# Unsupervised Multi-Class Domain Adaptation: Theory, Algorithms, and Practice

Yabin Zhang*, Bin Deng*, Hui Tang, Lei Zhang, and Kui Jia

**Abstract**—In this paper, we study the formalism of unsupervised multi-class domain adaptation (multi-class UDA), which underlies some recent algorithms whose learning objectives are only motivated empirically. A Multi-Class Scoring Disagreement (MCSD) divergence is presented by aggregating the absolute margin violations in multi-class classification; the proposed MCSD is able to fully characterize the relations between any pair of multi-class scoring hypotheses. By using MCSD as a measure of domain distance, we develop a new domain adaptation bound for multi-class UDA as well as its data-dependent, probably approximately correct bound, which naturally suggest adversarial learning objectives to align conditional feature distributions across the source and target domains. Consequently, an algorithmic framework of Multi-class Domain-adversarial learning Networks (McDalNets) is developed, whose different instantiations via surrogate learning objectives either coincide with or resemble a few of recently popular methods, thus (partially) underscoring their practical effectiveness. Based on our same theory of multi-class UDA, we also introduce a new algorithm of Domain-Symmetric Networks (SymmNets), which is featured by a novel adversarial strategy of domain confusion and discrimination. SymmNets afford simple extensions that work equally well under the problem settings of either closed set, partial, or open set UDA. We conduct careful empirical studies to compare different algorithms of McDalNets and our newly introduced SymmNets. Experiments verify our theoretical analysis and show the efficacy of our proposed SymmNets. We make our implementation codes publicly available.

**Index Terms**—Domain adaptation, multi-class classification, adversarial training, partial or open set domain adaptation



## 1 Introduction

STANDARD machine learning assumes that training and test data are drawn from a same underlying distribution. As such, uniform convergence bounds guarantee the generalization of models learned on the training data for use of testing [1]. Although standard machine learning has achieved great success in various tasks [2], [3], [4], even with few training data [5], [6] or training data of multiple modalities [7], in many practical scenarios, one may encounter the situation that annotated training data can only be collected easily from one or several distributions that are related to the testing one; in order words, the target data of interest follow a distribution differing from the training source ones. A typical example in deep learning based image analysis is that one may annotate as many synthetic images as possible, but often fails to annotate even a single real image; it is expected to adapt the models learned from synthetic images for testing of real images. This problem setting falls in the realm of transfer learning or domain adaptation [8]. In this work, we focus particularly on unsupervised domain adaptation (UDA) where the target data are completely unlabeled.

In literature, theoretical studies on domain adaptation characterize the conditions under which classifiers trained on the labeled source data can be adapted for use on the target domain [9], [10], [11], [12]. For example, Ben-David *et al.* [10] propose a notion of distribution divergence induced by the hypothesis space of binary classifiers, based on which a bound of expected error on the target domain is thus developed; Mansour *et al.* [11] extend the zero-loss loss used in [10] to arbitrary loss functions of binary classification. These theoretical results motivate many of existing UDA algorithms, including the recently popular ones based on domain-adversarial training of deep networks [13], [14], [15], [16], [17]. A common motivation in them is to design adversarial objectives that concern with minimax optimization, in order to reduce the hypothesis-induced domain divergence via the learning of domain-invariant feature representations. While theoretical adaptation conditions are strictly derived under the setting of binary classification with analysis-amenable loss functions, practical algorithms easier to be optimized are often expected to be applied to the cases of multiple classes. In other words, learning objectives in many of the recent algorithms are only inspired by, rather than strictly derived from domain adaptation bounds in [10], [11].

This gap between theories and algorithms is recently studied in [18], where a notion of margin disparity discrepancy induced by pairs of multi-class scoring hypotheses is introduced to measure the divergence between domain distributions, thus extending theories in [10], [11] to a multi-class setting closer to practical algorithms. In spite of the being bridged gap, the scalar-valued margin function defined in [18] is deficient in terms of fully characterizing the disagreements between pairs of multi-class scoring hypotheses; as a result, the theory developed in [18] cannot well explain the effectiveness of a series of recent UDA algorithms [16], [17], [19], [20], [21] whose designs take relationships among all of the multiple classes into account, and thus promote a better alignment of conditional feature distributions across domains.

In this work, we are motivated to develop theories of unsupervised multi-class domain adaptation (multi-class UDA) that connect more closely with the aforementioned practical, state-of-the-art algorithms [16], [17], [19], [20], [21]. Technically, we are inspired by the multi-class classification framework of Dogan *et*


- *Y. Zhang, B. Deng, H. Tang, and K. Jia are with the School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China. E-mails: zhang.yabin@mail.scut.edu.cn, bindeng@szu.edu.cn, eehuitang@mail.scut.edu.cn, kuijia@scut.edu.cn.* *These two authors contribute equally. Correspondence to: K. Jia*
- *L. Zhang is with the Department of Computing, the Hong Kong Polytechnic University, HongKong. E-mail: cslzhang@comp.polyu.edu.hk*

*al.* [22], which aggregates violations of class-wise absolute margins as a single loss, and propose a notion of matrix-formed *Multi-Class Scoring Disagreement (MCSD)* that takes a full account of the element-wise disagreements between any pair of multi-class scoring hypotheses. MCSDs defined over domain distributions induce a novel *MCSD divergence*, measuring distribution distance between the source and target domains. Based on MCSD divergence, we develop a new adaptation bound for multi-class UDA; a data-dependent, probably approximately correct (PAC) bound is also developed using the notion of Rademacher complexity. We connect our results with existing theories of either binary [10] or multi-class UDA [18] by introducing degenerate versions of MCSD divergence and the corresponding domain adaptation bounds. We show advantages of MCSD divergence over these degenerate versions.

The bounds derived in our theory of multi-class UDA based on either MCSD divergence or its degenerate versions naturally suggest adversarial objectives of minimax optimization, which promote learning of conditional feature distributions invariant across the source and target domains. We term such an algorithmic framework as *Multi-class Domain-adversarial learning Networks (McDalNets)*, as illustrated in Figure 2. While it is difficult to directly optimize the objectives of McDalNets, we show that a few optimization-friendly surrogate objectives instantiate the recently popular methods [16], [18], thus (partially) explaining the underlying mechanisms of their effectiveness. In addition to McDalNets, we introduce a new algorithm of *Domain-Symmetric Networks (SymmNets)* motivated from our same theory of multi-class UDA; Figure 3 gives an illustration. The proposed SymmNets are featured by a domain confusion and discrimination strategy that ideally achieves the same theoretically derived learning objective.

While most of the theories and algorithms presented in the paper concern with *closed set UDA*, where the two domains share a same label space, one may be also interested in other variant settings, such as the *partial* [23], [24], [25], [26], [27] or *open set* [28], [29] UDA. In this work, we present simple extensions of SymmNets that are able to achieve partial or open set UDA as well. We conduct careful ablation studies to compare different algorithms of McDalNets, its degenerate versions, and also our newly introduced SymmNets. As shown in Table 2, experiments on five commonly used benchmarks show that algorithms of McDalNets consistently improve over its degenerate versions, certifying the usefulness of fully characterizing disagreements between pairs of scoring hypotheses in multi-class UDA. Experiments under the settings of closed set, partial, and open set UDA also verify the effectiveness of our proposed SymmNets empirically.

## 1.1 Relations with Existing Works

### *1.1.1 Domain Adaptation Theories*

In literature, these exist theoretical domain adaptation results concerning mostly with the classification problem and also with regression [11], [30], [31]. For classification, these results consider either a setting where target data are partially labeled [32], [33], or the standard UDA setting from the perspectives of optimal transportation [12], [34] or hypothesis-induced domain divergence [9], [10], [11], [18], [35]. We focus on the later line of theories that are closely related to our contributed one.

Seminal domain adaptation theories [9], [10], [11] bound the expected target error for binary classification with terms characterizing the expected source error, the domain distance under certain metrics of distribution divergence, and constant ones depending on the capacity of hypothesis space; the term of domain distance differentiates these theoretical bounds. For example, Ben-David *et al.* [9], [10] propose for the binary classification the $\mathcal{H}$-distance/$\mathcal{H}\Delta\mathcal{H}$-distance by characterizing the disagreement of class predictions between any pair of hypotheses with the zero-one loss; Mansour *et al.* [11] introduce a notion of discrepancy distance by extending the zero-one loss of [9] to general loss functions of binary classification; by fixing one hypothesis of [11] to the ideal source minimizer, Kuroki *et al.* [35] propose a more tractable source-guided discrepancy (i.e., S-disc). Although many of recent algorithms [13], [14], [16], [17] are motivated from seminal theories [9], [10], the gap between theories of binary classification and practical algorithms of multi-class classification remains. To reduce the gap, Zhang *et al.* [18] extend the theories of [10], [11] to the case of multiple classes based on a notion of margin disparity discrepancy (MDD); MDD is a measure of domain distance built on a scalar-valued margin disparity function that characterizes the difference of multi-class scoring hypotheses.

While both of our MCSD and those of [10], [11], [18] are based on characterization of the disagreements between any pair of labeling/scoring hypotheses, our MCSD is capable of characterizing them at a finer level, especially in the multi-class setting (cf. Figure 1). Technically, our MCSD characterizes the element-wise disagreements of multi-class scoring hypotheses by aggregating violations of class-wise absolute margins; in contrast, the zero-one loss of [10] only characterizes the labeling disagreement, and the margin disparity of [18] improves over [10] with a scoring disagreement via a scalar-valued, relative margin. In fact, we show that degenerate versions of our MCSD connect more closely with those of [10], [18]. Consequently, the domain divergence induced by our MCSD can better explain the effectiveness of a series of recent UDA algorithms [16], [17], [19], [20], [21], whose designs take the relations of scores of all the multiple classes into account.

### *1.1.2 Algorithms of Multi-Class Domain Adaptation*

Existing algorithms of multi-class UDA are mainly motivated by learning domain-invariant feature representations [13], [14], [16], [17], [19], [20], [21], [36], [37], [38], [39], [40], or by minimizing the domain discrepancy in the image space via image generation [41], [42]. We briefly review the former line of algorithms, focusing on those based on the strategy of adversarial training.

Motivated to minimize the domain divergence measured by $\mathcal{H}\Delta\mathcal{H}$-distance of [10], Ganin *et al.* [13] introduce the first strategy of domain-adversarial training of neural networks (DANN), where a binary classifier is adopted as the domain discriminator, and domain distance is minimized by learning features of the two domains in a manner adversarial to the domain discriminator. Tzeng *et al.* [14] summarize three implementation manners of adversarial objective, including minimax [13], confusion [43], and GAN [44]. Domain discriminator of binary classifier enables learning alignment of marginal feature distributions across domains, but it is ineffective for alignment of conditional feature distributions, which is necessary in the practical UDA problems of multi-class setting. Recent methods [16], [17], [18], [19], [20], [21] strive to overcome this limitation by playing adversarial games between two classifiers. More specifically, Saito *et al.* [16] adopt the maximum $L_1$ distance of output probabilities of two symmetric classifiers as the surrogate domain discrepancy; Lee *et al.* [20] replace the $L_1$ distance in [16] with the Wasserstein distance [45], taking the advantage of its geometrical characterization; in [18], two classifiers are used asymmetrically to estimate the conditional feature distributions with marginal loss; in [19], two task classifiers are introduced implicitly by applying two random dropout to a same task classifier; a classifier concatenated by two task classifiers is adopted to implement the adversarial training objective in [17], [21].

Motivated by the domain adaptation bounds to be presented in Section 2, we propose an algorithmic framework of McDalNets whose optimization-friendly surrogate objectives instantiate these recently popular methods [13], [16], [18] (cf. Section 3.1), thus (partially) explaining the underlying mechanisms of their effectiveness. We also introduce a new algorithm of SymmNets whose learning objective aligns with our developed theoretical bound as well (cf. Section 3.2).

#### *1.1.3 Variants of Problem Settings*

The theories and algorithms discussed so far apply to the problem setting of closed set UDA, where a shared label space across domains is assumed. There exist other variant settings, e.g., partial [23] or open set [28] UDA; we discuss these settings and the corresponding methods as follows.

The setting of partial UDA assumes that classes of the target domain constitutes an unknown subset of those of the source domain. To address the challenge brought by this partial class coverage, a typical strategy is to weight source instances using collective prediction evidence of target instances [23], [25], [26], [27]. Simply extending our SymmNets with a weighting scheme empirically gives excellent results.

The setting of open set UDA assumes that both the source and target domains contain certain classes that are exclusive to each other, where for simplicity all the unshared classes in each domain are aggregated as a single (super-) unknown class. A key issue to extend methods of closed set UDA for use in the open set setting is to design appropriate criteria that reject target instances of unshared classes, for which Busto *et al.* [28] adopt a predefined distance threshold, and Saito *et al.* [29] learn the rejection automatically via adversarial training. Our algorithm of SymmNets is flexible to be applied to open set UDA, simply by adding an additional output neuron to the task classifier, which is responsible for the aggregated super-class, while keeping other algorithmic ingredients fixed.

### 1.2 Contributions

Many of recent algorithms for multi-class UDA [16], [19], [20], [21], including our preliminary work of SymmNets [17], rely on an adversarial strategy that learns to align conditional feature distributions across domains via a full account of relationships among hypotheses of classifiers. While these algorithms are inspired by classical domain adaptation theories [9], [10], [11], their learning objectives are largely designed empirically; as such, the connections between theories and algorithms remain loose. The present paper aims to connect these algorithms by formalizing a theory of multi-class UDA, which underlies these algorithms with a framework that also inspires new algorithms. We summarize our technical contributions as follows.

- We propose to aggregate violations of absolute margin functions to define a notion of matrix-formed *Multi-Class Scoring Disagreement (MCSD)*, which enables to fully characterize the relations between any pair of scoring hypotheses. Based on the induced *MCSD divergence* as a measure of domain distance, we develop a new adaptation bound for multi-class UDA; a data-dependent PAC bound is also developed using the notion of Rademacher complexity. We connect our results with existing theories of either binary or multi-class UDA by introducing degenerate versions of MCSD divergence and the corresponding adaptation bounds.
- Our developed theories naturally suggest adversarial objectives to learn aligned conditional feature distributions across domains; we term such an algorithmic framework based on deep networks as *Multi-class Domain-adversarial learning Networks (McDalNets)*. We show that different instantiations of McDalNets via surrogate learning objectives either coincide with or resemble some recently popular methods, thus (partially) underscoring their practical efficacy. We also introduce a new algorithm of *Domain-Symmetric Networks (SymmNets-V2)* based on our same theory of multi-class UDA, which improves over *SymmNets-V1* proposed in our preliminary work.
- While theories and algorithms presented in the paper are mostly concerned with the problem setting of closed set UDA, we also present simple extensions of SymmNets that work equally well under the settings of partial or open set UDA. We conduct careful ablation studies to compare different algorithms of McDalNets, its degenerate versions, and our newly introduced SymmNets. Experiments on commonly used benchmarks show the superiority of McDalNets and SymmNets over the degenerate counterparts, certifying the effectiveness of fully characterizing disagreements between pairs of scoring hypotheses in multi-class UDA. Codes are available at https://github.com/YBZh/MultiClassDA.

## 2 A Theory of Unsupervised Multi-Class Domain Adaptation

We present in this section a theory of unsupervised multi-class domain adaptation (multi-class UDA). Our theoretical derivations follow [10], [11], which are mostly concerned with binary classification, but with a key novelty of measuring the distance between domain distributions using a divergence that fully characterizes the relations between different hypotheses of multi-class classification. We also present degenerate versions of the proposed distribution divergence to connect with theoretical results developed in the literature. We start with basic notations of multi-class UDA. All proofs are given in the appendices.

### 2.1 Learning Setup

For a standard learning setting of multi-class classification, learners receive samples from a distribution $D$ over $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is the instance space and $\mathcal{Y} = \{1, ..., K\}$ is the label space. We also write $D_x$ as the corresponding marginal distribution over $\mathcal{X}$. Multi-class UDA assumes two different but related distributions over $\mathcal{X} \times \mathcal{Y}$, namely the source one $P$ and target one $Q$. Learners receive $n_s$ labeled examples $\{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ drawn i.i.d. from $P$ and $n_t$ unlabeled examples $\{\boldsymbol{x}_j^t\}_{j=1}^{n_t}$ drawn i.i.d. from $Q_x$. The goal of multi-class UDA is to identify a labeling hypothesis $h : \mathcal{X} \to \mathcal{Y}$ from a space $\mathcal{H}$ such that the following *expected error* over the target distribution is minimized

$$\mathcal{E}_Q(h) := \mathbb{E}_{(\boldsymbol{x},y)\sim Q} L(h(\boldsymbol{x}), y), \tag{1}$$

where $L$ is a properly defined loss function. For ease of theoretical analysis, Ben-David *et al.* [9], [10] assume $L$ as a 0-1 loss of the form $\mathbb{1}[h(\boldsymbol{x}) \neq y]$, where $\mathbb{1}$ is the indicator function, which is extended in [11] as general loss functions of binary classification. Domain adaptation theories [10], [11], [18] typically bound the expected target error (1) using derived meaningful quantities.

Consider a space $\mathcal{F}$ that contains scoring functions $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|} = \mathbb{R}^K$, which induces a labeling function $h_{\boldsymbol{f}} : \boldsymbol{x} \to \arg\max_{k \in \mathcal{Y}} f_k(\boldsymbol{x})$, where $f_k$ denotes the $k^{th}$ component of the vector-valued function $\boldsymbol{f}$. Adding a same function $g : \mathcal{X} \to \mathbb{R}$ to all components $f_k$ of $\boldsymbol{f}$ does not change classification decision, since $\arg\max_{k \in \mathcal{Y}} f_k(\boldsymbol{x}) = \arg\max_{k \in \mathcal{Y}}(f_k(\boldsymbol{x}) + g(\boldsymbol{x}))$, which could be problematic for obtaining unique solutions of scoring functions. Similar to [22], we fix this issue by enforcing the sum-to-zero constraint $\sum_{k=1}^{K} f_k(\boldsymbol{x}) = 0$ to the scoring functions.

### 2.2 Domain Distribution Divergence and Adaptation Bounds based on Multi-Class Scoring Disagreement

Unsupervised domain adaptation is made possible by assuming the closeness between the distributions $P$ and $Q$; otherwise classifiers learned from the labeled source data would be less relevant for the classification of target data. Some measure of distribution distances thus becomes a crucial factor to develop either UDA theories or the corresponding algorithms.

In the seminal work [10], a key innovation is the introduction of a distribution distance induced by a hypothesis space $\mathcal{H}^{\{0,1\}}$ of binary classification

$$d_{0\text{-}1}(P_x, Q_x) := \sup_{h,h' \in \mathcal{H}^{\{0,1\}}} \left| \mathbb{E}_{Q_x} \mathbb{1}[h \neq h'] - \mathbb{E}_{P_x} \mathbb{1}[h \neq h'] \right|, \tag{2}$$

where the *disagreement* between $h$ and $h'$ in fact specifies a measurable subset $\{\boldsymbol{x} \in \mathcal{X} | h(\boldsymbol{x}) \neq h'(\boldsymbol{x})\}$, and the distribution distance (termed $\mathcal{H}\Delta\mathcal{H}$-divergence in [10]) between $P_x$ and $Q_x$ is measured on the subsets by taking the supremum over all pairs of $h, h' \in \mathcal{H}^{\{0,1\}}$. Compared with the simple $\ell_1$ distribution divergence, the distance (2) is more relevant to the problem of domain adaptation and can be estimated from finite samples for an $\mathcal{H}^{\{0,1\}}$ of fixed VC dimension [10]. Based on the same idea of characterizing the *hypothesis disagreement*, Mansour *et al.* [11] extend the 0-1 loss based distance (2) to incorporate general loss functions $L$, giving rise to the distance (termed *discrepancy distance* in [11])

$$d_L(P_x, Q_x) := \sup_{h,h' \in \mathcal{H}} \left| \mathbb{E}_{Q_x} L(h, h') - \mathbb{E}_{P_x} L(h, h') \right|. \tag{3}$$

Note that (3) is symmetric and satisfies triangle inequality, but it does not strictly define a distance since it is possible that $d_L(P_x, Q_x) = 0$ for $P_x \neq Q_x$.

