,* 2004.

[27] H. Daumè III and D. Marcu, "Domain Adaptation for Statistical Classifiers," *J. Artificial Intelligence Research,* vol. 26, pp. 101-126, 2006.

[28] J. Jiang and C. Zhai, "Instance Weighting for Domain Adaptation in NLP," *Proc. 45th Ann. Meeting of the Assoc. for Computational Linguistics,* 2007.

[29] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "A Statistical Model for Multilingual Entity Detection and Tracking," *Proc. 2004 Conf. North Am. Chapter of the Assoc. for Computational Linguistics and Human Language Technology,* 2004.

[30] H. Daumè III, "Frustratingly Easy Domain Adaptation," *Proc. 45th Ann. Meeting of the Assoc. for Computational Linguistics,* 2007.

[31] J. Blitzer, R. McDonald, and F. Pereira, "Domain Adaptation with Structural Correspondence Learning," *Proc. 2006 Conf. Empirical Methods in Natural Language Processing,* 2006.

[32] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of Representation for Domain Adaptation," *Advances in Neural Information Processing Systems 19,* MIT Press, 2006.

[33] S. Satpal and S. Sarawagi, "Domain Adaptation of Conditional Probability Models via Feature Subsetting," *Proc. 11th European Conf. Principles and Practice of Knowledge Discovery in Databases,* 2007.

[34] W. Dai, G.R. Xue, Q. Yang, and Y. Yu, "Transferring Naïve Bayes Classifier for Text Classification," *Proc. 22nd Nat'l Conf. Artificial Intelligence,* 2007.

[35] W. Dai, G.R. Xue, Q. Yang, and Y. Yu, "Co-Clustering Based Classification for Out-of-Domain Documents," *Proc. ACM SIGKDD,* 2007.

[36] L. Bruzzone and D. Fernàndez Prieto, "Unsupervised Retraining of a Maximum-Likelihood Classifier for the Analysis of Multitemporal Remote-Sensing Images," *IEEE Trans. Geosciences and Remote Sensing,* vol. 39, pp. 456-460, 2001.

[37] L. Bruzzone and D. Fernàndez Prieto, "A Partially Unsupervised Approach to the Automatic Classification of Multitemporal Remote-Sensing Images," *Pattern Recognition Letters,* vol. 33, no. 9, pp. 1063-1071, 2002.

[38] L. Bruzzone and R. Cossu, "A Multiple-Cascade-Classifier System for a Robust and Partially Unsupervised Updating of Land-Cover Maps," *IEEE Trans. Geosciences and Remote Sensing,* vol. 40, no. 9, pp. 1984-1996, Sept. 2002.

[39] L. Bruzzone, R. Cossu, and G. Vernazza, "Combining Parametric and Non-Parametric Algorithms for a Partially Unsupervised Classification of Multitemporal Remote-Sensing Images," *Information Fusion,* vol. 3, no. 4, pp. 289-297, 2002.

[40] S. Tajudin and D. Landgrebe, "Robust Parameter Estimation for Mixture Model," *IEEE Trans. Geoscience and Remote Sensing,* vol. 38, pp. 439-445, 2000.

[41] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines.* Cambridge Univ. Press, 2000.

[42] J. Platt, "Fast Training of Support Vector Machines Using Sequential Minimal Optimization," *Advances in Kernel Methods: Support Vector Learning,* B. Schölkopf, C. Burges, and A. Smola, eds., pp. 185-208, MIT Press, 1998.

[43] http://ida.first.fraunhofer.de/projects/bci/competition_iii/, 2008.

[44] T. Lal, T. Hinterberger, G. Widman, M. Schröder, J. Hill, W. Rosenstiel, C. Elger, B. Schölkopf, and N. Birbaumer, "Methods Towards Invasive Human Brain Computer Interfaces," *Advances in Neural Information Processing Systems 17,* MIT Press, 2004.