In spite of being more general, the distance (3) applies only to UDA problems of binary classification. To develop multi-class UDA, disagreement of multi-class hypotheses should be taken into account. The key issue here is to extend binary loss functions $L$, especially *margin-based ones*, to the case of multiple classes [46]. In literature, there exists no a canonical formulation of multi-class classification; various formulation variants have been proposed depending on different notions of multi-class margin and margin based loss [47], [48], [49], where margins are usually defined either by comparing components $\{f_k\}_{k=1}^{K}$ of a $K$-class scoring function $\boldsymbol{f}$ (i.e., relative margins), or directly on the components $\{f_k\}_{k=1}^{K}$ themselves (i.e., absolute margins). Notably, Dogan *et al.* [22] unify these variants by a framework that decomposes a multi-class loss function into class-wise margins and margin violations (i.e., large-margin losses), and then aggregates these violations as a single loss value.

Motivated by this framework, we propose in this paper a matrix-formed *multi-class scoring disagreement (MCSD)* to fully characterize the difference between any pair of scoring functions $\boldsymbol{f}', \boldsymbol{f}'' \in \mathcal{F}$, which is later used to define a distribution distance tailored to multi-class UDA.

**Definition 1 (Absolute margin function).** *For a multi-class scoring function $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|} = \mathbb{R}^K$, its absolute margin function $\boldsymbol{\mu} : \mathbb{R}^K \times \mathcal{Y} \to \mathbb{R}^K$ is defined on an example $(\boldsymbol{x}, y)$ as*

$$\mu_k(f(\boldsymbol{x}), y) = \begin{cases} +f_k(\boldsymbol{x}), & k = y \\ -f_k(\boldsymbol{x}), & k \in \mathcal{Y} \setminus \{y\} \end{cases}. \tag{4}$$

Given the sum-to-zero constraint $\sum_{k=1}^{K} f_k(\boldsymbol{x}) = 0$, the defined margin function enjoys the following properties [22].

- $\mu_y(\boldsymbol{f}(\boldsymbol{x}), y)$ is non-decreasing w.r.t. $f_y(\boldsymbol{x})$ ,
- $\mu_k(\boldsymbol{f}(\boldsymbol{x}), y)$ is non-increasing w.r.t. $f_k(\boldsymbol{x}) \; \forall \, k \in \mathcal{Y} \setminus \{y\}$
- When $\mu_k(\boldsymbol{f}(\boldsymbol{x}), y) \geq 0 \; \forall \, k \in \mathcal{Y}$ and $\exists k \in \mathcal{Y}$ such that $\mu_k(\boldsymbol{f}(\boldsymbol{x}), y) > 0$, we have $\arg\max_{k \in \mathcal{Y}} f_k(\boldsymbol{x}) = y$ .

The third property characterizes correct classification by checking non-negativeness/positiveness of absolute margins. To develop MCSD, we consider the following ramp loss to penalize margin violations

$$\Phi_\rho(x) := \begin{cases} 0, & \rho \leq x \\ 1 - x/\rho, & 0 < x < \rho \\ 1, & x \leq 0 \end{cases}. \tag{5}$$

For $\rho > 0$ and a distribution $D$ over $\mathbb{R}$, ramp loss has the nice property of $\mathbb{E}_{x \sim D} \Phi_\rho(x) \geq \mathbb{E}_{x \sim D} \mathbb{1}[x \leq 0]$, which is important to bound the target error $\mathcal{E}_Q(h_{\boldsymbol{f}})$ using margin-based loss functions defined over the scoring function $\boldsymbol{f}$.

**Definition 2 (Multi-class scoring disagreement).** *For a pair of scoring functions $\boldsymbol{f}', \boldsymbol{f}'' \in \mathcal{F}$, the multi-class scoring disagreement (MCSD) is defined with respect to a distribution $D$ over the domain $\mathcal{X}$ as*

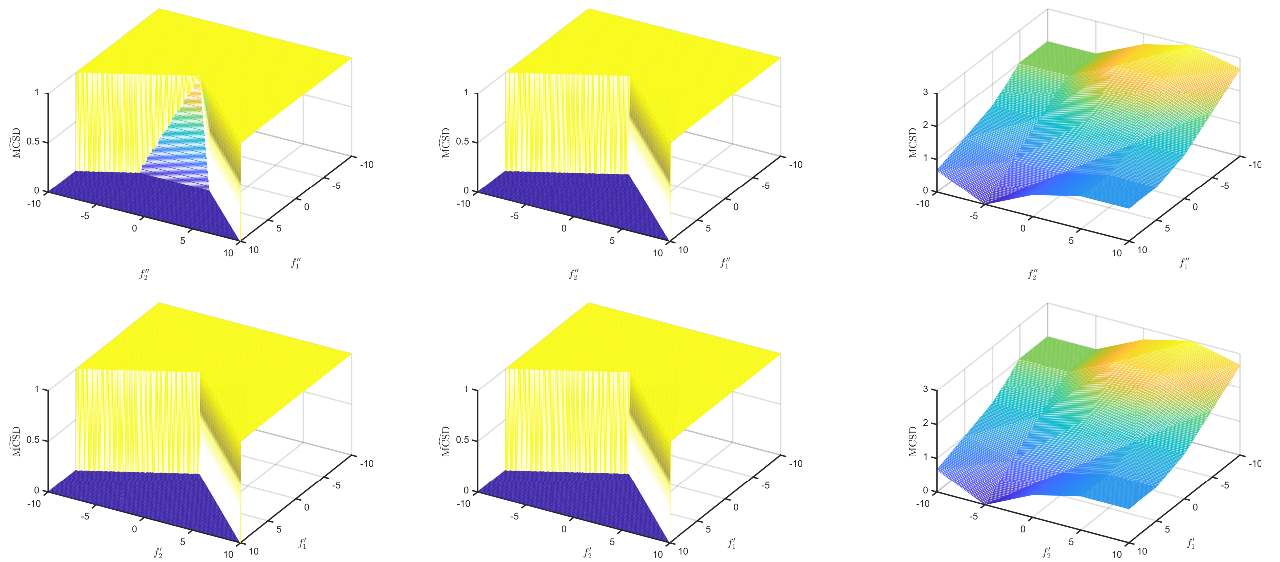
$$\text{MCSD}_D^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'') := \frac{1}{K} \mathbb{E}_{\boldsymbol{x} \sim D} \| \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}''(\boldsymbol{x})) \|_1, \tag{6}$$

*where $\| \cdot \|_1$ is the $L_1$ norm and $\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) \in [0,1]^{K \times K}$ is the matrix of absolute margin violations defined as*

$$\boldsymbol{M}_{i,j}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) = \Phi_\rho(\mu_i(\boldsymbol{f}(\boldsymbol{x}), j)). \tag{7}$$

Each column $\boldsymbol{M}_{:,k}^{(\rho)}$ of the matrix $\boldsymbol{M}^{(\rho)}$ computes violations of the absolute margin function $\boldsymbol{\mu}(\boldsymbol{f}(\cdot), k)$ w.r.t. a class $k \in \mathcal{Y}$, and the corresponding $\|\boldsymbol{M}_{:,k}^{(\rho)}(\boldsymbol{f}') - \boldsymbol{M}_{:,k}^{(\rho)}(\boldsymbol{f}'')\|_1$ measures difference of margin violations between the scoring functions $\boldsymbol{f}'$ and $\boldsymbol{f}''$. The proposed MCSD (6) is based on absolute value aggregation of these disagreements. To have an intuitive understanding of the behaviors of $\boldsymbol{f}'$, $\boldsymbol{f}''$, and the $\text{MCSD}_D^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'')$, we plot in Figure 1(c) the value of $\text{MCSD}_D^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'')$ (firing on a single instance $\boldsymbol{x}$) in the case of $K = 3$ and $\rho = 5$, by fixing either $\boldsymbol{f}'(\boldsymbol{x})$ or $\boldsymbol{f}''(\boldsymbol{x})$ and using the other as the argument.

We have the following definition of distribution distance based on the proposed MCSD.



(a) (b) (c)

Fig. 1. Plots of various disagreements between two scoring functions $\boldsymbol{f}'$ and $\boldsymbol{f}''$ firing on a single instance $\boldsymbol{x}$ in a case of $K = 3$ and $\rho = 5$, where the scoring functions satisfy the sum-to-zero constraint. Top row: fix $\boldsymbol{f}'(\boldsymbol{x})$ to be $[10; -5; -5]$ and use $\boldsymbol{f}''(\boldsymbol{x}) = [f''_1; f''_2; -(f''_1 + f''_2)]$ as the argument; Bottom row: fix $\boldsymbol{f}''(\boldsymbol{x})$ to be $[10; -5; -5]$ and use $\boldsymbol{f}'(\boldsymbol{x}) = [f'_1; f'_2; -(f'_1 + f'_2)]$ as the argument. (a) The degenerate $\widetilde{\text{MCSD}}$ defined as (12), which can be considered as a variant of margin disparity (MD) [18]; (b) the degenerate $\widehat{\text{MCSD}}$ defined as (13), which is equivalent to the $\mathcal{H}\Delta\mathcal{H}$-distance proposed in [10]; (c) our proposed MCSD (6).

**Definition 3 (MCSD divergence).** *Given the definition of MCSD, we define the divergence between distributions $P_x$ and $Q_x$ over the domain $\mathcal{X}$ with respect to the space $\mathcal{F}$ as*

$$d^{(\rho)}_{MCSD}(P_x, Q_x) := \sup_{\boldsymbol{f}', \boldsymbol{f}'' \in \mathcal{F}} [\text{MCSD}^{(\rho)}_{Q_x}(\boldsymbol{f}', \boldsymbol{f}'') - \text{MCSD}^{(\rho)}_{P_x}(\boldsymbol{f}', \boldsymbol{f}'')]. \quad (8)$$

The proposed MCSD divergence (8) satisfies the properties of non-negativity and triangle inequality, but it is not symmetric w.r.t. $P_x$ and $Q_x$. Nevertheless, we show its usefulness for multi-class UDA by developing the following bound.

**Theorem 1.** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$, the following holds over the source and target distributions $P$ and $Q$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + d^{(\rho)}_{MCSD}(P_x, Q_x) + \lambda, \quad (9)$$

*where the constant $\lambda = \mathcal{E}^{(\rho)}_P(\boldsymbol{f}^*) + \mathcal{E}^{(\rho)}_Q(\boldsymbol{f}^*)$ with $\boldsymbol{f}^* = \arg\min_{\boldsymbol{f} \in \mathcal{F}} \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + \mathcal{E}^{(\rho)}_Q(\boldsymbol{f})$, and*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) := \mathbb{E}_{(\boldsymbol{x},y)\sim Q} \mathbb{1}[h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y], \quad (10)$$

$$\mathcal{E}^{(\rho)}_P(\boldsymbol{f}) := \mathbb{E}_{(\boldsymbol{x},y)\sim P} \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}), y)). \quad (11)$$

Theorem 1 has a form similar to the domain adaptation bound proposed by Ben-David *et al.* [10]; differently, it replies on absolute margin based loss function and MCSD divergence to achieve a full characterization of multi-class UDA. As the bound (9) suggests, given fixed $\lambda$, the expected target error $\mathcal{E}_Q(h_{\boldsymbol{f}})$ is determined by the distance $d^{(\rho)}_{MCSD}(P_x, Q_x)$ (and the expected loss $\mathcal{E}^{(\rho)}_P(\boldsymbol{f})$ over the source domain); smaller $d^{(\rho)}_{MCSD}(P_x, Q_x)$ indicates better adaptation of multi-class UDA. To connect with domain adaptation bounds developed in literature, notably those proposed in [10], [18], we first present the following two *scalar-valued, degenerate* versions of MCSD as

$$\widetilde{\text{MCSD}}^{(\rho)}_D(\boldsymbol{f}', \boldsymbol{f}'') := \mathbb{E}_{\boldsymbol{x}\sim D} \Phi_{\rho/2}[\boldsymbol{\mu}_{h_{\boldsymbol{f}''(\boldsymbol{x})}}(\boldsymbol{f}''(\boldsymbol{x}), h_{\boldsymbol{f}'}(\boldsymbol{x}))], \quad (12)$$

$$\widehat{\text{MCSD}}^{(\rho)}_D(\boldsymbol{f}', \boldsymbol{f}'') := \mathbb{E}_{\boldsymbol{x}\sim D} \mathbb{1}[\Phi_\rho[\boldsymbol{\mu}_{h_{\boldsymbol{f}''(\boldsymbol{x})}}(\boldsymbol{f}''(\boldsymbol{x}), h_{\boldsymbol{f}'}(\boldsymbol{x}))] = 1], \quad (13)$$

which gives the degenerate distribution divergences $d^{(\rho)}_{\widetilde{MCSD}}$ and $d^{(\rho)}_{\widehat{MCSD}}$. We have the following propositions for the degenerate versions of MCSD.

**Proposition 1.** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + d^{(\rho)}_{\widetilde{MCSD}}(P_x, Q_x) + \lambda, \quad (14)$$

**Proposition 2.** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + d^{(\rho)}_{\widehat{MCSD}}(P_x, Q_x) + \lambda. \quad (15)$$

Compared with the scalar-valued, degenerate versions (12) and (13) (and also their corresponding ones in [18] and [10]), our matrix-formed MCSD (6) is able to characterize finer details of the scoring disagreement, as illustrated in Figure 1. Consequently, domain adaptation bound developed on the induced MCSD divergence would be beneficial to characterize multi-class UDA in a finer manner, which possibly inspires better UDA algorithms.

### 2.3 A Data-Dependent Multi-Class Domain Adaptation Bound

In this section, we extend the multi-class UDA bound in Theorem 1 to a PAC bound, by showing that both terms of $\mathcal{E}^{(\rho)}_P(\boldsymbol{f})$ and $d^{(\rho)}_{MCSD}(P_x, Q_x)$ can be estimated from finite samples. Our extension is based on the following notion of Rademacher complexity.

**Definition 4 (Rademacher complexity).** *Let $\mathcal{G}$ be a space of functions mapping from $\mathcal{Z}$ to $[a, b]$ and $\mathcal{S} = \{z_1, ..., z_m\}$ a fixed sample of size $m$ draw from the distribution $D$ over $\mathcal{Z}$. Then, the empirical Rademacher complexity of $\mathcal{G}$ with respect to the sample $\mathcal{S}$ is defined as*

$$\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G}) := \frac{1}{m}\mathbb{E}_{\sigma} \sup_{g\in\mathcal{G}} \sum_{i=1}^{m} \sigma_i g(z_i), \tag{16}$$

*where $\{\sigma_i\}_{i=1}^{m}$ are independent uniform random variables taking values in $\{-1, +1\}$. The Rademacher complexity of $\mathcal{G}$ is defined as the expectation of $\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G})$ over all samples of size $m$*

$$\mathfrak{R}_{m,D}(\mathcal{G}) := \mathbb{E}_{\mathcal{S}\sim D^m}\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G}). \tag{17}$$

The Rademacher complexity captures the richness of a function space by measuring the degree to which it can fit random noise. The empirical version has the additional advantage that it is data-dependent and can be estimated from finite samples. We have the following definition from [50], before introducing our Rademacher complexity based adaptation bound.

**Definition 5.** *For a space $\mathcal{F}$ of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^{|\mathcal{Y}|}$, we define*

$$\Pi_1(\mathcal{F}) := \{\boldsymbol{x} \to f_k(\boldsymbol{x}) | k \in \mathcal{Y}, \boldsymbol{f} \in \mathcal{F}\}. \tag{18}$$

The defined space $\Pi_1(\mathcal{F})$ can be seen as the union of projections of $\mathcal{F}$ onto each output dimension.

**Theorem 2.** *Let $\mathcal{F}$ be the space of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^K$. Let $P$ and $Q$ be the source and target distributions over $\mathcal{X}\times\mathcal{Y}$, and $P_x$ and $Q_x$ be the corresponding marginal distributions over $\mathcal{X}$. Let $\widehat{P}$ and $\widehat{Q}_x$ denote the corresponding empirical distributions for a sample $\mathcal{S} = \{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ and a sample $\mathcal{T} = \{\boldsymbol{x}_j^t\}_{j=1}^{n_t}$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1 - 3\delta$, the following holds for all $\boldsymbol{f} \in \mathcal{F}$*

$$\begin{aligned}\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq & \mathcal{E}_{\widehat{P}}^{(\rho)}(\boldsymbol{f}) + d_{MCSD}^{(\rho)}(\widehat{P}_x, \widehat{Q}_x) \\ & + (\frac{2K^2}{\rho} + \frac{4K}{\rho})\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1\mathcal{F}) + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{T}}(\Pi_1(\mathcal{F})) \\ & + 6K\sqrt{\frac{\log\frac{4}{\delta}}{2n_s}} + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_t}} + \lambda,\end{aligned} \tag{19}$$

*where the constant $\lambda = \min_{\boldsymbol{f}\in\mathcal{F}} \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + \mathcal{E}_Q^{(\rho)}(\boldsymbol{f})$, and*

$$\mathcal{E}_{\widehat{P}}^{(\rho)}(\boldsymbol{f}) := \frac{1}{n_s}\sum_{i=1}^{n_s}\sum_{k=1}^{K}\Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}_i^s), y_i^s)). \tag{20}$$

## 3 Connecting Theory with Algorithms

In the derived bound (19) of multi-class UDA, the constant $\lambda$ and complexity terms are assumed to be fixed given the hypothesis space $\mathcal{F}$. To minimize the expected target error $\mathcal{E}_Q(h_{\boldsymbol{f}})$, one is tempted to minimize the first two terms of $\mathcal{E}_{\widehat{P}}^{(\rho)}(\boldsymbol{f})$ and $d_{MCSD}^{(\rho)}(\widehat{P}_x, \widehat{Q}_x)$. In practice, a function $\psi$ of feature extractor is typically used to lift the input space $\mathcal{X}$ to a feature space $\mathcal{X}^\psi = \{\psi(\boldsymbol{x}) | x \in \mathcal{X}\}$, where with a slight abuse of notation, the space $\mathcal{F}$ of scoring functions $\boldsymbol{f} : \mathcal{X}^\psi \to \mathbb{R}^{|\mathcal{Y}|} = \mathbb{R}^K$ and the induced labeling functions $h_{\boldsymbol{f}} : \psi(\boldsymbol{x}) \to \arg\max_{k\in\mathcal{Y}} f_k(\psi(\boldsymbol{x}))$ are again well defined. We correspondingly write as $P^\psi$ and $Q^\psi$ for the source and target distributions over the lifted domain $\mathcal{X}^\psi \times \mathcal{Y}$, and their empirical (or marginal) versions as $\widehat{P}^\psi$ and $\widehat{Q}^\psi$ (or $P_x^\psi$ and $Q_x^\psi$). The function $\psi$ is typically implemented as a learnable deep network.

Given learnable $\psi$, minimizing right hand side of the bound (19) can be achieved by identifying $\psi^*$ that minimizes $d_{MCSD}^{(\rho)}(\widehat{P}_x^\psi, \widehat{Q}_x^\psi)$, and additionally identifying $\boldsymbol{f}^*$ with $\psi^*$ that minimize $\mathcal{E}_{\widehat{P}^\psi}^{(\rho)}(\boldsymbol{f})$. Recall that the MCSD divergence (8) is defined by taking supremum over all pairs of $\boldsymbol{f}', \boldsymbol{f}'' \in \mathcal{F}$. Spelling $d_{MCSD}^{(\rho)}(\widehat{P}_x^\psi, \widehat{Q}_x^\psi)$ out gives the following general objective of minimax optimization for multi-class UDA

$$\begin{aligned}&\min_{\boldsymbol{f},\psi} \mathcal{E}_{\widehat{P}^\psi}^{(\rho)}(\boldsymbol{f}) + [\mathrm{MCSD}_{\widehat{Q}_x^\psi}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'') - \mathrm{MCSD}_{\widehat{P}_x^\psi}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'')], \\ &\max_{\boldsymbol{f}',\boldsymbol{f}''} [\mathrm{MCSD}_{\widehat{Q}_x^\psi}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'') - \mathrm{MCSD}_{\widehat{P}_x^\psi}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'')],\end{aligned} \tag{21}$$

which suggests an adversarial learning strategy to promote domain-invariant conditional feature distributions via the learned $\psi$, thus extending [13] to account for multi-class UDA. We term the general algorithm (21) via the adversarial learning strategy as *Multi-Class Domain-Adversarial Learning Networks (McDalNets)*. Figure 2 gives an architectural illustration, where the scoring function $\boldsymbol{f}$ is for the multi-class classification task of interest, and $\boldsymbol{f}'$ and $\boldsymbol{f}''$ are auxiliary functions for learning of $\psi$. Since $\boldsymbol{f}$, $\boldsymbol{f}'$, and $\boldsymbol{f}''$ contain all the parameters of classifiers, we also use them to respectively refer to the task and auxiliary classifiers.

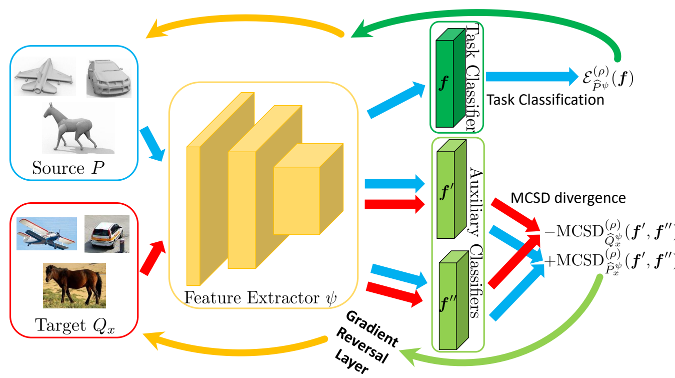



Fig. 2. An architectural illustration of Multi-Class Domain-Adversarial Learning Networks (McDalNets), which are motivated from the theoretically derived objective (21). The gradient reversal layer is adopted here to implement the adversarial objective; we note that other implementations (e.g., those discussed in [14]) would apply as well.

### 3.1 Different Algorithms of Multi-Class Domain-Adversarial Learning Networks

The proposed MCSD divergence is amenable to the theoretical analysis of multi-class UDA. However, it is difficult to directly optimize the MCSD based problem (21) via stochastic gradient descent (SGD), due to the use of ramp loss $\Phi_\rho$ in MCSD (6) that causes an issue of vanishing gradients. [1] To develop specific algorithms of McDalNets that are optimization friendly, we consider surrogate functions of MCSD (6), which are easier to be trained by SGD and also able to characterize the disagreements

1. We have tried to train a multi-class domain-adversarial learning network (illustrated in Figure 2) with the exact objective (21). However, it turns out that the optimization stagnates after a few iterations, since absolute values of the outputs of scoring functions increase over the predefined $\rho$, and the gradients thus vanish.

of all $K$ pairs of the corresponding elements in scoring functions $\boldsymbol{f}', \boldsymbol{f}'' \in \mathcal{F}$. These surrogates give the following objectives of specific algorithms

$$\min_{\boldsymbol{f},\psi} \mathcal{L}_{\widehat{P}^\psi}(\boldsymbol{f}) + [\text{SurMCSD}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') - \text{SurMCSD}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'')],$$
$$\max_{\boldsymbol{f}',\boldsymbol{f}''} [\text{SurMCSD}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') - \text{SurMCSD}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'')], \quad (22)$$

*respectively* with $\text{SurMCSD}_{D^\psi}(\boldsymbol{f}', \boldsymbol{f}'')$ over a distribution $D^\psi$ as

$$(L_1/\text{MCD [16]}) : \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{K}\|\phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))) - \phi(\boldsymbol{f}''(\psi(\boldsymbol{x})))\|_1, \quad (23)$$

$$(\text{KL}) : \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[\text{KL}(\phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}''(\psi(\boldsymbol{x})))) + \text{KL}(\phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))))], \quad (24)$$

$$(\text{CE}) : \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[\text{CE}(\phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}''(\psi(\boldsymbol{x})))) + \text{CE}(\phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))))], \quad (25)$$

where $\phi(\cdot)$ is the softmax operator, $\text{KL}(\cdot,\cdot)$ is the Kullback-Leibler divergence, and $\text{CE}(\cdot,\cdot)$ is the cross-entropy function, and due to the same issue from ramp loss, we have used a standard log loss

$$\mathcal{L}_{\widehat{P}^\psi}(\boldsymbol{f}) = \mathbb{E}_{(\boldsymbol{x},y)\sim\widehat{P}} - \log[\phi_y(\boldsymbol{f}(\psi(x)))], \quad (26)$$

to replace the term $\mathcal{E}^{(\rho)}_{\widehat{P}^\psi}(\boldsymbol{f})$ of empirical source error in (21). While MCSD (6) takes a matrix-formed difference, the optimization-friendly surrogates (23), (24), and (25) generally take vector forms that characterize scoring disagreements between $K$ entry pairs of $\boldsymbol{f}'$ and $\boldsymbol{f}''$. In fact, we have the following proposition to show the equivalance of the matrix-formed MCSD to an aggregation of $K$ disagreements between any entry pair of $\boldsymbol{f}'$ and $\boldsymbol{f}''$.

**Proposition 3.** *Given the ramp loss $\Phi_\rho$ defined as (5), there exists a distance measure $\varphi : \mathbb{R}\times\mathbb{R} \to \mathbb{R}_+$ defined as*

$$\varphi(a, b) = (K-1)|\Phi_\rho(-a) - \Phi_\rho(-b)| + |\Phi_\rho(a) - \Phi_\rho(b)|,$$

*such that the matrix-formed $\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}''(\boldsymbol{x}))\|_1$ in MCSD (6) can be calculated as the sum of $\varphi$-distance values of $K$ entry pairs between $f'_k(\boldsymbol{x})$ and $f''_k(\boldsymbol{x})$, i.e.,*

$$\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}''(\boldsymbol{x}))\|_1 = \sum_{k=1}^{K}\varphi(f'_k(\boldsymbol{x}), f''_k(\boldsymbol{x})).$$

We also consider an algorithm that replaces the MCSD terms of (21) with surrogate function of the scalar-valued, degenerate MCSD version of (12), giving rise to

$$\min_{\boldsymbol{f},\psi} \mathcal{L}_{\widehat{P}^\psi}(\boldsymbol{f}) + [\widetilde{\text{SurMCSD}}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') - \widetilde{\text{SurMCSD}}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'')],$$
$$\max_{\boldsymbol{f}',\boldsymbol{f}''} [\widetilde{\text{SurMCSD}}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') - \widetilde{\text{SurMCSD}}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'')], \quad (27)$$

with $\widetilde{\text{SurMCSD}}_{D^\psi}(\boldsymbol{f}', \boldsymbol{f}'')$ over a distribution $D^\psi$ as [2]

$$(\text{MDD [18] variant}) : \mathbb{E}_{\boldsymbol{x}\sim D} - \log[\phi_{h_{\boldsymbol{f}'}(\psi(\boldsymbol{x}))}(\boldsymbol{f}''(\psi(\boldsymbol{x})))]. \quad (29)$$

2. For better optimization, we follow [18], [44] and practically implement the surrogate disagreement terms in (27) as

$$\widetilde{\text{SurMCSD}}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') = \mathbb{E}_{\boldsymbol{x}\sim\widehat{Q}}\log[1 - \phi_{h_{\boldsymbol{f}'}(\psi(\boldsymbol{x}))}(\boldsymbol{f}''(\psi(\boldsymbol{x})))],$$
$$\widetilde{\text{SurMCSD}}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') = \mathbb{E}_{\boldsymbol{x}\sim\widehat{P}} - \log[\phi_{h_{\boldsymbol{f}'}(\psi(\boldsymbol{x}))}(\boldsymbol{f}''(\psi(\boldsymbol{x})))]. \quad (28)$$

Similarly, an algorithm based on the degenerate (13) can be considered as an equivalence of the DANN algorithm [13] with $\widehat{\text{SurMCSD}}_{D^\psi}(\boldsymbol{f}', \boldsymbol{f}'')$ over a distribution $D^\psi$ as [3]

$$(\text{DANN [13]}) : \mathbb{E}_{\boldsymbol{x}\sim D} - \log[\text{sigmoid}(d(\psi(\boldsymbol{x})))], \quad (31)$$

where $d : D^\psi \to \mathbb{R}$ is a mapping function, and $\mathbb{1}[d(\psi(\boldsymbol{x})) > 0] = \mathbb{1}[h_{\boldsymbol{f}'}(\psi(\boldsymbol{x})) = h_{\boldsymbol{f}''}(\psi(\boldsymbol{x}))]$.

We note that algorithms discussed above resemble some recently proposed ones in the literature of UDA. For example, the objective (22) with the surrogate (23) is equivalent to the MCD algorithm [16]; the objective (27) with the degenerate surrogate (29) can be considered as a variant of MDD [18]. In Section 5, we conduct ablation studies to investigate the efficacy of these algorithms, and compare with a new one to be presented shortly, which is motivated from the same theoretically derived objective (21).

### 3.2 A New Algorithm of Domain-Symmetric Networks

Apart from the task classifier $\boldsymbol{f}$, algorithms of McDalNets presented above use two auxiliary classifiers $\boldsymbol{f}'$ and $\boldsymbol{f}''$ only for learning $\psi$, which is less efficient in use of parameters. To improve the efficiency, we propose an integrated scheme that concatenates $\boldsymbol{f}'$ and $\boldsymbol{f}''$ as $[\boldsymbol{f}'; \boldsymbol{f}''] \in \mathbb{R}^{2K}$, and lets them be respectively responsible for classification of the source and target instances, as shown in Figure 3. We correspondingly use the notations of $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ to replace $\boldsymbol{f}'$ and $\boldsymbol{f}''$, and denote the concatenated classifier as $\boldsymbol{f}^{st}$, which *shares parameters with* $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$. We term such a network as Domain-Symmetric Networks (SymmNets) due to the symmetry of class-wise neuron distributions in $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$.

To achieve the theoretically motivated learning objective (21), we have the following two designs to train SymmNets.

- Since target data $\{\boldsymbol{x}_j^t\}_{j=1}^{n_t}$ are unlabeled, to enforce symmetric predictions between the respective $K$ neurons of $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$, we use a *cross-domain training scheme* that trains the target classifier $\boldsymbol{f}^t$ using labeled source data $\{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{n_s}$.
- While different algorithms presented in Section 3.1 take adversarial training strategy (e.g., a manner of reverse gradients [13]) to learn domain-invariant conditional feature distributions, for SymmNets, we instead use a *domain confusion (and discrimination) training scheme* on the concatenated classifier $\boldsymbol{f}^{st}$ to achieve the same goal.

We introduce the following notations before presenting the algorithm of SymmNets. For an input $\boldsymbol{x}$, $\boldsymbol{f}^s(\psi(\boldsymbol{x})) \in \mathbb{R}^K$ and $\boldsymbol{f}^t(\psi(\boldsymbol{x})) \in \mathbb{R}^K$ are the output vectors before the softmax operator $\phi$, and we denote $\boldsymbol{p}^s(\boldsymbol{x}) = \phi(\boldsymbol{f}^s(\psi(\boldsymbol{x}))) \in [0,1]^K$ and $\boldsymbol{p}^t(\boldsymbol{x}) = \phi(\boldsymbol{f}^t(\psi(\boldsymbol{x}))) \in [0,1]^K$. We also apply softmax to output of the concatenated classifier $f^{st}$, resulting in $\boldsymbol{p}^{st}(\boldsymbol{x}) = \phi([\boldsymbol{f}^s(\psi(\boldsymbol{x})); \boldsymbol{f}^t(\psi(\boldsymbol{x}))]) \in [0,1]^{2K}$. For ease of subsequent notations, we also write $p_k^s(\boldsymbol{x})$ (*resp.* $p_k^t(\boldsymbol{x})$ or $p_k^{st}(\boldsymbol{x})$), $k \in \{1,\ldots,K\}$, for the $k^{th}$ element of probability vector $\boldsymbol{p}^s(\boldsymbol{x})$ (*resp.* $\boldsymbol{p}^t(\boldsymbol{x})$ or $\boldsymbol{p}^{st}(\boldsymbol{x})$) predicted by $\boldsymbol{f}^s$ (*resp.* $\boldsymbol{f}^t$ or $\boldsymbol{f}^{st}$).

3. For better optimization, we follow [13] and practically implement the surrogate disagreement terms as

$$\widehat{\text{SurMCSD}}_{\widehat{Q}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') = \mathbb{E}_{\boldsymbol{x}\sim\widehat{Q}}\log[1 - \text{sigmoid}(d(\psi(\boldsymbol{x})))],$$
$$\widehat{\text{SurMCSD}}_{\widehat{P}_x^\psi}(\boldsymbol{f}', \boldsymbol{f}'') = \mathbb{E}_{\boldsymbol{x}\sim\widehat{P}} - \log[\text{sigmoid}(d(\psi(\boldsymbol{x})))]. \quad (30)$$

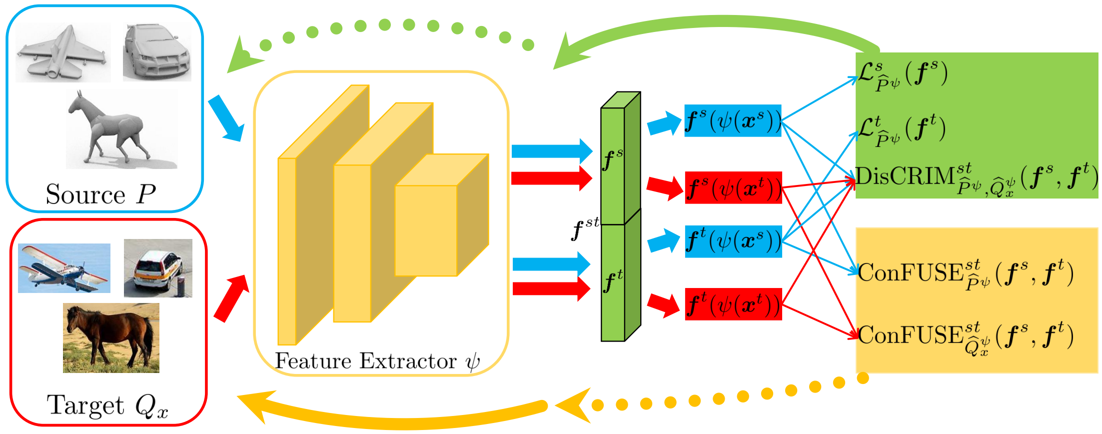



Fig. 3. The architecture of our proposed SymmNets, which includes a feature extractor $\psi$ and three classifiers of $f^s$, $f^t$, and $f^{st}$. Note that the classifier $f^{st}$ shares its layer neurons with those of $f^s$ and $f^t$. Parameters of the classifiers (i.e., $f^s$, $f^t$, and $f^{st}$) and those of feature extractor $\psi$ are respectively updated by gradients from loss terms in green and yellow boxes. Please refer to the main text for how the objectives are defined.

**Learning of the Source and Target Task Classifiers** We train the task classifier $f^s$ using a standard log loss over the labeled source data as follows

$$\min_{f^s} \mathcal{L}^s_{\widehat{P}\psi}(f^s) = -\frac{1}{n_s}\sum_{i=1}^{n_s} \omega_{y_i^s} \log(p^s_{y_i^s}(x_i^s)), \tag{32}$$

where $\omega_{y_i^s} \in [0, 1]$ is fixed as the value of 1 for closed set and open set UDA, and will be turned active in Section 4 for the extension of SymmNets to the setting of partial UDA.

To account for element-wise disagreements between predictions of $f^s$ and $f^t$, it is necessary to establish neuron-wise correspondence between them. To this end, we propose a cross-domain training scheme that trains the target classifier $f^t$ again using the labeled source data

$$\min_{f^t} \mathcal{L}^t_{\widehat{P}\psi}(f^t) = -\frac{1}{n_s}\sum_{i=1}^{n_s} \omega_{y_i^s} \log(p^t_{y_i^s}(x_i^s)). \tag{33}$$

At a first glance, it seems that training $f^t$ on $\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ only makes it a duplicate classifier of $f^s$. However, its effect on establishing neuron correspondence between $f^s$ and $f^t$ is very essential to achieve learning of domain-invariant features via the objectives of domain confusion and discrimination, as presented shortly. We also present ablation studies in Section 5 that verify the efficacy of the scheme (33).

**Adversarial Feature Learning via Domain Confusion and Discrimination** Algorithms in Section 3.1 use surrogate MCSD functions and minimize the induced MCSD divergence to learn $\psi$, in order to align conditional feature distributions across the source and target domains. Instead of using surrogate MCSD functions in SymmNets, we propose domain confusion objectives to directly reduce domain divergence, by learning $\psi$ such that it produces features whose scoring disagreements between $f^s$ and $f^t$ (*via their parameter-sharing* $f^{st}$) on both the source and target domains are *equally small (and ideally null)*. Our confusion objectives are as follows

$$\begin{aligned}\min_{\psi} \text{ConFUSE}^{st}_{\widehat{P}\psi}(f^s, f^t) = &-\frac{1}{2n_s}\sum_{i=1}^{n_s} \omega_{y_i^s} \log(p^{st}_{y_i^s}(x_i^s)) \\ &-\frac{1}{2n_s}\sum_{i=1}^{n_s} \omega_{y_i^s} \log(p^{st}_{y_i^s+K}(x_i^s)),\end{aligned} \tag{34}$$

$$\begin{aligned}\min_{\psi} \text{ConFUSE}^{st}_{\widehat{Q}^\psi_x}(f^s, f^t) = &-\frac{1}{2n_t}\sum_{j=1}^{n_t}\sum_{k=1}^{K} p^{st}_{k+K}(x_j^t) \log(p^{st}_k(x_j^t)) \\ &-\frac{1}{2n_t}\sum_{j=1}^{n_t}\sum_{k=1}^{K} p^{st}_{k}(x_j^t) \log(p^{st}_{k+K}(x_j^t)),\end{aligned} \tag{35}$$

where for a source example $(x^s, y^s)$ with the label $y^s = k$, we identify its corresponding pair of the $k^{th}$ and $(k+K)^{th}$ neurons in $f^{st}$, and use a cross-entropy between (two-way) uniform distribution and probabilities on this neuron pair; for a target example $x^t$, we simply use a cross-entropy between probabilities respectively on the first and second half sets of neurons in $f^{st}$. We again fix $\omega_{y_i^s} = 1$ for closed set UDA.

To provide an adversarial objective to the confusion ones (34) and (35), we use the following domain discrimination loss

$$\begin{aligned}\min_{f^s, f^t} \text{DisCRIM}^{st}_{\widehat{P}\psi,\widehat{Q}^\psi_x}(f^s, f^t) = &-\frac{1}{n_s}\sum_{i=1}^{n_s} \omega_{y_i^s} \log(p^{st}_{y_i^s}(x_i^s)) \\ &-\frac{1}{n_t}\sum_{j=1}^{n_t} \log(\sum_{k=1}^{K} p^{st}_{k+K}(x_j^t)),\end{aligned} \tag{36}$$

where $\omega_{y_i^s} = 1$ for closed set UDA, and $p^{st}_k(x)$ and $p^{st}_{k+K}(x)$ can be viewed as the probabilities of classifying an example $x$ of class $k$ as the source and target domains respectively.

**Overall Learning Objective** Combining (34) and (35), and (32), (33), and (36) gives the following objective to train SymmNets

$$\begin{aligned}&\min_{\psi} \text{ConFUSE}^{st}_{\widehat{P}\psi}(f^s, f^t) + \lambda \text{ConFUSE}^{st}_{\widehat{Q}^\psi_x}(f^s, f^t), \\ &\min_{f^s, f^t} \mathcal{L}^s_{\widehat{P}\psi}(f^s) + \mathcal{L}^t_{\widehat{P}\psi}(f^t) + \text{DisCRIM}^{st}_{\widehat{P}\psi,\widehat{Q}^\psi_x}(f^s, f^t),\end{aligned} \tag{37}$$

where $\lambda \in [0,1]$ is a trade-off parameter to suppress less stable signals of $\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ at early stages of training, since signals of $\text{ConFUSE}^{st}_{\widehat{P}_{\psi}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ from labeled source data are authentic and thus more stable. We note that the objective (37) of SymmNets is different from that in our preliminary work [17]: in (37), the scoring disagreements between $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ are minimized *explicitly* on target data, the entropy objective is achieved *implicitly* in the target confusion objective (35), and both the class and domain supervision of source data is adopted in the domain discrimination objective (36); in our preliminary work [17], the scoring disagreements between $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ are minimized *implicitly* on target data, the entropy objective is adopted *explicitly*, and only the domain supervision of source data is adopted in the domain discrimination objective. we use **SymmNets-V1** and **SymmNets-V2** to report the results respectively from these two versions of our algorithms.

**Theoretical Connection** We discuss the conditions on which the objective (37) of SymmNets connects with the theoretically derived objective (21). We first show with the following proposition that the objective (37) minimizes the term in (21) of empirical source error defined on both the $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$.