[45] C. Toro, G. Deuschl, R. Thatcher, S. Sato, C. Kufta, and M. Hallett, "Event-Related Desynchronization and Movement-Related Cortical Potentials on the ECoG and EEG," *Electroencephalography and Clinical Neurophysiology,* vol. 93, pp. 380-389, 1994.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.

[46] C. Babiloni, F. Carducci, F. Cincotti, P.M. Rossini, C. Neuper, G. Pfurtscheller, and F. Babiloni, "Human Movement-Related Potentials vs Desynchronization of EEG Alpha Rhythm: A High-Resolution EEG Study," *Neuroimage,* vol. 10, pp. 658-665, 1999.

[47] Y. Wang, P. Berg, and M. Scherg, "Common Spatial Subspace Decomposition Applied to Analysis of Brain Responses Under Multiple Task Conditions: A Simulation Study," *Clinical Neurophysiology,* vol. 110, pp. 604-614, 1999.

Lorenzo Bruzzone received the laurea (MS) degree in electronic engineering (summa cum laude) and the PhD degree in telecommunications from the University of Genoa, Italy, in 1993 and 1998, respectively. He is currently a full professor of telecommunications at the University of Trento, Italy, where he teaches remote sensing, pattern recognition, radar, and electrical communications. His current research interests are in the areas of remote sensing, signal processing, and pattern recognition (analysis of multitemporal images, feature extraction and selection, classification, regression and estimation, data fusion, machine learning). He conducts and supervises research on these topics within the frameworks of several national and international projects. He is the author (or coauthor) of 86 scientific publications in refereed international journals (58 in IEEE journals), more than 140 papers in conference proceedings, and 11 book chapters. He is editor/coeditor of nine books/conference proceedings. He is a referee for many international journals and has served on the scientific committees of several international conferences. He is a member of the managing committee of the Italian Interuniversity Consortium on Telecommunications and a member of the scientific committee of the India-Italy Center for Advanced Research. Since 2009, he has been a member of the administrative committee of the IEEE Geoscience and Remote Sensing Society. He ranked first place in the Student Prize Paper Competition of the 1998 IEEE International Geoscience and Remote Sensing Symposium (Seattle, July 1998). He was a recipient of the Recognition of the *IEEE Transactions on Geoscience and Remote Sensing* Best Reviewers in 1999 and was a guest coeditor of several special issues of the *IEEE Transactions on Geoscience and Remote Sensing*. He was the general chair and cochair of the First and Second IEEE International Workshops on the Analysis of Multitemporal Remote Sensing Images (MultiTemp), and is currently a member of the permanent steering committee of this series of workshops. Since 2003, he has been the chair of the SPIE Conference on Image and Signal Processing for Remote Sensing. From 2004 to 2006, he served as an associate editor of the *IEEE Geoscience and Remote Sensing Letters* and currently is an associate editor for the *IEEE Transactions on Geoscience and Remote Sensing* and the *Canadian Journal of Remote Sensing*. In 2008 he was appointed a member of the joint NASA/ESA Science Definition Team for *Outer Planet Flagship Missions*. He is also a member of the International Association for Pattern Recognition and the Italian Association for Remote Sensing (AIT). He is a fellow of the IEEE.

Mattia Marconcini received the "laurea" (BS) and the "laurea specialistica" (MS) degrees in telecommunication engineering (summa cum laude) and the PhD degree in communication and information technologies from the University of Trento, Italy, in 2002, 2004, and 2008, respectively. He is presently with the Remote Sensing Laboratory in the Department of Information Engineering and Computer Science, University of Trento. His current research activities are in the area of machine learning, pattern recognition, and remote sensing. In particular, his interests are related to transfer learning and domain adaptation classification and image segmentation problems. He conducts research on these topics within the frameworks of several national and international projects. He was a finalist in the Student Prize Paper Competition of the 2007 IEEE International Geoscience and Remote Sensing Symposium (Barcelona, July 2007). He is a member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: UNIVERSITA TRENTO. Downloaded on April 28,2010 at 07:12:19 UTC from IEEE Xplore. Restrictions apply.