**Proposition 4.** *Let $\mathcal{F}$ be a rich enough space of continuous and bounded scoring functions, with the sum-to-zero constraint $\sum_{k=1}^{K} f_k = 0$. For $\boldsymbol{f}^s, \boldsymbol{f}^t \in \mathcal{F}$ and a fixed function $\psi$ that satisfies $\psi(\boldsymbol{x}_1) \neq \psi(\boldsymbol{x}_2)$ when $y_1 \neq y_2$, $\exists\ \rho > 0$ such that, minimizer $\boldsymbol{f}^{s*}$ of $\mathcal{L}^s_{\widehat{P}_{\psi}}(\boldsymbol{f}^s)$ in (37) also minimizes a term $\mathcal{E}^{(\rho)}_{\widehat{P}_{\psi}}(\boldsymbol{f}^s)$ in (21) of empirical source error defined on $\boldsymbol{f}^s$, and minimizer $\boldsymbol{f}^{t*}$ of $\mathcal{L}^t_{\widehat{P}_{\psi}}(\boldsymbol{f}^t)$ in (37) also minimizes a term $\mathcal{E}^{(\rho)}_{\widehat{P}_{\psi}}(\boldsymbol{f}^t)$ in (21) of empirical source error defined on $\boldsymbol{f}^t$.*

We note that the assumption of continuous and bounded scoring functions in Proposition 4 could be *practically* met with function implementation of fully-connected network layer; the assumption of $\psi(\boldsymbol{x}_1) \neq \psi(\boldsymbol{x}_2)$ when $y_1 \neq y_2$ is also reasonable with properly initialized and learned $\psi$. The objective (21) promotes alignment of conditional feature distributions across the two domains, by learning $\psi$ that reduces MCSD divergence. We show with the following proposition that the objective (37) has a same effect.

**Proposition 5.** *For $\psi$ of a function space of enough capacity and fixed functions $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ with the same range, minimizer $\psi^*$ of* $\text{ConFUSE}^{st}_{\widehat{P}_{\psi}}(\boldsymbol{f}^s,\boldsymbol{f}^t) + \lambda \text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ *with the parameter $\lambda > 0$ in (37) zeroizes* $\text{MCSD}^{(\rho)}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t) - \text{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ *in (21) of empirical* MCSD *divergence defined on $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$.*

We finally note that given fixed $\psi$, minimizing the domain discrimination term $\text{DisCRIM}^{st}_{\widehat{P}_{\psi},\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ in (37) over $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ (together with minimization of $\mathcal{L}^s_{\widehat{P}_{\psi}}(\boldsymbol{f}^s)$ and $\mathcal{L}^t_{\widehat{P}_{\psi}}(\boldsymbol{f}^t)$) will increase divergence between $\widehat{P}^{\psi}_{x}$ and $\widehat{Q}^{\psi}_{x}$, thus providing an adversarial feature learning signal similar to that provided by maximizing the MCSD divergence in (21). Specifically, $\text{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ is minimized by minimizing $\mathcal{L}^s_{\widehat{P}_{\psi}}(\boldsymbol{f}^s) + \mathcal{L}^t_{\widehat{P}_{\psi}}(\boldsymbol{f}^t)$ based on the Lemma A.2 in appendices (i.e., $\text{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t) \leq \mathcal{E}^{(\rho)}_{\widehat{P}_{\psi}}(\boldsymbol{f}^s) + \mathcal{E}^{(\rho)}_{\widehat{P}_{\psi}}(\boldsymbol{f}^t)$) and the Proposition 4. On the other hand, minimizing $\text{DisCRIM}^{st}_{\hat{P}_{\psi},\hat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ maximizes the output diversity of $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$, thus resulting in the maximization of $\text{MCSD}^{(\rho)}_{\hat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$.

# 4 Extensions for Partial and Open Set Domain Adaptation

The theories and algorithms discussed so far apply to the *closed set* setting of multi-class UDA, where a shared label space between the source and target domains is assumed. In this section, we show that simple extensions of our proposed algorithm of SymmNets can be used for either the *partial* [23], [25], [26], [27] or the *open set* [28], [29] multi-class UDA.

**Partial Domain Adaptation** The partial setting of multi-class UDA assumes that classes of the target domain constitutes an *unknown subset* of that of the source domain. As the setting suggests, a key challenge here is to identify the source instances that share the same classes with the target domain. To this end, we leverage the class-wise symmetry of neuron predictions between $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ in a SymmNet, and propose a soft class weighting scheme that simply weights source instances using collective prediction evidence of target instances from $\boldsymbol{f}^t$. Specifically, we compute the following class-wise averages of prediction probabilities for target instances, and use these averaged probabilities $\{\omega_{y^s_i}\}_{i=1}^{n_s}$ as weights for terms in the objectives (32), (33), (34), and (36) that involve labeled source data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n_s}$

$$\omega_k = \frac{1}{n_t}\sum_{j=1}^{n_t} p^t_k(\boldsymbol{x}^t_j),\ k \in \{1,\dots,K\}. \tag{38}$$

Such a scheme has the effect that source instances that are potentially of the classes exclusive to the target domain would be weighted down in the instance-reweighting version of the learning objective (37), thus promoting partial adaptation. In practice, we use more balanced class-wise weights in the early stages of training via

$$\omega_k \leftarrow \xi \frac{\omega_k}{\max_{k\in\mathcal{Y}} \omega_k} + (1-\xi),\ k \in \{1,\dots,K\}. \tag{39}$$

where $\xi$ is a parameter set to be smaller in the early stages of training. We note that similar soft weighting schemes are also used in [23], [25].

**Open Set Domain Adaptation** The open set setting of multi-class UDA takes a step further to assume that the target domain contains certain classes that are exclusive to the source domain as well. Let $K^s$ and $K^t$ respectively denote the numbers of classes in the source and target domains, and $\widetilde{K}$ be the number of classes common to them, which is assumed known in [28], [29]. We have $\widetilde{K} \leq K^s$ and $\widetilde{K} \leq K^t$. Extending a SymmNet for the open set setting can be simply achieved by adapting its $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ to respectively have $\widetilde{K}+1$ output neurons, where the final neuron of $\boldsymbol{f}^s$ is responsible for an aggregated prediction of the domain-specific $K^s - \widetilde{K}$ classes, and the same applies to the adapted $\boldsymbol{f}^t$. Although domain-specific classes in the source domain are treated as a single, super class, to achieve effective training of the adapted SymmNet via SGD, we still respect their overall population by sampling a $\nu \geq 1$ factor of more source examples from the super class than those from each of the $\widetilde{K}$ shared classes, when constituting a training source batch. We investigate different values of $\nu$ in Section 5; setting $\nu = 6$ consistently gives good results. Since target instances are unlabeled, we simply sample them randomly to constitute training target batches.

# 5 Experiments

In this section, we conduct experiments to investigate the practice of our introduced theory and algorithms. We compare different algorithms or implementations of McDalNets, its degenerate versions, and our proposed SymmNets-V1 [17] and SymmNets-V2 under the closed set setting of multi-class UDA. We also evaluate the efficacy of our SymmNets for partial and open set settings. These experiments are conducted on six benchmark datasets by implementing algorithms on three backbone networks, which are specified shortly. Additional experiments, results, and analysis are provided in the appendices.

**Datasets** We use the benchmark datasets summarized in Table 1 for our evaluation. In the closed set UDA, we follow standard protocols [36], [51] for the datasets of Office-31 [52], Office-Home [53], ImageCLEF-DA [54], and VisDA-2017 [55]: all labeled source and target samples are used for training; for the Digits datasets of [56], [57], [58], we follow the protocols in [19]. In partial UDA, all labeled source samples construct the source domain, and the target domain is constructed following the protocols of [23], [25]: for Office-31 [52], the samples of ten classes shared by Office-31 [52] and Caltech-256 [59] are selected as the target domain; for Office-Home [53], we choose (in alphabetic order) the first 25 classes as target classes and select all samples of these 25 classes as the target domain. In open set UDA, the samples of ten classes shared by Office-31 [52] and Caltech-256 [59] are selected as shared classes across domains. In alphabetical order, samples of Class 21∼Class 31 and Class 11∼Class 20 are used as unknown samples in the target and source domains, respectively; we follow the standard split for the benchmark dataset of Syn2Real [60].

TABLE 1
Summary of datasets. "C", "P", and "O" indicate the respective settings of closed set, partial, and open set domain adaptation.

| Dataset | Involved Tasks | No. of Domains | No. of Classes | No. of Samples |
|---|---|---|---|---|
| Office-31 [52] | C+P+O | 3 | 31 | 4,110 |
| Office-Home [53] | C+P | 4 | 65 | 15,500 |
| ImageCLEF-DA [54] | C | 3 | 12 | 1,800 |
| Digits [56], [57], [58] | C | 3 | 10 | 172.5K |
| VisDA-2017 [55], [60] | C | 2 | 12 | 280K |
| Syn2Real [60] | O | 2 | 13 | 248K |

**Implementations Details** All our methods are implemented using the Pytorch library. For the close set and partial settings of UDA, we adopt a ResNet pre-trained on ImageNet [61], after removing the last fully connected (FC) layer, as the feature extractor $\psi$. We fine-tune the feature extractor $\psi$ and train a classifier $\boldsymbol{f}^{st}$ from scratch through back propagation. Learning rate for the newly added layers is set as 10 times of that of the pre-trained layers. All parameters are updated by SGD with momentum of 0.9. We follow [51] to employ the annealing strategy of learning rate and the progressive strategy of $\lambda$: the learning rate is adjusted by $\eta_p = \frac{\eta_0}{(1+\alpha p)^\beta}$, where $p$ is the progress of training epochs linearly changing from 0 to 1, $\eta_0 = 0.01$, $\alpha = 10$, and $\beta = 0.75$, which are optimized to promote convergence and low errors on source samples; $\lambda$ is gradually changed from 0 to 1 by $\lambda_p = \frac{2}{1+\exp(-\gamma \cdot p)} - 1$, where $\gamma$ is set to 10 in all experiments. We empirically set $\xi = \lambda$ in all the experiments. Our classification results are obtained from the target task classifier $\boldsymbol{f}^t$ unless otherwise specified, and the comparison between the performance of the source and target task classifiers is illustrated in Figure 4. For the open set UDA, we follow [60] to replace the very top FC layer of an ImageNet pre-trained ResNet with three FC layers powered by batch normalization [62] and Leaky ReLU activation; the feature extractor $\psi$ is defined by pre-trained layers together with first two of the three added FC layers, and the last FC layer is the classifier $\boldsymbol{f}^{st}$. We freeze parameters of pre-trained layers and update those of the added FC layers with a learning rate of $0.001$, following [29]. We also follow [28], [29] to report OS as the accuracy averaged over all classes and OS$^*$ as that averaged over the domain shared classes only. We additionally implement our methods based on the AlexNet [2] and modified LeNet [14], [63] to testify its generalization to different architectures. Please refer to the appendices for more details. For fair comparison, results of other methods are either directly reported from their original papers if available or quoted from [15], [25] and [29], [60] for the closed set, partial and open set settings of UDA, respectively.

### 5.1 Analysis on Different Instantiations of McDalNets

In this section, we investigate different instantiations of McDalNets that are achieved by using surrogate functions (23), (24), or (25) to replace the MCSD terms in the general objective (21), by comparing with the counterparts based on surrogate functions (30) or (28) of degenerate MCSD (13) or (12). These experiments are conducted on the datasets of Office-31 [52], ImageCLEF-DA [54], Office-Home [53], Digits [56], [57], [58], and VisDA-2017 [55] under the setting of closed set UDA. In practice, we downweight the MCSD divergence in (21) with respect to the feature extractor $\psi$ at early stages of training, resulting in the following objective

$$\min_{\boldsymbol{f},\psi} \mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}) + \zeta[\mathrm{MCSD}^{(\rho)}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}', \boldsymbol{f}'') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}', \boldsymbol{f}'')],$$
$$\max_{\boldsymbol{f}',\boldsymbol{f}''} [\mathrm{MCSD}^{(\rho)}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}', \boldsymbol{f}'') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}', \boldsymbol{f}'')], \quad (40)$$

where we empirically set $\zeta = \lambda$ as described in the beginning of Section 5. The weight $\zeta$ is similarly applied to objectives based on surrogate MCSD functions. We adopt the gradient reversal layer to implement the adversarial objective. Therefore, the instantiation of McDalNets with surrogate function (30) of degenerate MCSD (13) coincides with that of DANN [13], [51]. The implementation details of other settings are the same as those described in the beginning of Section 5, except that we train three classifiers $\boldsymbol{f}$, $\boldsymbol{f}'$, and $\boldsymbol{f}''$ from scratch and the classification results are obtained from the task classifier $\boldsymbol{f}$. For ease of optimization, we also train auxiliary classifiers $\boldsymbol{f}'$ and $\boldsymbol{f}''$ using a standard log loss over labeled source data. The "Source Only" indeed gives a lower bound, where we fine-tune a model on the source data only.

Results in Table 2 show that all instantiations of McDalNets improve over the baseline of "Source Only", certifying the efficacy of MCSD divergence in domain discrepancy minimization. The McDalNets based on MCSD surrogates (23), (24), and (25) generally achieve better results than those based on surrogates (30) and (28) of the degenerate MCSD (13) and (12), testifying the advantage of characterizing finer details of the scoring disagreement in multi-class UDA. McDalNets based on the MCSD surrogate of CE (25) generally achieve better results than those based on $L_1$/MCD [16] (23) and KL (24), probably due to the mechanism that the CE based surrogate (25) also makes predictions of lower entropy; further explanation via illustration is given in the appendixes. Among all algorithms, SymmNets-V2 proposed in the present paper achieve the best results across all tasks, confirming its efficacy in multi-class UDA.

TABLE 2
Accuracies (%) of different instantiations of McDalNets on the datasets of Office-31 [52], ImageCLEF-DA [54], Office-Home [53], Digits [56], [57], [58] and VisDA-2017 [55] under the setting of closed set UDA. Each accuracy reported here is a *result averaged over individual tasks of a specific dataset*. All the results of individual tasks for the respective datasets are given in the appendices.

| Methods | Office-31 | ImageCLEF-DA | Office-Home | Digits | VisDA-2017 |
|---|---|---|---|---|---|
| Source Only | 81.8 | 82.7 | 58.9 | 70.5 | 41.8 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | | |
| DANN [13], [51] (30) | 82.8 | 84.2 | 60.0 | 72.5 | 58.4 |
| MDD [18] variant (28) | 84.5 | 86.7 | 61.1 | did not converge | did not converge |
| McDalNets based on the following surrogates of MCSD (6) | | | | | |
| $L_1$/MCD [16] (23) | 84.7 | 87.0 | 62.0 | 90.6 | 70.4 |
| KL (24) | 84.6 | 87.6 | 63.3 | 82.9 | 69.0 |
| CE (25) | 85.3 | 87.8 | 64.0 | 94.9 | 70.5 |
| SymmNets-V2 (37) | **89.1** | **89.7** | **68.1** | **96.0** | **71.3** |

TABLE 3
Ablation experiments on components of SymmNets-V2 using the datasets of Office-31 [52] and VisDA-2017 [55] under the setting of closed set UDA. All methods are based on models adapted from a 50-layer ResNet. Please refer to the main text for specifics of these methods.

| Methods | A → W | A → D | D → A | W → A | Synthetic → Real |
|---|---|---|---|---|---|
| SymmNets-V2 (w/o $\mathcal{L}^t_{\widehat{P}\psi}$) | 71.0±0.8 | 74.5±0.9 | 63.3±0.2 | 62.8±0.1 | 41.9 |
| SymmNets-V2 (w/o adversarial training) | 78.3±0.3 | 83.3±0.2 | 64.6±0.5 | 66.6±0.1 | 41.6 |
| SymmNets-V2 | **94.2**±0.1 | **93.5**±0.3 | **74.4**±0.1 | **73.4**±0.2 | **71.3** |

We also plot convergence curves for different instantiations of McDalNets in Figure 4, where we observe that those based on MCSD surrogates of $L_1$/MCD [16] (23), KL (24), and CE (25) converge generally smoother than those based on degenerate MCSD surrogates (28) and (30). It could be attributed to the in-built function property of MCSD (6), as illustrated in Figure 1. We particularly note that McDalNets based on degenerate MCSD surrogate (28) do not converge on the datasets of Digits and VisDA-2017. In comparison, SymmNets-V2 achieves the lowest classification error and the smoothest convergence.

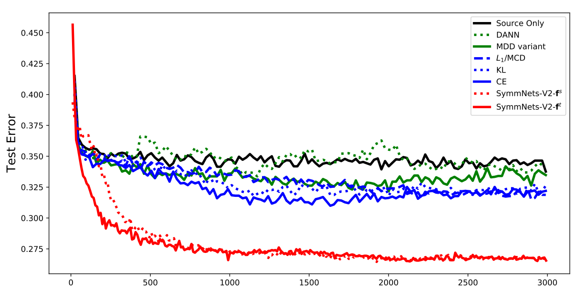



Fig. 4. Convergence plottings on the adaptation task **W** → **A** of the Office-31 [52] by Source Only, McDalNets based on the degenerate MCSD surrogates MDD [18] variant (28) and DANN [13], [51] (30), McDalNets based on the MCSD surrogates $L_1$/MCD [16] (23), KL (24), and CE (25), and SymmNets-V2. SymmNets-V2-$\boldsymbol{f}^s$ and SymmNets-V2-$\boldsymbol{f}^t$ represent the results obtained from the source classifier $\boldsymbol{f}^s$ and target classifier $\boldsymbol{f}^t$, respectively.

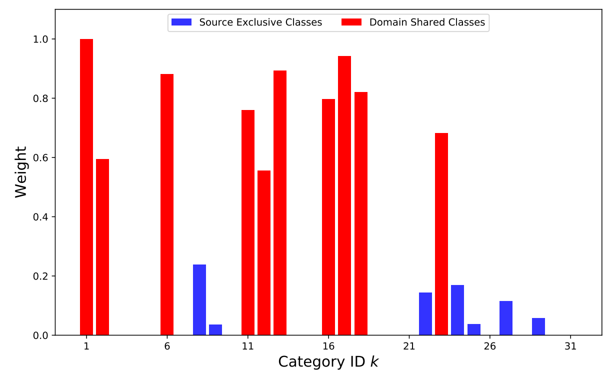



Fig. 5. Histograms of class weight $\omega_k$ learned by SymmNets-V2 (with active $\omega_k$) on the task of **A** → **W** under the setting of partial UDA. Model is adapted from a 50-layer ResNet.

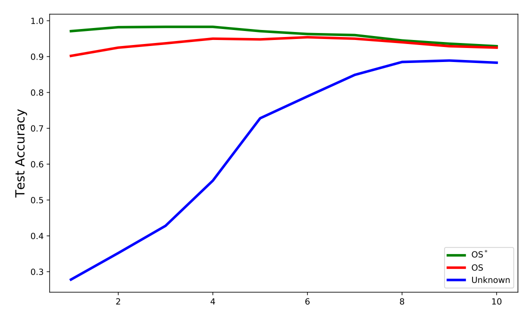
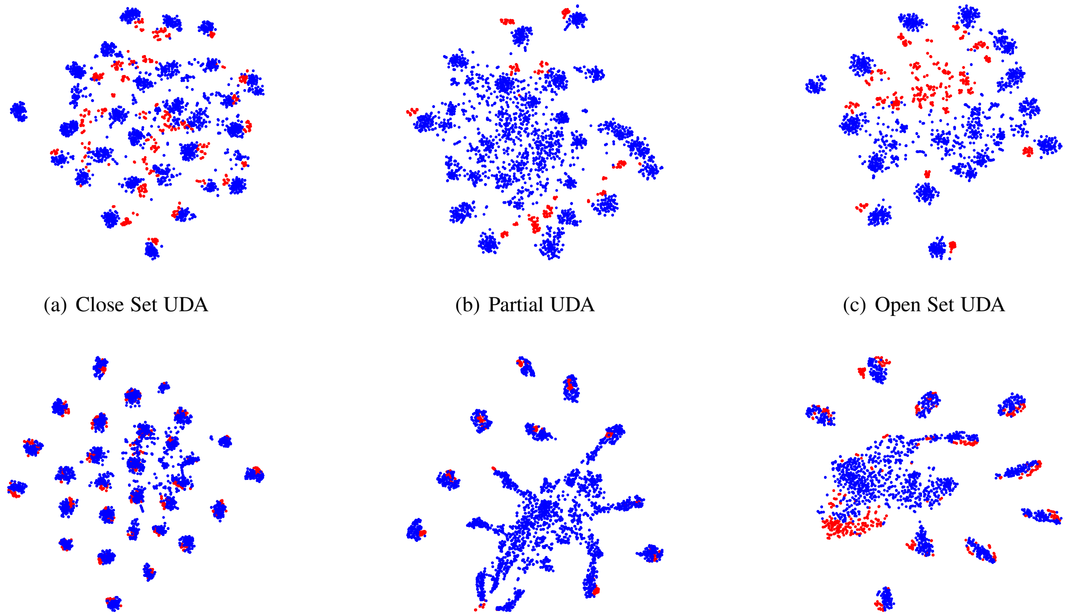



Fig. 6. Curve plottings for test accuracy of the unknown class (Unknown) and the mean accuracies over all classes (OS) and domain-shared classes (OS*), when setting different values of $\nu$ in open set UDA. The results are reported on the **A** → **W** task of Office-31 dataset [52] based on the SymmNets-V2 adapted from a 50-layer ResNet.

### 5.2 Ablation Studies of SymmNets

In this section, we investigate the effects of different components in our proposed SymmNets-V2 by conducting ablation experiments on the datasets of Office-31 [52] and VisDA-2017 [55] under the setting of closed set UDA, where networks are adapted from a 50-layer ResNet. To investigate how the cross-domain training term $\mathcal{L}^t_{\widehat{P}\psi}$ (33) contributes to a better adaptation in our overall adversarial learning objective (37), we remove it from (37) and denote the method as "SymmNets-V2 (w/o $\mathcal{L}^t_{\widehat{P}\psi}$)". To




(a) Close Set UDA (b) Partial UDA (c) Open Set UDA

(d) Close Set UDA (e) Partial UDA (f) Open Set UDA

Fig. 7. The t-SNE visualization of feature representations learned by DANN (top row) and SymmNets-V2 (bottom row) under the settings of closed set, partial, and open set UDA. Blue and red points are the respective samples from the source domain **A** and target domain **W**. For partial UDA, we illustrate the feature representations learned by SymmNets-V2 (With active $\omega_k$), where we focus on domain-shared classes, and leave the source classes exclusive to the target domain as an indistinguishable cluster via the soft class weighting scheme, as discussed in Section 4. A visualization with class label information is given in the appendixes.

evaluate the efficacy of our adversarial training, we remove the domain discrimination loss $\text{DisCRIM}^{st}_{\widehat{P}^{\psi},\widehat{Q}^{\psi}_{x}}$ (36) and the domain confusion loss of target data $\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}$ (35) from the overall objective (37), and use the following degenerate form to replace the domain confusion loss of source data $\text{ConFUSE}^{st}_{\widehat{P}^{\psi}}$ (34)

$$\min_{\psi} -\frac{1}{2n_s}\sum_{i=1}^{n_s}\omega_{y_i^s}\log(p^{st}_{y_i^s}(\boldsymbol{x}_i^s)) - \frac{1}{2n_s}\sum_{i=1}^{n_s}\omega_{y_i^s}\log(p^{st}_{y_i^s+K}(\boldsymbol{x}_i^s)); \quad (41)$$

we denote this method as "SymmNets-V2 (w/o adversarial training) ". Note that classification results for SymmNets (w/o $\mathcal{L}^{t}_{\widehat{P}^{\psi}}$) are obtained from the source task classifier $\boldsymbol{f}^s$ due to the inexistence of the direct supervision signals from target task classifier $\boldsymbol{f}^t$. Results in Table 3 show that SymmNets-V2 outperforms "SymmNets-V2 (w/o adversarial training)" by a large margin, verifying the efficacy of the discrepancy minimization via our proposed adversarial training. The performance slump of "SymmNets-V2 (w/o $\mathcal{L}^{t}_{\widehat{P}^{\psi}}$)" manifests the importance of cross-domain training term $\mathcal{L}^{t}_{\widehat{P}^{\psi}}$ (33) for learning a well-performed target task classifier via adversarial training.

**Soft Class Weighting Scheme in Partial UDA** To investigate the efficacy of soft class weighting scheme, we activate it with the strategy described in Section 4, giving rise to the method of "SymmNets-V2 (with active $\omega_k$)". Tables 8 and 9 show that results of SymmNets-V2 (with active $\omega_k$) improve over those of SymmNets-V2, empirically verifying its effectiveness. To have an intuitive understanding on what has happened, we illustrate in Figure 5 the learned weight of each source class on the adaptation task of **A** $\rightarrow$ **W** . SymmNets-V2 (with active $\omega_k$) assigns much larger weights to domain-shared classes than to the classes exclusive to the source domain, thus suppressing misalignment across the two domains.

**Effects of the Values of $\nu$ in Open Set UDA** We conduct experiments on the Office-31 dataset to investigate the effects of different values of $\nu$ for open set UDA. We plot in Figure 6 the accuracy of unknown class, and mean accuracies over domain-shared classes (OS*) and all classes (OS) with different values of $\nu$. As $\nu$ increases, the accuracy of the unknown class improves significantly whereas the mean accuracy of domain-shared classes drops slightly. We empirically set $\nu = 6$ in all experiments, which consistently gives good results.

**Feature Visualization** To have an intuitive understanding on what features comparative methods have learned, we visualize via t-SNE [64] in Figure 7 the network activations respectively from the feature extractors of DANN [13], [51] and SymmNets-V2 on the adaptation task of **A** $\rightarrow$ **W**. Compared with features learned by DANN, those by SymmNets-V2 are better aligned across the two domains for shared classes under all the settings of closed set, partial, and open set UDA, and they are well distinguished for domain-specific classes under the settings of partial and open set UDA; the visualization confirms the fineness of SymmNets-V2 in characterizing multi-class UDA.

### 5.3 Comparisons with the State of the Art

**Closed Set UDA** We report in Table 4, Table 5, Table 6, and Table 7 the classification results respectively on the popular closed set UDA datasets of Office-31 [52], ImageCLEF-DA [54], VisDA-2017 [55], and Office-Home [53]. Compared with existing adversarial learning based methods, including the seminal one of DANN [13] and the recent ones of MCD [16], CDAN [15], MDD [18], and SWD [20], our SymmNets-V2 achieves better

TABLE 4
Accuracy (%) on the Office-31 dataset [52] for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A → W | D → W | W → D | A → D | D → A | W → A | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [65] | 68.4±0.2 | 96.7±0.1 | 99.3±0.1 | 68.9±0.2 | 62.5±0.3 | 60.7±0.3 | 76.1 |
| DAN [36] | 80.5±0.4 | 97.1±0.2 | 99.6±0.1 | 78.6±0.2 | 63.6±0.3 | 62.8±0.2 | 80.4 |
| RTN [66] | 84.5±0.2 | 96.8±0.1 | 99.4±0.1 | 77.5±0.3 | 66.2±0.2 | 64.8±0.3 | 81.6 |
| DANN [13], [51] | 82.0±0.4 | 96.9±0.2 | 99.1±0.1 | 79.7±0.4 | 68.2±0.4 | 67.4±0.5 | 82.2 |
| ADDA [14] | 86.2±0.5 | 96.2±0.3 | 98.4±0.3 | 77.8±0.3 | 69.5±0.4 | 68.9±0.5 | 82.9 |
| JAN-A [37] | 86.0±0.4 | 96.7±0.3 | 99.7±0.1 | 85.1±0.4 | 69.2±0.3 | 70.7±0.5 | 84.6 |
| MADA [24] | 90.0±0.1 | 97.4±0.1 | 99.6±0.1 | 87.8±0.2 | 70.3±0.3 | 66.4±0.3 | 85.2 |
| SimNet [67] | 88.6±0.5 | 98.2±0.2 | 99.7±0.2 | 85.3±0.3 | 73.4±0.8 | 71.8±0.6 | 86.2 |
| MCD [16] | 89.6±0.2 | 98.5±0.1 | 100.0±.0 | 91.3±0.2 | 69.6±0.1 | 70.8±0.3 | 86.6 |
| CDAN+E [15] | 94.1±0.1 | 98.6±0.1 | **100.0**±.0 | 92.9±0.2 | 71.0±0.3 | 69.3±0.3 | 87.7 |
| MDD [18] | **94.5**±0.3 | 98.4±0.1 | **100.0**±.0 | 93.5±0.2 | **74.6**±0.3 | 72.2±0.1 | 88.9 |
| **SymmNets-V1** [17] | 90.8±0.1 | **98.8**±0.3 | **100.0**±.0 | **93.9**±0.5 | **74.6**±0.6 | 72.5±0.5 | 88.4 |
| **SymmNets-V2** | 94.2±0.1 | **98.8**±0.0 | **100.0**±.0 | 93.5±0.3 | 74.4±0.1 | **73.4**±0.2 | **89.1** |
| Kang *et al.* [68] | 86.8±0.2 | **99.3**±0.1 | **100.0**±.0 | 88.8±0.4 | 74.3±0.2 | 73.9±0.2 | 87.2 |
| TADA [69] | 94.3±0.3 | 98.7±0.1 | 99.8±0.2 | 91.6±0.3 | 72.9±0.2 | 73.0±0.3 | 88.4 |
| CADA-P [70] | **97.0**±0.2 | 99.3±0.1 | **100.0**±.0 | **95.6**±0.1 | 71.5±0.2 | 73.1±0.3 | 89.5 |
| CAN [71] | 94.5±0.3 | 99.1±0.2 | 99.8±0.2 | 95.0±0.3 | **78.0**±0.3 | **77.0**±0.3 | 90.6 |
| **SymmNets-V2-SC** | 94.9±0.3 | 99.1±0.1 | **100.0**±.0 | **95.6**±0.3 | 77.6±0.4 | **77.0**±0.3 | **90.7** |

TABLE 5
Accuracy (%) on the ImageCLEF-DA dataset [54] for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | I → P | P → I | I → C | C → I | C → P | P → C | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [65] | 74.8±0.3 | 83.9±0.1 | 91.5±0.3 | 78.0±0.2 | 65.5±0.3 | 91.2±0.3 | 80.7 |
| DAN [36] | 74.5±0.4 | 82.2±0.2 | 92.8±0.2 | 86.3±0.4 | 69.2±0.4 | 89.8±0.4 | 82.5 |
| DANN [13], [51] | 75.0±0.6 | 86.0±0.3 | 96.2±0.4 | 87.0±0.5 | 74.3±0.5 | 91.5±0.6 | 85.0 |
| JAN [37] | 76.8±0.4 | 88.0±0.2 | 94.7±0.2 | 89.5±0.3 | 74.2±0.3 | 91.7±0.3 | 85.8 |
| MADA [24] | 75.0±0.3 | 87.9±0.2 | 96.0±0.3 | 88.8±0.3 | 75.2±0.2 | 92.2±0.3 | 85.8 |
| CDAN+E [15] | 77.7±0.3 | 90.7±0.2 | **97.7**±0.3 | 91.3±0.3 | 74.2±0.2 | 94.3±0.3 | 87.7 |
| **SymmNets-V1** [17] | **80.2**±0.3 | **93.6**±0.2 | 97.0±0.3 | **93.4**±0.3 | 78.7±0.3 | **96.4**±0.1 | **89.9** |
| **SymmNets-V2** | 79.0±0.3 | 93.5±0.2 | 96.9±0.2 | **93.4**±0.3 | **79.2**±0.3 | 96.2±0.1 | 89.7 |
| CADA-P [70] | 78.0 | 90.5 | 96.7 | 92.0 | 77.2 | 95.5 | 88.3 |
| **SymmNets-V2-SC** | **79.2**±0.2 | **96.2**±0.3 | **96.8**±0.1 | **93.8**±0.2 | **77.8**±0.4 | **96.2**±0. | **90.0** |

TABLE 6
Accuracy (%) on the VisDA-2017 dataset [55] for closed set UDA. All comparative methods are based on a 101-layer ResNet except the MDD and CDAN+E, which are based on a 50-layer ResNet

| Methods | Synthetic → Real |
|---|---|
| Source Only [65] | 52.4 |
| DANN [13] | 57.4 |
| CDAN+E [15] | 70.0 |
| MCD [16] | 71.9 |
| ADR [19] | 73.5 |
| MDD [18] | 74.6 |
| SWD [20] | 76.4 |
| **SymmNets-V1** [17] | 72.1 |
| **SymmNets-V2** | **76.8** |
| TPN [72] | 80.4 |
| CAN [71] | **87.2** |
| **SymmNets-V2-SC** | 86.0 |

performance on most of these benchmarks, demonstrating the efficacy and fineness of SymmNets-V2 in characterizing multi-class UDA. We note that there exist a few recent methods that focus on other strategies, such as the feature attention strategy [15], [69], prototypical network [72], prediction consistency w.r.t input perturbation [40], and intra- and inter-class discrepancies [71], all of which are orthogonal to the strategy of adversarial training studied in the present work. To compare with these methods more fairly, we consider a few strategies of these methods amenable to adversarial training, including the class-aware sampling [71] empowered by alternative optimization [71], use of pseudo labels of target data as in prototypical network [72], and the min-entropy consensus [40], resulting in a variant of our method termed as "SymmNets-V2 Strengthened for Closed Set UDA (SymmNets-V2-SC)". SymmNets-V2-SC boosts the performance of SymmNets-V2 on the closed set UDA, especially on the VisDA-2017 dataset [55], indicating a promising direction of combining multiple strategies for the setting of closed set UDA.

**Partial UDA** We report in Table 8 and Table 9 the classification results respectively on the popular partial UDA datasets of Office-31 [52] and Office-Home [53]. The seminal methods [13], [36] achieve worse results than the Source Only baseline; in contrast, our SymmNets-V2 improves over the Source Only baseline by a large margin, confirming the effectiveness of our method in characterizing the domain distance at a finer level. Our SymmNets-V2 (with active $\omega_k$) outperforms all state-of-the-art methods on the two benchmark datasets, again confirming the effectiveness of our method.

**Open Set UDA** We report in Table 10 and Table 11 the classification results respectively on the popular open set UDA datasets of Office-31 [52] and Syn2Real [60]. Our SymmNets-V2 ($\nu = 6$) outperforms all state-of-the-art methods on the two benchmarks, confirming the effectiveness of our method in aligning both the domain-shared classes and the unknown class across the source and target domains.

## 6 Conclusion

In this paper, we study the formalism of unsupervised multi-class domain adaptation. We contribute a new bound for multi-

TABLE 7
Accuracy (%) on the Office-Home dataset [53] for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Only [65] | 34.9 | 50.0 | 58.0 | 37.4 | 41.9 | 46.2 | 38.5 | 31.2 | 60.4 | 53.9 | 41.2 | 59.9 | 46.1 |
| DAN [36] | 43.6 | 57.0 | 67.9 | 45.8 | 56.5 | 60.4 | 44.0 | 43.6 | 67.7 | 63.1 | 51.5 | 74.3 | 56.3 |
| DANN [13], [51] | 45.6 | 59.3 | 70.1 | 47.0 | 58.5 | 60.9 | 46.1 | 43.7 | 68.5 | 63.2 | 51.8 | 76.8 | 57.6 |
| JAN [37] | 45.9 | 61.2 | 68.9 | 50.4 | 59.7 | 61.0 | 45.8 | 43.4 | 70.3 | 63.9 | 52.4 | 76.8 | 58.3 |
| CDAN+E [15] | 50.7 | 70.6 | 76.0 | 57.6 | 70.0 | 70.0 | 57.4 | 50.9 | 77.3 | 70.9 | 56.7 | 81.6 | 65.8 |
| MDD [18] | **54.9** | 73.7 | 77.8 | 60.0 | 71.4 | 71.8 | 61.2 | **53.6** | 78.1 | 72.5 | **60.2** | 82.3 | **68.1** |
| **SymmNets-V1 [17]** | 47.7 | 72.9 | 78.5 | 64.2 | 71.3 | **74.2** | 64.2 | 48.8 | 79.5 | **74.5** | 52.6 | 82.7 | 67.6 |
| **SymmNets-V2** | 48.1 | **74.3** | **78.7** | **64.6** | **71.8** | 74.1 | **64.4** | 50.0 | **80.2** | 74.3 | 53.1 | **83.2** | **68.1** |
| DWT-MEC [40] | 50.3 | 72.1 | 77.0 | 59.6 | 69.3 | 70.2 | 58.3 | 48.1 | 77.3 | 69.3 | 53.6 | 82.0 | 65.6 |
| TADA [69] | 53.1 | 72.3 | 77.2 | 59.1 | 71.2 | 72.1 | 59.7 | 53.1 | 78.4 | 72.4 | 60.0 | 82.9 | 67.6 |
| CADA-P [70] | **56.9** | 76.4 | **80.7** | 61.3 | **75.2** | 75.2 | 63.2 | **54.5** | 80.7 | **73.9** | **61.5** | **84.1** | **70.2** |
| **SymmNets-V2-SC** | 51.6 | **76.9** | 80.3 | **68.6** | 71.8 | **78.3** | **65.8** | 50.5 | **81.2** | 73.1 | 54.2 | 82.4 | 69.6 |

TABLE 8
Accuracy (%) on the Office-31 dataset [52] for partial UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A → W | D → W | W → D | A → D | D → A | W → A | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [65] | 54.52 | 94.57 | 94.27 | 65.61 | 73.17 | 71.71 | 75.64 |
| DAN [36] | 46.44 | 53.56 | 58.60 | 42.68 | 65.66 | 65.34 | 55.38 |
| DANN [13], [51] | 41.35 | 46.78 | 38.85 | 41.36 | 41.34 | 44.68 | 42.39 |
| ADDA [14] | 43.65 | 46.48 | 40.12 | 43.66 | 42.76 | 45.95 | 43.77 |
| RTN [66] | 75.25 | 97.12 | 98.32 | 66.88 | 85.59 | 85.70 | 84.81 |
| JAN [37] | 43.39 | 53.56 | 41.40 | 35.67 | 51.04 | 51.57 | 46.11 |
| PADA [25] | 86.54 | 99.32 | **100.00** | 82.17 | 92.69 | 95.41 | 92.69 |
| ETN [27] | 94.52 | **100.00** | **100.00** | 95.03 | **96.21** | 94.64 | 96.73 |
| **SymmNets-V2** | 83.10 | 92.91 | 94.27 | 77.71 | 74.42 | 73.49 | 82.61 |
| **SymmNets-V2** (With active $\omega_k$) | **99.83** | 98.64 | **100.00** | **97.85** | 93.25 | **96.00** | **97.60** |

TABLE 9
Accuracy (%) on the Office-Home dataset [53] for partial UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Only [65] | 38.57 | 60.78 | 75.21 | 39.94 | 48.12 | 52.90 | 49.68 | 30.91 | 70.79 | 65.38 | 41.79 | 70.42 | 53.71 |
| DAN [36] | 44.36 | 61.79 | 74.49 | 41.78 | 45.21 | 54.11 | 46.92 | 38.14 | 68.42 | 64.37 | 45.37 | 68.85 | 54.48 |
| DANN [13], [51] | 44.89 | 54.06 | 68.97 | 36.27 | 34.34 | 45.22 | 44.08 | 38.03 | 68.69 | 52.98 | 34.68 | 46.50 | 47.39 |
| PADA [25] | 51.95 | 67.00 | 78.74 | 52.16 | 53.78 | 59.03 | 52.61 | 43.22 | 78.79 | 73.73 | 56.60 | 77.09 | 62.06 |
| ETN [27] | **59.24** | 77.03 | 79.54 | 62.92 | 65.73 | 75.01 | 68.29 | 55.37 | 84.37 | 75.72 | **57.66** | **84.54** | 70.45 |
| **SymmNets-V2** | 53.12 | 67.87 | 73.57 | 62.43 | 56.73 | 64.08 | **56.26** | 59.61 | 69.36 | 66.64 | 52.30 | 69.56 | 62.63 |
| **SymmNets-V2** (With active $\omega_k$) | 55.46 | **78.71** | **84.59** | **70.98** | **67.39** | **77.91** | **76.22** | 54.45 | **88.46** | **77.23** | 57.07 | 83.75 | **72.69** |

TABLE 10
Accuracy (%) on the Office-31 dataset [52] for open set UDA. Results of all methods are based on models adapted from a 50-layer ResNet.

| Methods | A→D | | A→W | | D→A | | D→W | | W→A | | W→D | | AVG | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | OS | OS* | OS | OS* | OS | OS* | OS | OS* | OS | OS* | OS | OS* | OS | OS* |
| Source Only [65] | 85.2 | 85.5 | 82.5 | 82.7 | 71.6 | 71.5 | 94.1 | 94.3 | 75.5 | 75.2 | 96.6 | 97.0 | 84.2 | 84.4 |
| DANN [13] | 86.5 | 87.7 | 85.3 | 87.7 | 75.7 | 76.2 | 97.5 | **98.3** | 74.9 | 75.6 | **99.5** | **100.0** | 86.6 | 87.6 |
| ATI-$\lambda$ [28] | 84.3 | 86.6 | 87.4 | 88.9 | 78.0 | 79.6 | 93.6 | 95.3 | 80.4 | 81.4 | 96.5 | 98.7 | 86.7 | 88.4 |
| AODA [29] | 88.6 | 89.2 | 86.5 | 87.6 | 88.9 | 90.6 | 97.0 | 96.5 | 85.8 | 84.9 | 97.9 | 98.7 | 90.8 | 91.3 |
| STA [73] | 93.7 | 96.1 | 89.5 | 92.1 | 89.1 | **93.5** | 97.5 | 96.5 | 87.9 | 87.4 | **99.5** | 99.6 | 92.9 | 94.1 |
| **SymmNets-V2** ($\nu = 6$) | **96.3** | **97.5** | **95.7** | **96.1** | **91.6** | 91.7 | **97.8** | **98.3** | **92.3** | **92.9** | 99.2 | **100.0** | **95.5** | **96.1** |

class UDA based on a novel notion of Multi-Class Scoring Disagreement (MCSD); a corresponding data-dependent PAC bound is also developed based on the notion of Rademacher complexity. The proposed MCSD is able to fully characterize the relations between any pair of multi-class scoring hypotheses, which is finer compared with those in existing domain adaptation bounds. Our derived bounds naturally suggest a Multi-class Domain-adversarial learning Networks (McDalNets), which promote alignment of conditional feature distributions across the source and target domains. We show that different instantiations of McDalNets via surrogate learning objectives either coincide with or resemble a few of recently popular methods, thus (partially) underscoring their practical effectiveness. Based on our same theory of multi-class UDA, we also introduce a new algorithm of Domain-Symmetric Networks (SymmNets), which is featured by a novel adversarial strategy of domain confusion and discrimination. SymmNets afford simple extensions that work equally well under the problem settings of either closed set, partial, or open set UDA.

TABLE 11
Accuracy (%) on Syn2Real dataset [60] for open set UDA. Results of all methods are based on models adapted from a 152-layer ResNet.

| Methods | plane | bcycle | bus | car | horse | knife | mcycl | person | plant | sktbrd | train | trunk | unk | OS* | OS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Known-to-Unknown Ratio = 1:1** | | | | | | | | | | | | | | | |
| Source Only [65] | 36 | 27 | 21 | 49 | 66 | 0 | 69 | 1 | 42 | 8 | 59 | 0 | 81 | 31 | 35 |
| DANN [13] | 53 | 5 | 31 | 61 | 75 | 3 | 81 | 11 | 63 | 29 | 68 | 5 | 76 | 43 | 40 |
| AODA [29] | 85 | 71 | 65 | 53 | 83 | **10** | 79 | 36 | 73 | 56 | 79 | **32** | **87** | 60 | 62 |
| **SymmNets-V2** ($\nu = 6$) | **93** | **79** | **85** | **75** | **92** | 3 | **91** | **80** | **84** | **69** | **75** | 2 | 57 | **69** | **68** |
| **Known-to-Unknown Ratio = 1:10** | | | | | | | | | | | | | | | |
| Source Only [65] | 23 | 24 | 43 | 40 | 44 | 0 | 56 | 2 | 24 | 8 | 47 | 1 | **93** | 26 | 31 |
| AODA [29] | 80 | 63 | 59 | 63 | 83 | 12 | 89 | 5 | **61** | 14 | 79 | 0 | 69 | 51 | 52 |
| **SymmNets-V2** ($\nu = 6$) | **90** | **72** | **76** | **68** | **90** | **14** | **94** | **18** | 59 | **20** | **83** | **5** | 70 | **59** | **59** |

Careful empirical studies show that algorithms of McDalNets consistently improve over its degenerate versions. Experiments under the settings of closed set, partial, and open set UDA also confirm the effectiveness of our proposed SymmNets empirically. The contributed theory and algorithms connect better with the practice in multi-class UDA. We expect they could provide useful principles for algorithmic design in future research.

## Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (Grant No.: 61771201), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No.: 2017ZT07X183), and the Guangdong R&D key project of China (Grant No.: 2019B010155001).

## References

[1] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[4] K. Jia, S. Li, Y. Wen, T. Liu, and D. Tao, "Orthogonal deep neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[5] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," in *Advances in neural information processing systems*, 2016, pp. 3630–3638.

[6] Y. Zhang, K. Jia, and Z. Wang, "Part-aware fine-grained object categorization using weakly supervised part detection network," *IEEE Transactions on Multimedia*, 2019.

[7] K. Jia, J. Lin, M. Tan, and D. Tao, "Deep multi-view learning using neuron-wise correlation-maximizing regularizers," *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 5121–5134, 2019.

[8] S. J. Pan, Q. Yang *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[9] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Advances in neural information processing systems*, 2007, pp. 137–144.

[10] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[11] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," in *22nd Conference on Learning Theory, COLT 2009*, 2009.

[12] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 9, pp. 1853–1865, 2016.

[13] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2017.

[14] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 4.

[15] M. Long, Z. CAO, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems 31*, 2018, pp. 1640–1650.

[16] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3723–3732.

[17] Y. Zhang, H. Tang, K. Jia, and M. Tan, "Domain-symmetric networks for adversarial domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5031–5040.

[18] Y. Zhang, T. Liu, M. Long, and M. Jordan, "Bridging theory and algorithm for domain adaptation," in *International Conference on Machine Learning*, 2019, pp. 7404–7413.

[19] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," *arXiv preprint arXiv:1711.01575*, 2017.

[20] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 285–10 295.

[21] S. Cicek and S. Soatto, "Unsupervised domain adaptation via regularized conditional alignment," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[22] Ü. Dogan, T. Glasmachers, and C. Igel, "A unified view on multi-class support vector classification." *Journal of Machine Learning Research*, vol. 17, no. 45, pp. 1–32, 2016.

[23] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[24] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *AAAI Conference on Artificial Intelligence*, 2018.

[25] Z. Cao, L. Ma, M. Long, and J. Wang, "Partial adversarial domain adaptation," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[26] J. Zhang, Z. Ding, W. Li, and P. Ogunbona, "Importance weighted adversarial nets for partial domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8156–8164.

[27] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2985–2994.

[28] P. P. Busto, A. Iqbal, and J. Gall, "Open set domain adaptation for image and action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.

[29] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 153–168.

[30] C. Cortes and M. Mohri, "Domain adaptation and sample bias correction theory and algorithm for regression," *Theoretical Computer Science*, vol. 519, pp. 103–126, 2014.

[31] C. Cortes, M. Mohri, and A. Muñoz Medina, "Adaptation algorithm and theory based on generalized discrepancy," in *Proceedings of the 21th*

*ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 169–178.

[32] M. Mohri and A. M. Medina, "New analysis and algorithm for learning with drifting distributions," in *International Conference on Algorithmic Learning Theory*. Springer, 2012, pp. 124–138.

[33] C. Zhang, L. Zhang, and J. Ye, "Generalization bounds for domain adaptation," in *Advances in neural information processing systems*, 2012, pp. 3320–3328.

[34] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," in *Advances in Neural Information Processing Systems*, 2017, pp. 3730–3739.

[35] S. Kuroki, N. Charoenphakdee, H. Bao, J. Honda, I. Sato, and M. Sugiyama, "Unsupervised domain adaptation based on source-guided discrepancy," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 4122–4129.

[36] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, pp. 97–105. [Online]. Available: http://dl.acm.org/citation.cfm?id=3045118.3045130

[37] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 2208–2217.

[38] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[39] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond sharing weights for deep domain adaptation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 4, pp. 801–814, 2018.

[40] S. Roy, A. Siarohin, E. Sangineto, S. R. Bulo, N. Sebe, and E. Ricci, "Unsupervised domain adaptation using feature-whitening and consensus loss," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9471–9480.

[41] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 7.

[42] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[43] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4068–4076.

[44] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[45] C. Villani, *Optimal transport: old and new*. Springer Science & Business Media, 2008, vol. 338.

[46] V. N. Vapnik, *Statistical Learning Theory*. John Wiley and Sons, 1998.

[47] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 67–81, 2004.

[48] Y. Liu and M. Yuan, "Reinforced multicategory support vector machines," *Journal of Computational and Graphical Statistics*, vol. 20, no. 4, pp. 901–919, 2011.

[49] S. Szedmak, J. Shawe-Taylor *et al.*, "Learning via linear operators: Maximum margin regression," in *In Proceedings of 2001 IEEE International Conference on Data Mining*. Citeseer, 2005.

[50] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT press, 2012.

[51] Y. Ganin and V. S. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2015, pp. 1180–1189.

[52] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *European conference on computer vision*. Springer, 2010, pp. 213–226.

[53] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. CVPR*, 2017, pp. 5018–5027.

[54] "Imageclef-da dataset," http://imageclef.org/2014/adaptation/.

[55] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, "Visda: The visual domain adaptation challenge," *arXiv preprint arXiv:1710.06924*, 2017.

[56] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[57] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS workshop on deep learning and unsupervised feature learning*, vol. 2011, no. 2, 2011, p. 5.

[58] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[59] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2066–2073.

[60] X. Peng, B. Usman, K. Saito, N. Kaushik, J. Hoffman, and K. Saenko, "Syn2real: A new benchmark forsynthetic-to-real visual domain adaptation," *CoRR*, vol. abs/1806.09755, 2018.

[61] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[62] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[63] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[64] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[65] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[66] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 136–144.

[67] P. O. Pinheiro and A. Element, "Unsupervised domain adaptation with similarity learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8004–8013.

[68] G. Kang, L. Zheng, Y. Yan, and Y. Yang, "Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[69] X. Wang, L. Li, W. Ye, M. Long, and J. Wang, "Transferable attention for domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5345–5352.

[70] V. K. Kurmi, S. Kumar, and V. P. Namboodiri, "Attending to discriminative certainty for domain adaptation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[71] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4893–4902.

[72] Y. Pan, T. Yao, Y. Li, Y. Wang, C.-W. Ngo, and T. Mei, "Transferrable prototypical networks for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2239–2247.

[73] H. Liu, Z. Cao, M. Long, J. Wang, and Q. Yang, "Separate to adapt: Open set domain adaptation via progressive separation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2927–2936.

[74] X. Ma, T. Zhang, and C. Xu, "Gcan: Graph convolutional adversarial network for unsupervised domain adaptation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[75] J. Zhang, Z. Ding, W. Li, and P. Ogunbona, "Importance weighted adversarial nets for partial domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8156–8164.

**Yabin Zhang** received the B.E. degree in School of Electronic and Information Engineering from South China University of Technology, Guangzhou, China, in 2017, where he is currently pursuing the master's degree. His current research interests include computer vision and deep learning, especially the deep transfer learning.

**Bin Deng** received the B.S. degree in information and computing science from South China Agricultural University, Guangzhou, China, in 2015, and the M.S. degree in pattern recognition and intelligent system from Shenzhen University, Shenzhen, China, in 2018.

He is currently a Research Assistant with the School of Electronic and Information Engineering, South China University of Technology. His research interests include machine learning, pattern recognition, and hyperspectral image processing.

**Hui Tang** received the B.E. degree in the School of Electronic and Information Engineering, South China University of Technology, in 2018. She is currently pursuing the Ph.D. degree in the School of Electronic and Information Engineering, South China University of Technology. Her research interests are in computer vision and machine learning.

**Lei Zhang** (F18) received his B.Sc. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, and M.Sc. and Ph.D degrees in Control Theory and Engineering from Northwestern Polytechnical University, Xian, P.R.China, in 1998 and 2001, respectively. From 2001 to 2002, he was a research associate in the Department of Computing, The Hong Kong Polytechnic University. From January 2003 to January 2006 he worked as a Postdoctoral Fellow in the Department of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an Assistant Professor. Since July 2017, he has been a Chair Professor in the same department. His research interests include Computer Vision, Image and Video Analysis, Pattern Recognition, and Biometrics, etc. Prof. Zhang has published more than 200 papers in those areas. As of 2019, his publications have been cited more than 43,000 times in literature. Prof. Zhang is a Senior Associate Editor of IEEE Trans. on Image Processing, and is/was an Associate Editor of SIAM Journal of Imaging Sciences, IEEE Trans.on CSVT, and Image and Vision Computing, etc. He is a Clarivate Analytics Highly Cited Researcher from 2015 to 2018. More information can be found in his homepage http://www4.comp.polyu.edu.hk/~cslzhang/.

**Kui Jia** received the B.E. degree from Northwestern Polytechnic University, Xian, China, in 2001, the M.E. degree from the National University of Singapore, Singapore, in 2004, and the Ph.D. degree in computer science from the Queen Mary University of London, London, U.K., in 2007.

He was with the Shenzhen Institute of Advanced Technology of the Chinese Academy of Sciences, Shenzhen, China, Chinese University of Hong Kong, Hong Kong, the Institute of Advanced Studies, University of Illinois at Urbana-Champaign, Champaign, IL, USA, and the University of Macau, Macau, China. He is currently a Professor with the School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China. His recent research focuses on theoretical deep learning and its applications in vision and robotic problems, including deep learning of 3D data and deep transfer learning.

## Appendix A
## Proof of Theorem 1

We begin with the following lemmas to prove Theorem 1.

**Lemma A.1.** *Fix $\rho > 0$. For any scoring functions $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$, the following holds for any distribution $D$ over $\mathcal{X} \times \mathcal{Y}$,*

$$\mathcal{E}_D(h_{\boldsymbol{f}}) \leq \mathcal{E}_D^{(\rho)}(\boldsymbol{f}') + \mathrm{MCSD}_{D_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}') \tag{42}$$

*where*

$$\mathcal{E}_D(h_{\boldsymbol{f}}) := \mathbb{E}_{(\boldsymbol{x},y)\sim D} \mathbb{1}[h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y], \tag{43}$$

$$\mathcal{E}_D^{(\rho)}(\boldsymbol{f}') := \mathbb{E}_{(\boldsymbol{x},y)\sim D} \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}'(\boldsymbol{x}), y)). \tag{44}$$

*Proof.* To prove the above inequality, we only need to prove that for any $(\boldsymbol{x}, y) \sim D$ and $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$, the inequality

$$\begin{aligned}\mathbb{1}[h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y] \leq & L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) + \\ & \frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1\end{aligned} \tag{45}$$

holds, where $L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) = \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}'(\boldsymbol{x}), y))$.

We prove in the following the inequality (45) holds in three separate cases that concern with the relationship between $h_{\boldsymbol{f}}(\boldsymbol{x})$, $h_{\boldsymbol{f}'}(\boldsymbol{x})$, and the class label $y$. For convenience, we also denote $h_{\boldsymbol{f}}(\boldsymbol{x}) = y_h$ and $h_{\boldsymbol{f}'}(\boldsymbol{x}) = y_h'$.

*Case 1:* When $h_{\boldsymbol{f}}(\boldsymbol{x}) = y_h = y$, no matter whether $h_{\boldsymbol{f}'}(\boldsymbol{x}) = y_h' = y$ or not, we have $\mathbb{1}(h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y) = 0$, and the inequality (45) holds obviously.

*Case 2:* When $h_{\boldsymbol{f}}(\boldsymbol{x}) = y_h \neq y$ and $h_{\boldsymbol{f}'}(\boldsymbol{x}) = y_h' \neq y$, due to the sum-to-zero constraint of $\sum_{k\in\mathcal{Y}} f_k'(\boldsymbol{x})$, we have $f_{y_h'}'(\boldsymbol{x}) >= 0$, and therefore

$$\begin{aligned}& L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) \geq \\ & \Phi_\rho(\mu_{y_h'}(\boldsymbol{f}'(\boldsymbol{x}), y)) = \Phi_\rho(-f_{y_h'}'(\boldsymbol{x})) = 1 = \mathbb{1}[h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y].\end{aligned}$$

Then the inequality (45) holds.

*Case 3:* When $h_{\boldsymbol{f}}(\boldsymbol{x}) = y_h \neq y$ and $h_{\boldsymbol{f}'}(\boldsymbol{x}) = y_h' = y$, we first show that when there exists $k \neq y$ such that $f_k'(\boldsymbol{x}) \geq 0$, we have $L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) \geq 1$, resulting in (45) directly; meanwhile, we show that when $f_k'(\boldsymbol{x}) < 0$ for all $k \neq y$ and $f_y'(\boldsymbol{x}) \leq \rho$, we have

$$\begin{aligned}L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) &= \sum_{k\neq y}[1 + \frac{f_k'(\boldsymbol{x})}{\rho}] + 1 - \frac{f_y'(\boldsymbol{x})}{\rho} = \\ & K - 1 - \frac{f_y'(\boldsymbol{x})}{\rho} + 1 - \frac{f_y'(\boldsymbol{x})}{\rho} = K - 2\frac{f_y'(\boldsymbol{x})}{\rho} \geq 1,\end{aligned}$$

and the inequality (45) holds; we proceed to discuss under the conditions of $f_k'(\boldsymbol{x}) < 0$ for all $k \neq y$ and $f_y'(\boldsymbol{x}) > \rho$.

1) Consider that $f_{y_h}'(\boldsymbol{x}) \leq -\rho$, we discuss under the separate conditions of either $f_{y_h}(\boldsymbol{x}) \geq \rho$ or $0 \leq f_{y_h}(\boldsymbol{x}) < \rho$. When $f_{y_h}(\boldsymbol{x}) \geq \rho$, we have

$$\begin{aligned}&\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\ &\geq \frac{1}{K}[\sum_{k\neq y_h} |\Phi_\rho(-f_{y_h}(\boldsymbol{x})) - \Phi_\rho(-f_{y_h}'(\boldsymbol{x}))| \\ &+ |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))|] = \frac{1}{K}[K - 1 + 1] = 1;\end{aligned}$$

when $0 \leq f_{y_h}(\boldsymbol{x}) < \rho$, we further discuss under the separate conditions of either $f_y(\boldsymbol{x}) \leq 0$ or $f_y(\boldsymbol{x}) > 0$. When $f_y(\boldsymbol{x}) \leq 0$, we have

$$\begin{aligned}&\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\ &\geq \frac{1}{K}[\sum_{k\neq y_h} |\Phi_\rho(-f_{y_h}(\boldsymbol{x})) - \Phi_\rho(-f_{y_h}'(\boldsymbol{x}))| + \\ &|\Phi_\rho(f_y'(\boldsymbol{x})) - \Phi_\rho(f_y(\boldsymbol{x}))|] = \frac{1}{K}[K - 1 + 1] = 1;\end{aligned}$$

when $f_y(\boldsymbol{x}) > 0$, we have

$$\begin{aligned}&\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\ &\geq \frac{1}{K}[\sum_{k\neq y_h} |\Phi_\rho(-f_{y_h}(\boldsymbol{x})) - \Phi_\rho(-f_{y_h}'(\boldsymbol{x}))| \\ &\quad + |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))| + |\Phi_\rho(f_y'(\boldsymbol{x})) - \Phi_\rho(f_y(\boldsymbol{x}))|] \\ &= \frac{1}{K}[K - 1 + 1 + \frac{f_{y_h}(\boldsymbol{x})}{\rho} - \frac{f_y(\boldsymbol{x})}{\rho}] \geq 1.\end{aligned}$$

Therefore, the inequality (45) holds.

2) Consider that $-\rho < f_{y_h}'(\boldsymbol{x}) < 0$, we discuss when both the conditions of $0 \leq f_{y_h}(\boldsymbol{x}) < \rho$ and $f_{y_h}(\boldsymbol{x}) \geq f_y(\boldsymbol{x}) > 0$ are met, or either of them is not. When $f_{y_h}(\boldsymbol{x}) \geq \rho$ or $f_y(\boldsymbol{x}) \leq 0$, we have

$$\begin{aligned}&\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\ &\geq \frac{1}{K}[\sum_{k\neq y_h} |\Phi_\rho(-f_{y_h}(\boldsymbol{x})) - \Phi_\rho(-f_{y_h}'(\boldsymbol{x}))| \\ &\quad + |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))| + |\Phi_\rho(f_y(\boldsymbol{x})) - \Phi_\rho(f_y'(\boldsymbol{x}))|] \\ &= \frac{1}{K}[-(K-1)\frac{f_{y_h}'(\boldsymbol{x})}{\rho} + |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))| \\ &\quad + |\Phi_\rho(f_y(\boldsymbol{x})) - \Phi_\rho(f_y'(\boldsymbol{x}))|] \\ &\geq \frac{1}{K}[-(K-1)\frac{f_{y_h}'(\boldsymbol{x})}{\rho} + 1] \\ &\geq \frac{1}{K}[-K\frac{f_{y_h}'(\boldsymbol{x})}{\rho}] = -\frac{f_{y_h}'(\boldsymbol{x})}{\rho};\end{aligned}$$

when $0 \leq f_{y_h}(\boldsymbol{x}) < \rho$ and $f_y(\boldsymbol{x}) > 0$, we have

$$\begin{aligned}&\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\ &\geq \frac{1}{K}[\sum_{k\neq y_h} |\Phi_\rho(-f_{y_h}(\boldsymbol{x})) - \Phi_\rho(-f_{y_h}'(\boldsymbol{x}))| \\ &\quad + |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))| + |\Phi_\rho(f_y(\boldsymbol{x})) - \Phi_\rho(f_y'(\boldsymbol{x}))|] \\ &\geq \frac{1}{K}[-(K-1)\frac{f_{y_h}'(\boldsymbol{x})}{\rho} + |\Phi_\rho(f_{y_h}(\boldsymbol{x})) - \Phi_\rho(f_{y_h}'(\boldsymbol{x}))| \\ &\quad + |\Phi_\rho(f_y(\boldsymbol{x})) - \Phi_\rho(f_y'(\boldsymbol{x}))|] \\ &\geq \frac{1}{K}[-(K-1)\frac{f_{y_h}'(\boldsymbol{x})}{\rho} + 1 + \frac{f_{y_h}(\boldsymbol{x})}{\rho} - \frac{f_y(\boldsymbol{x})}{\rho}] \\ &\geq \frac{1}{K}[-K\frac{f_{y_h}'(\boldsymbol{x})}{\rho}] \geq -\frac{f_{y_h}'(\boldsymbol{x})}{\rho}.\end{aligned}$$

Therefore, $\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \geq -\frac{f_{y_h}'(\boldsymbol{x})}{\rho}$ holds. At the same time, we also have $L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) \geq 1 + \frac{f_{y_h}'(\boldsymbol{x})}{\rho}$, thus the inequality (45) holds.

The proof is finished. □

**Lemma A.2.** *Fix $\rho > 0$. For any scoring functions $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$, the following holds for any distribution $D$ over $\mathcal{X} \times \mathcal{Y}$,*

$$\mathrm{MCSD}_{D_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}') \leq \mathcal{E}_D^{(\rho)}(\boldsymbol{f}) + \mathcal{E}_D^{(\rho)}(\boldsymbol{f}') \tag{46}$$

*where*

$$\mathcal{E}_D^{(\rho)}(\boldsymbol{f}) := \mathbb{E}_{(\boldsymbol{x},y)\sim D} \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}), y)). \tag{47}$$

*Proof.* To prove above inequality, we only need to prove that for any $(\boldsymbol{x}, y) \sim D$ and $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$, the inequality

$$\frac{1}{K}\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \leq L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) \tag{48}$$

holds, where $L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) = \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}), y))$.

Before proving (48), we first show that the following inequality

$$|\Phi_\rho(f_{y'}(\boldsymbol{x})) - \Phi_\rho(f'_{y'}(\boldsymbol{x}))| + \sum_{k\neq y'} |\Phi_\rho(-f_k(\boldsymbol{x})) - \Phi_\rho(-f'_k(\boldsymbol{x}))| \leq \Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})) + \sum_{k\neq y}[\Phi_\rho(-f_k(\boldsymbol{x})) + \Phi_\rho(-f'_k(\boldsymbol{x}))] \tag{49}$$

holds for any $y' \in \mathcal{Y}$. If $y' = y$, the inequality (49) holds obviously. We then discuss in the following under the condition of $y' \neq y$. In this case, the left hand side of the inequality (49) is equal to

$$|\Phi_\rho(f_{y'}(\boldsymbol{x})) - \Phi_\rho(f'_{y'}(\boldsymbol{x}))| + |\Phi_\rho(-f_y(\boldsymbol{x})) - \Phi_\rho(-f'_y(\boldsymbol{x}))| + \sum_{k\neq y,y'} |\Phi_\rho(-f_k(\boldsymbol{x})) - \Phi_\rho(-f'_k(\boldsymbol{x}))|, \tag{50}$$

and the right hand side of the inequality (49) is equal to

$$\Phi_\rho(-f_{y'}(\boldsymbol{x})) + \Phi_\rho(-f'_{y'}(\boldsymbol{x})) + \Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})) + \sum_{k\neq y,y'} [\Phi_\rho(-f_k(\boldsymbol{x})) + \Phi_\rho(-f'_k(\boldsymbol{x}))]. \tag{51}$$

By observing equations (50) and (51), it is obvious that the inequality

$$\sum_{k\neq y,y'} |\Phi_\rho(-f_k(\boldsymbol{x})) - \Phi_\rho(-f'_k(\boldsymbol{x}))| \leq \sum_{k\neq y,y'} [\Phi_\rho(-f_k(\boldsymbol{x})) + \Phi_\rho(-f'_k(\boldsymbol{x}))] \tag{52}$$

holds. We then discuss when both the conditions of $f_y(\boldsymbol{x}) > 0$ and $f'_y(\boldsymbol{x}) > 0$ are met, or either of them is not. When $f_y(\boldsymbol{x}) \leq 0$ or $f'_y(\boldsymbol{x}) \leq 0$, we have

$$\Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})) \geq 1 \geq |\Phi_\rho(-f_y(\boldsymbol{x})) - \Phi_\rho(-f'_y(\boldsymbol{x}))|;$$

when $f_y(\boldsymbol{x}) > 0$ and $f'_y(\boldsymbol{x}) > 0$, we have

$$|\Phi_\rho(-f_y(\boldsymbol{x})) - \Phi_\rho(-f'_y(\boldsymbol{x}))| = |1 - 1| = 0 \leq \Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})).$$

Therefore, we have

$$|\Phi_\rho(-f_y(\boldsymbol{x})) - \Phi_\rho(-f'_y(\boldsymbol{x}))| \leq \Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})). \tag{53}$$

Similarly, we also have

$$|\Phi_\rho(f_{y'}(\boldsymbol{x})) - \Phi_\rho(f'_{y'}(\boldsymbol{x}))| \leq \Phi_\rho(-f_{y'}(\boldsymbol{x})) + \Phi_\rho(-f'_{y'}(\boldsymbol{x})). \tag{54}$$

By combining the inequalities (52), (53), and (54), we can get the result of inequality (49). Therefore the inequality (49) holds for any $y' \in \mathcal{Y}$.

We now turn to prove that the inequality (48) holds. Based on the inequality of (49), we therefore have

$$\begin{aligned}
&\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1 \\
&= \sum_{y'\in\mathcal{Y}} [|\Phi_\rho(f_{y'}(\boldsymbol{x})) - \Phi_\rho(f'_{y'}(\boldsymbol{x}))| \\
&\quad + \sum_{k\neq y'} |\Phi_\rho(-f_k(\boldsymbol{x})) - \Phi_\rho(-f'_k(\boldsymbol{x}))|] \\
&\leq K[\Phi_\rho(f_y(\boldsymbol{x})) + \Phi_\rho(f'_y(\boldsymbol{x})) \\
&\quad + \sum_{k\neq y}[\Phi_\rho(-f_k(\boldsymbol{x})) + \Phi_\rho(-f'_k(\boldsymbol{x}))]] \\
&= K[L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y)],
\end{aligned}$$

thus resulting in (48) directly. The proof is finished. □

**Theorem A.1 (Theorem 1).** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$, the following holds over the source and target distributions $P$ and $Q$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + d_{MCSD}^{(\rho)}(P_x, Q_x) + \lambda, \tag{55}$$

*where the constant $\lambda = \mathcal{E}_P^{(\rho)}(\boldsymbol{f}^*) + \mathcal{E}_Q^{(\rho)}(\boldsymbol{f}^*)$ with $\boldsymbol{f}^* = \arg\min_{\boldsymbol{f}\in\mathcal{F}} \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + \mathcal{E}_Q^{(\rho)}(\boldsymbol{f})$, and*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) := \mathbb{E}_{(\boldsymbol{x},y)\sim Q} \mathbb{1}[h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y], \tag{56}$$

$$\mathcal{E}_P^{(\rho)}(\boldsymbol{f}) := \mathbb{E}_{(\boldsymbol{x},y)\sim P} \sum_{k=1}^{K} \Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}), y)). \tag{57}$$

*Proof.* Based on the Lemma A.1 and Lemma A.2, we have

$$\begin{aligned}
&\mathcal{E}_Q(h_{\boldsymbol{f}}) \\
&\leq \mathcal{E}_Q^{(\rho)}(\boldsymbol{f}^*) + \mathrm{MCSD}_{Q_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}^*) \\
&\leq \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + \mathcal{E}_P^{(\rho)}(\boldsymbol{f}^*) - \mathrm{MCSD}_{P_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}^*) \\
&\quad + \mathcal{E}_Q^{(\rho)}(\boldsymbol{f}^*) + \mathrm{MCSD}_{Q_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}^*) \\
&= \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + \mathrm{MCSD}_{Q_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}^*) - \mathrm{MCSD}_{P_x}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}^*) + \lambda \\
&\leq \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + \lambda \\
&\quad + \sup_{\boldsymbol{f}',\boldsymbol{f}''\in\mathcal{F}} [\mathrm{MCSD}_{Q_x}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'') - \mathrm{MCSD}_{P_x}^{(\rho)}(\boldsymbol{f}', \boldsymbol{f}'')] \\
&= \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + d_{MCSD}^{(\rho)}(P_x, Q_x) + \lambda.
\end{aligned}$$

□

## Appendix B
## Degenerate Versions of MCSD

**Lemma B.1 (Proposition 1).** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$, the following holds over the source and target distributions $P$ and $Q$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}_P^{(\rho)}(\boldsymbol{f}) + d_{\widetilde{MCSD}}^{(\rho)}(P_x, Q_x) + \lambda, \tag{58}$$

*where $\mathcal{E}_Q(h_{\boldsymbol{f}})$, $\mathcal{E}_P^{(\rho)}(\boldsymbol{f})$, and $\lambda$ are defined as the same as in Theorem A.1.*

*Proof.* The proof follows the same argument as for the proof of Theorem A.1. The only difference is that the term $d^{(\rho)}_{MCSD}(P_x, Q_x)$ is replaced by $d^{(\rho)}_{\widetilde{MCSD}}(P_x, Q_x)$. Therefore, to prove the above point, we only need to prove that

$$\mathcal{E}_D(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_D(\boldsymbol{f}') + \widetilde{\text{MCSD}}^{(\rho)}_{D_x}(\boldsymbol{f}, \boldsymbol{f}') \tag{59}$$

and

$$\widetilde{\text{MCSD}}^{(\rho)}_{D_x}(\boldsymbol{f}, \boldsymbol{f}') \leq \mathcal{E}^{(\rho)}_D(\boldsymbol{f}) + \mathcal{E}^{(\rho)}_D(\boldsymbol{f}') \tag{60}$$

satisfy for any scoring functions $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$ with respect to any distribution $D$ over $\mathcal{X} \times \mathcal{Y}$. We now turn to prove (59) and (60) respectively in the following.

To prove (59), we only need to prove that for any $(\boldsymbol{x}, y) \sim D$, the inequality

$$\begin{aligned}&\mathbb{1}(h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y) \\ &\quad\leq L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) + \Phi_{\rho/2}[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))]\end{aligned} \tag{61}$$

holds, where $L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) = \sum_{k=1}^K \Phi_\rho(\mu_k(\boldsymbol{f}'(\boldsymbol{x}), y))$. If $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq h_{\boldsymbol{f}}(\boldsymbol{x})$ or $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq y$, the right-hand side of the above inequality will reach the value of 1, which is obviously an upper bound of the left-hand side. Otherwise $h_{\boldsymbol{f}'}(\boldsymbol{x}) = h_{\boldsymbol{f}}(\boldsymbol{x}) = y$, and

$$\mathbb{1}(h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y) = 0 \leq L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y).$$

Therefore, the inequality (61) holds and then (59) holds.

To prove (60), we only need to prove that for any $(\boldsymbol{x}, y) \sim D$, the inequality

$$\begin{aligned}&\Phi_{\rho/2}[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] \\ &\qquad\leq L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y)\end{aligned} \tag{62}$$

holds. If $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq y$ or $h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y$, the right-hand side of the above inequality will reach the value of 1, which is obviously an upper bound of the left-hand side. Otherwise $h_{\boldsymbol{f}'}(\boldsymbol{x}) = h_{\boldsymbol{f}}(\boldsymbol{x}) = y$, and

$$\begin{aligned}&\Phi_{\rho/2}[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] \\ &\quad\leq L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) \leq L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y).\end{aligned}$$

Therefore, the inequality (62) holds and then (60) holds. □

**Lemma B.2 (Proposition 2).** *Fix $\rho > 0$. For any scoring function $\boldsymbol{f} \in \mathcal{F}$, the following holds over the source and target distributions $P$ and $Q$,*

$$\mathcal{E}_Q(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + d^{(\rho)}_{\widehat{MCSD}}(P_x, Q_x) + \lambda, \tag{63}$$

*where $\mathcal{E}_Q(h_{\boldsymbol{f}})$, $\mathcal{E}^{(\rho)}_P(\boldsymbol{f})$, and $\lambda$ are defined as the same as in Theorem A.1.*

*Proof.* Following the similar proof of Lemma B.1, to show the above result, we only need to prove that

$$\mathcal{E}_D(h_{\boldsymbol{f}}) \leq \mathcal{E}^{(\rho)}_D(\boldsymbol{f}') + \widehat{\text{MCSD}}^{(\rho)}_{D_x}(\boldsymbol{f}, \boldsymbol{f}') \tag{64}$$

and

$$\widehat{\text{MCSD}}^{(\rho)}_{D_x}(\boldsymbol{f}, \boldsymbol{f}') \leq \mathcal{E}^{(\rho)}_D(\boldsymbol{f}) + \mathcal{E}^{(\rho)}_D(\boldsymbol{f}') \tag{65}$$

satisfy for any scoring functions $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$ with respect to any distribution $D$ over $\mathcal{X} \times \mathcal{Y}$. We now turn to prove (64) and (65) respectively in the following. To prove (64), we only need to prove that for any $(\boldsymbol{x}, y) \sim D$, the inequality

$$\begin{aligned}&\mathbb{1}(h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y) \\ &\leq L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) + \mathbb{1}[\Phi_\rho[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] = 1]\end{aligned}$$

holds. If $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq h_{\boldsymbol{f}}(\boldsymbol{x})$ or $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq y$, the right-hand side of the above inequality will reach the value of 1, which is obviously an upper bound of the left-hand side. Otherwise $h_{\boldsymbol{f}'}(\boldsymbol{x}) = h_{\boldsymbol{f}}(\boldsymbol{x}) = y$, and

$$\begin{aligned}&\mathbb{1}(h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y) = 0 \\ &\quad\leq L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y) + \mathbb{1}[\Phi_\rho[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] = 1].\end{aligned}$$

To prove (65), we only need to prove that for any $(\boldsymbol{x}, y) \sim D$, the inequality

$$\begin{aligned}&\mathbb{1}[\Phi_\rho[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] = 1] \\ &\qquad\qquad\leq L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y)\end{aligned}$$

holds. If $h_{\boldsymbol{f}'}(\boldsymbol{x}) \neq y$ or $h_{\boldsymbol{f}}(\boldsymbol{x}) \neq y$, the right-hand side of the above inequality will reach the value of 1, which is obviously an upper bound of the left-hand side. Otherwise $h_{\boldsymbol{f}'}(\boldsymbol{x}) = h_{\boldsymbol{f}}(\boldsymbol{x}) = y$, and

$$\begin{aligned}&\mathbb{1}[\Phi_\rho[\boldsymbol{\mu}_{h_{\boldsymbol{f}'(\boldsymbol{x})}}(\boldsymbol{f}'(\boldsymbol{x}), h_{\boldsymbol{f}}(\boldsymbol{x}))] = 1] = 0 \\ &\qquad\qquad\leq L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}), y) + L^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}), y).\end{aligned}$$

□

## Appendix C
## Proof of Theorem 2

We begin with the following lemmas to prove Theorem 2.

**Lemma C.1.** *(Two-sided Rademacher complexity bound, a modified version of Theorem 3.1, Mohri et al. [50]) Let $\mathcal{G}$ be a family of functions mapping from $\mathcal{Z}$ to $[0, 1]$. Let $D$ is any distribution over $\mathcal{Z}$, and a sample $\mathcal{S} = \{z_1, ..., z_m\}$ drawn i.i.d. from $D$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $g \in \mathcal{G}$:*

$$|\mathbb{E}[g(z)] - \frac{1}{m}\sum_{i=1}^m g(z_i)| \leq 2\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{G}) + 3\sqrt{\frac{\log\frac{4}{\delta}}{2m}}. \tag{66}$$

**Lemma C.2.** *(Talagrand's lemma, Lemma 4.2 of Mohri et al. [50]) Let $\Phi : \mathbb{R} \to \mathbb{R}$ be an l-Lipschitz. Then, for any hypothesis set $\mathcal{H}$ of real-valued functions, the following inequality holds:*

$$\widehat{\mathfrak{R}}_{\widehat{D}}(\Phi \circ \mathcal{H}) \leq l\widehat{\mathfrak{R}}_{\widehat{D}}(\mathcal{H}). \tag{67}$$

**Lemma C.3.** *Let $\mathcal{F}$ be the space of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^K$. Let $D$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ and let $\widehat{D}$ denote the corresponding empirical distribution for a sample $\mathcal{S} = \{(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_m, y_m)\}$ drawn i.i.d. from $D$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $\boldsymbol{f} \in \mathcal{F}$:*

$$|\mathcal{E}^{(\rho)}_D(\boldsymbol{f}) - \mathcal{E}^{(\rho)}_{\widehat{D}}(\boldsymbol{f})| \leq \frac{2K^2}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1\mathcal{F}) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2m}} \tag{68}$$

*where*

$$\begin{aligned}\mathcal{E}^{(\rho)}_D(\boldsymbol{f}) &:= \mathbb{E}_{(\boldsymbol{x},y)\sim D}\sum_{k=1}^K \Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}_i), y_i)) \\ &= \mathbb{E}_{(\boldsymbol{x},y)\sim D} L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y_i).\end{aligned} \tag{69}$$

*Proof.* Since the loss function $L^{(\rho)}$ is bounded by $K$, we scale the loss $L^{(\rho)}$ to $[0,1]$ by dividing by $K$, and denote the new class by $L^{(\rho)}/K$. By Lemma C.1 applied to $L^{(\rho)}/K$, for any $\delta > 0$, with probability at least $1-\delta$, the following inequality holds for all $\boldsymbol{f} \in \mathcal{F}$,

$$|\frac{\mathcal{E}_D^{(\rho)}(\boldsymbol{f})}{K} - \frac{\mathcal{E}_{\widehat{D}}^{(\rho)}(\boldsymbol{f})}{K}| \le 2\widehat{\mathfrak{R}}_{\mathcal{S}}(L^{(\rho)}/K) + 3\sqrt{\frac{\log\frac{4}{\delta}}{2m}}.$$

Based on the property of the Rademacher complexity, we have $\widehat{\mathfrak{R}}_{\mathcal{S}}(L^{(\rho)}/K) = \frac{1}{K}\widehat{\mathfrak{R}}_{\mathcal{S}}(L^{(\rho)})$, and based on Lemma C.2 and the sub-additivity of sup, we have

$$\begin{aligned}
&\widehat{\mathfrak{R}}_{\mathcal{S}}(L^{(\rho)}) \\
&= \frac{1}{m}\mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y_i)] \\
&= \frac{1}{m}\mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sum_{y\in\mathcal{Y}} \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)\mathbb{1}(y=y_i)] \\
&\le \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)\mathbb{1}(y=y_i)] \\
&= \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)(\frac{2\mathbb{1}(y=y_i)-1}{2} + \frac{1}{2})] \\
&\le \frac{1}{2m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i\epsilon_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)] + \\
&\qquad \frac{1}{2m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)] \\
&= \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i L^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i), y)] \\
&= \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i[\sum_{k\neq y}\Phi_\rho(-f_k(\boldsymbol{x}_i)) + \Phi_\rho(f_y(\boldsymbol{x}_i))]] \\
&\le \frac{1}{m}\sum_{y\in\mathcal{Y}}\sum_{k\neq y} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i\Phi_\rho(-f_k(\boldsymbol{x}_i))] + \\
&\qquad \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i\Phi_\rho(f_y(\boldsymbol{x}_i))] \\
&\le \frac{1}{m}\sum_{y\in\mathcal{Y}}\sum_{k\neq y} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i\Phi_\rho(-f(\boldsymbol{x}_i))] + \\
&\qquad \frac{1}{m}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i\Phi_\rho(f(\boldsymbol{x}_i))] \\
&\le \frac{1}{m\rho}\sum_{y\in\mathcal{Y}}\sum_{k\neq y} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i[-f(\boldsymbol{x}_i)]] + \\
&\qquad \frac{1}{m\rho}\sum_{y\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i f(\boldsymbol{x}_i)] \\
&= \frac{1}{m\rho}\sum_{y\in\mathcal{Y}}\sum_{k\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i f(\boldsymbol{x}_i)] \\
&= \frac{K^2}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F})),
\end{aligned}$$

where $\epsilon_i = 2\mathbb{1}(y=y_i) - 1 \in \{-1, 1\}$ and we use the fact that $\epsilon_i\sigma_i$ has the same distribution as $\sigma_i$. The proof is finished by combing the above inequalities. □

**Lemma C.4.** *Let $\mathcal{F}$ be the space of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^K$. Let $D$ be a distribution over $\mathcal{X}$ and let $\widehat{D}$ denote the corresponding empirical distribution for a sample $\mathcal{S} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_m\}$ drawn i.i.d. from $D$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1-\delta$, the following holds for all $\boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}$:*

$$\begin{aligned}
&|\mathrm{MCSD}_D^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}') - \mathrm{MCSD}_{\widehat{D}}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}')| \\
&\le \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F})) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2m}}
\end{aligned} \tag{70}$$

*Proof.* Denote hypothesis set $\mathcal{M} := \{\boldsymbol{x} \to \|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1/K^2 | \boldsymbol{f}, \boldsymbol{f}' \in \mathcal{F}\}$ as a new class. Then the class $\mathcal{M}$ is a family of functions mapping from $\mathcal{X}$ to $[0,1]$. By Lemma C.1 applied to $\mathcal{M}$, for any $\delta > 0$, with probability at least $1-\delta$ we have

$$|\frac{\mathrm{MCSD}_D^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}')}{K} - \frac{\mathrm{MCSD}_{\widehat{D}}^{(\rho)}(\boldsymbol{f}, \boldsymbol{f}')}{K}| \le 2\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{M}) + 3\sqrt{\frac{\log\frac{4}{\delta}}{2m}}$$

and based on Lemma C.2 and the sup-additivity of sup,

$$\begin{aligned}
&\widehat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{M}) \\
&= \frac{1}{K^2 m}\mathbb{E}_\sigma[\sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}} \sum_{i=1}^m \sigma_i\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}(\boldsymbol{x}_i)) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x}))\|_1] \\
&\le \frac{1}{K^2 m}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}} \sum_{i=1}^m \sigma_i|\Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}_i), k')) \\
&\qquad - \Phi_\rho(\mu_k(\boldsymbol{f}'(\boldsymbol{x}_i), k'))|] \\
&\le \frac{1}{K^2 m}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}} \sum_{i=1}^m \sigma_i[\Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}_i), k')) \\
&\qquad - \Phi_\rho(\mu_k(\boldsymbol{f}'(\boldsymbol{x}_i), k'))]] \\
&\le \frac{2}{K^2 m}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i\Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}_i), k'))] \\
&\le \frac{2}{K^2 m\rho}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i\mu_k(\boldsymbol{f}(\boldsymbol{x}_i), k')] \\
&= \frac{2}{K^2 m\rho}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{\boldsymbol{f}\in\mathcal{F}} \sum_{i=1}^m \sigma_i f_k(\boldsymbol{x}_i)] \\
&\le \frac{2}{K^2 m\rho}\sum_{k,k'\in\mathcal{Y}} \mathbb{E}_\sigma[\sup_{f\in\Pi_1(\mathcal{F})} \sum_{i=1}^m \sigma_i f(\boldsymbol{x}_i)] \\
&= \frac{2}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F})).
\end{aligned}$$

The proof is finished by combing the above inequalities. □

**Lemma C.5.** *Let $\mathcal{F}$ be the space of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^K$. Let $P_x$ and $Q_x$ be source and target marginal distributions over $\mathcal{X}$ and let $\widehat{P}_x$ and $\widehat{Q}_x$ denote the corresponding empirical distributions for a sample of $\mathcal{S} = \{\boldsymbol{x}_i^s\}_{i=1}^{n_s}$ and a sample of $\mathcal{T} = \{\boldsymbol{x}_i^t\}_{i=1}^{n_t}$ respectively. Fix $\rho > 0$. Then, for any*

*$\delta > 0$, with probability at least $1-2\delta$, the following holds:*

$$d^{(\rho)}_{MCSD}(P_x, Q_x) \le d^{(\rho)}_{MCSD}(\widehat{P}_x, \widehat{Q}_x) + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F})) + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{T}}(\Pi_1(\mathcal{F})) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_s}} + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_t}} \tag{71}$$

*Proof.* Based on the Lemma C.4 and the sub-additivity of sup, by using the union bound, for any $\delta > 0$, with probability at least $1-2\delta$, we have

$$\begin{aligned}
&d^{(\rho)}_{MCSD}(P_x, Q_x)\\
&= \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{Q_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{P_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&= \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{Q_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}')\\
&\quad + \mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}')\\
&\quad + \mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{P_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\le \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{Q_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\quad + \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\quad + \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{P_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\le \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\quad + \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}|\mathrm{MCSD}^{(\rho)}_{Q_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}')|\\
&\quad + \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}|\mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{P_x}(\boldsymbol{f},\boldsymbol{f}')|\\
&\le \sup_{\boldsymbol{f},\boldsymbol{f}'\in\mathcal{F}}[\mathrm{MCSD}^{(\rho)}_{\widehat{Q}_x}(\boldsymbol{f},\boldsymbol{f}') - \mathrm{MCSD}^{(\rho)}_{\widehat{P}_x}(\boldsymbol{f},\boldsymbol{f}')]\\
&\quad + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{T}}(\Pi_1(\mathcal{F})) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_t}}\\
&\quad + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F})) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_s}}\\
&= d^{(\rho)}_{MCSD}(\widehat{P}_x, \widehat{Q}_x) + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1(\mathcal{F}))\\
&\quad + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{T}}(\Pi_1(\mathcal{F})) + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_s}} + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_t}}.
\end{aligned}$$

□

**Theorem C.1 (Theorem 2).** *Let $\mathcal{F}$ be the space of scoring functions mapping from $\mathcal{X}$ to $\mathbb{R}^K$. Let $P$ and $Q$ be the source and target distributions over $\mathcal{X}\times\mathcal{Y}$, and $P_x$ and $Q_x$ be the corresponding marginal distributions over $\mathcal{X}$. Let $\widehat{P}$ and $\widehat{Q}_x$ denote the corresponding empirical distributions for a sample $\mathcal{S} = \{(\boldsymbol{x}^s_i, y^s_i)\}_{i=1}^{n_s}$ and a sample $\mathcal{T} = \{\boldsymbol{x}^t_j\}_{j=1}^{n_t}$. Fix $\rho > 0$. Then, for any $\delta > 0$, with probability at least $1-3\delta$, the following holds for all $\boldsymbol{f}\in\mathcal{F}$*

$$\begin{aligned}
\mathcal{E}_Q(h_{\boldsymbol{f}}) \le& \mathcal{E}^{(\rho)}_{\widehat{P}}(\boldsymbol{f}) + d^{(\rho)}_{MCSD}(\widehat{P}_x, \widehat{Q}_x)\\
&+ (\frac{2K^2}{\rho} + \frac{4K}{\rho})\widehat{\mathfrak{R}}_{\mathcal{S}}(\Pi_1\mathcal{F}) + \frac{4K}{\rho}\widehat{\mathfrak{R}}_{\mathcal{T}}(\Pi_1(\mathcal{F}))\\
&+ 6K\sqrt{\frac{\log\frac{4}{\delta}}{2n_s}} + 3K\sqrt{\frac{\log\frac{4}{\delta}}{2n_t}} + \lambda,
\end{aligned} \tag{72}$$

*where the constant $\lambda = \min_{\boldsymbol{f}\in\mathcal{F}} \mathcal{E}^{(\rho)}_P(\boldsymbol{f}) + \mathcal{E}^{(\rho)}_Q(\boldsymbol{f})$, and*

$$\mathcal{E}^{(\rho)}_{\widehat{P}}(\boldsymbol{f}) := \frac{1}{n_s}\sum_{i=1}^{n_s}\sum_{k=1}^{K}\Phi_\rho(\mu_k(\boldsymbol{f}(\boldsymbol{x}^s_i), y^s_i)). \tag{73}$$

*Proof.* The bound is achieved by applying Lemma C.3, Lemma C.5, Lemma A.1, and the union bound. □

## Appendix D
## Connecting Theory with Algorithms

**Connections between KL (24) and CE (25)**

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[\mathrm{CE}(\phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))))\\
&\qquad + \mathrm{CE}(\phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))))]\\
&= \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[-\sum_{k=1}^{K}\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x})))\log(\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x}))))\\
&\qquad - \sum_{k=1}^{K}\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x})))\log(\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x}))))]\\
&= \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[-\sum_{k=1}^{K}\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x})))\log(\frac{\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x})))}{\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x})))})\\
&\qquad - \sum_{k=1}^{K}\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x})))\log(\frac{\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x})))}{\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x})))})]\\
&+ \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[-\sum_{k=1}^{K}\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x})))\log(\phi_k(\boldsymbol{f}'(\psi(\boldsymbol{x}))))\\
&\qquad - \sum_{k=1}^{K}\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x})))\log(\phi_k(\boldsymbol{f}''(\psi(\boldsymbol{x}))))]\\
&= \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[\mathrm{KL}(\phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))))\\
&\qquad + \mathrm{KL}(\phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))), \phi(\boldsymbol{f}'(\psi(\boldsymbol{x}))))] +\\
&\quad \mathbb{E}_{\boldsymbol{x}\sim D}\frac{1}{2}[\mathrm{H}(\phi(\boldsymbol{f}'(\psi(\boldsymbol{x})))) + \mathrm{H}(\phi(\boldsymbol{f}''(\psi(\boldsymbol{x}))))].
\end{aligned} \tag{74}$$

It is obvious that the objective of CE (25) equals to the combination of objective of KL (24) and terms related to the entropy of class probabilities of $\boldsymbol{f}'$ and $\boldsymbol{f}''$.

**Proposition D.1 (Proposition 3).** *Given the ramp loss $\Phi_\rho$ defined as (5), there exists a distance measure $\varphi: \mathbb{R}\times\mathbb{R}\to\mathbb{R}_+$ defined as*

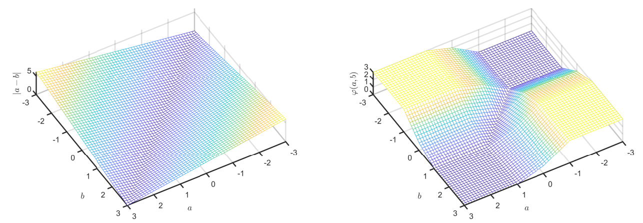
$$\varphi(a,b) = (K-1)|\Phi_\rho(-a) - \Phi_\rho(-b)| + |\Phi_\rho(a) - \Phi_\rho(b)|,$$

*such that the matrix-formed $\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}''(\boldsymbol{x}))\|_1$ in MCSD (6) can be calculated as the sum of $\varphi$-distance values of $K$ entry pairs between $f'_k(\boldsymbol{x})$ and $f''_k(\boldsymbol{x})$, i.e.,*

$$\|\boldsymbol{M}^{(\rho)}(\boldsymbol{f}'(\boldsymbol{x})) - \boldsymbol{M}^{(\rho)}(\boldsymbol{f}''(\boldsymbol{x}))\|_1 = \sum_{k=1}^{K}\varphi(f'_k(\boldsymbol{x}), f''_k(\boldsymbol{x})).$$

*Proof.* It is obvious that the function $\varphi$ satisfies the properties of symmetry, non-negative, and triangle inequality, and thus it is a distance measure. The proposition follows directly from the definitions of $\varphi$ and $\boldsymbol{M}$ (see Equation (7)). □

To intuitively understand the $\varphi$-distance defined above, we plot in Figure D the values of $|a-b|$ and $\varphi(a,b)$ as the functions of $a$ and $b$. We can see that the defining $\varphi$-distance $\varphi(a,b)$ can be considered as a variant form of the absolute distance $|a-b|$. From the Figure D, we can also see that maximizing (minimizing) $\varphi(a,b)$ can be achieved by maximizing (minimizing) the difference between $a$ and $b$.



(a) $|a-b|$ (b) $\varphi(a,b)$

Fig. D. Plotting of the values of (a) $|a-b|$ and (b) $\varphi(a,b)$ with the arguments $a$ and $b$. Here we set $K=3$ and $\rho=1$. The quality of plotting is similar for other values of $K$ and $\rho$.

**Explanation of** $\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}$ **(35)**

Based on the equations of (74), we have

$$\begin{aligned}&\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)\\ =&\frac{1}{2n_t}\sum_{j=1}^{n_t}[\text{KL}(\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t_j),\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t_j))\\ &\qquad+\text{KL}(\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t_j),\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t_j))]\\ +&\frac{1}{2n_t}\sum_{j=1}^{n_t}[\widetilde{\text{H}}(\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t_j))+\widetilde{\text{H}}(\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t_j))],\end{aligned} \tag{75}$$

where the $\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t)$ and $\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t)$ are vectors composed with the first $K$ and last $K$ values of $\boldsymbol{p}^{st}(\boldsymbol{x}^t)$, respectively. The KL-divergence terms encourage the agreement of $\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t)$ and $\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t)$ for any target instance $\boldsymbol{x}^t$. The $\widetilde{\text{H}}$ terms share the same formulation with the entropy function. Although $\sum_{k=1}^{K}p^{st}_k(\boldsymbol{x}^t)\leq 1$ and $\sum_{k=K}^{2K}p^{st}_k(\boldsymbol{x}^t)\leq 1$, minimizing the $\widetilde{\text{H}}$ terms encourages both the $\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t_j)$ and $\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t_j)$ to be vectors with only one non-zero value, whose proof is almost the same as that of the entropy loss. In consideration of the terms of KL-divergence and $\widetilde{\text{H}}$, as well as the intrinsic sum-to-one-constraint, i.e., $\sum_{k=1}^{2K}p^{st}_k(\boldsymbol{x}^t)=1$, minimizing $\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ leads to $\boldsymbol{p}^{st}_{1:K}(\boldsymbol{x}^t)$ and $\boldsymbol{p}^{st}_{K:2K}(\boldsymbol{x}^t)$ as the same vector with only one non-zero value of 0.5 for any target instance $\boldsymbol{x}^t$.

**Proposition D.2 (Proposition 4).** *Let $\mathcal{F}$ be a rich enough space of continuous and bounded scoring functions, with the sum-to-zero constraint $\sum_{k=1}^{K}f_k=0$. For $\boldsymbol{f}^s,\boldsymbol{f}^t\in\mathcal{F}$ and a fixed function $\psi$ that satisfies $\psi(\boldsymbol{x}_1)\neq\psi(\boldsymbol{x}_2)$ when $y_1\neq y_2$, $\exists\,\rho>0$ such that, minimizer $\boldsymbol{f}^{s*}$ of $\mathcal{L}^s_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)$ in (37) also minimizes a term $\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)$ in (21) of empirical source error defined on $\boldsymbol{f}^s$, and minimizer $\boldsymbol{f}^{t*}$ of $\mathcal{L}^t_{\widehat{P}^{\psi}}(\boldsymbol{f}^t)$ in (37) also minimizes a term $\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^t)$ in (21) of empirical source error defined on $\boldsymbol{f}^t$.*

*Proof.* We first restate the definition of $\mathcal{L}^s_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)$ and $\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)$ as

$$\begin{aligned}\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)&=\mathbb{E}_{(\boldsymbol{x},y)\sim\widehat{P}}\sum_{k=1}^{K}\Phi_\rho(\mu_k(\boldsymbol{f}^s(\psi(\boldsymbol{x})),y)),\\ \mathcal{L}^s_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)&=\mathbb{E}_{(\boldsymbol{x},y)\sim\widehat{P}}-\log(\phi_y(\boldsymbol{f}^s(\psi(\boldsymbol{x})))),\end{aligned} \tag{76}$$

where $\phi$ is the softmax operator. Under the assumption that $\psi(\boldsymbol{x}_1)\neq\psi(\boldsymbol{x}_2)$ for each example with $y_1\neq y_2$, if the scoring function space is rich enough, then minimizer $\boldsymbol{f}^{s*}$ of $\mathcal{L}^s_{\widehat{P}^{\psi}}(\boldsymbol{f}^s)$ results in $\phi_y(\boldsymbol{f}^{s*}(\psi(\boldsymbol{x})))$ reaching to the maximum value for each example $(\boldsymbol{x},y)\sim\widehat{P}$. Since the scoring function is bounded, we assume that $\|\boldsymbol{f}^{s*}(\psi(\boldsymbol{x}))\|_\infty\leq M$, and

$$\phi_y(\boldsymbol{f}^{s*}(\psi(\boldsymbol{x})))=\frac{\exp(f^{s*}_y(\psi(\boldsymbol{x})))}{\exp(f^{s*}_y(\psi(\boldsymbol{x})))+\sum_{k\neq y}\exp(f^{s*}_k(\psi(\boldsymbol{x})))}.$$

With the sum-to-zero constraint $\sum_{k=1}^{K}f^{s*}_k(\psi(\boldsymbol{x}))=0$, it is not hard to verify that $f^{s*}_y(\psi(\boldsymbol{x}))=M$ and $f^{s*}_k(\psi(\boldsymbol{x}))=-M/(K-1), k\neq y$. Therefore, for any $\rho\leq M/(K-1)$, we have $\sum_{k=1}^{K}\Phi_\rho(\mu_k(\boldsymbol{f}^{s*}(\psi(\boldsymbol{x})),y))=0$ and thus $\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^{s*})=0$. Similarly, minimizer $\boldsymbol{f}^{t*}$ of $\mathcal{L}^t_{\widehat{P}^{\psi}}(\boldsymbol{f}^t)$ results in $\mathcal{E}^{(\rho)}_{\widehat{P}^{\psi}}(\boldsymbol{f}^{t*})=0$. □

**Proposition D.3 (Proposition 5).** *For $\psi$ of a function space of enough capacity and fixed functions $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ with the same range, minimizer $\psi^*$ of $\text{ConFUSE}^{st}_{\widehat{P}^{\psi}}(\boldsymbol{f}^s,\boldsymbol{f}^t)+\lambda\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ with the parameter $\lambda>0$ in (37) zeroizes $\text{MCSD}^{(\rho)}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)-\text{MCSD}^{(\rho)}_{\widehat{P}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ in (21) of empirical MCSD divergence defined on $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$.*

*Proof.* The proof is trivial because minimizer $\psi^*$ of $\text{ConFUSE}^{st}_{\widehat{P}^{\psi}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ results in $\boldsymbol{f}^s(\psi^*(\boldsymbol{x}))=\boldsymbol{f}^t(\psi^*(\boldsymbol{x}))$ for each example $(\boldsymbol{x},y)\sim\widehat{P}$, and therefore $\text{MCSD}^{(\rho)}_{\widehat{P}^{\psi^*}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)=0$; Similarly, minimizer $\psi^*$ of $\text{ConFUSE}^{st}_{\widehat{Q}^{\psi}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)$ also results in $\text{MCSD}^{(\rho)}_{\widehat{Q}^{\psi^*}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)=0$ and thus $\text{MCSD}^{(\rho)}_{\widehat{Q}^{\psi^*}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)-\text{MCSD}^{(\rho)}_{\widehat{P}^{\psi^*}_{x}}(\boldsymbol{f}^s,\boldsymbol{f}^t)=0$. □

# APPENDIX E EXPERIMENTS

## E.1 Datasets and Implementations

**Office-31** The office-31 dataset [52] is a standard benchmark dataset for domain adaptation, which contains $4,110$ images of 31 categories shared by three distinct domains: *Amazon* (**A**), *Webcam* (**W**) and *DSLR* (**D**). We adopt it in the closed set, partial, and open set UDA.

**ImageCLEF-DA** The ImageCLEF-DA dataset [54] is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge, which contains three domains: *Caltech-256* (**C**), *ImageNet ILSVRC 2012* (**I**) and *Pascal VOC 2012* (**P**). For each domain, there are 12 categories and 50 images in each class. The three domains in this dataset are of the same size, which is a good complementation of the Office-31 dataset where different domains are of different sizes. We adopt it in the closed set settng of UDA.

**Office-Home** The Office-Home dataset [53] is a very challenging dataset for domain adaptation, which contains $15,500$ images from 65 categories of everyday objects in the office and home scenes, shared by four significantly different domains: Artistic images (**A**), Clip Art (**C**), Product images (**P**) and Real-World images (**R**). We adopt it in the closed set and partial UDA.

**Syn2Real** The Syn2Real dataset [55], [60] is a challenging simulation-to-real dataset, which contains over 280K images of 12 categories. We adopt the training domain, which contains synthetic images generated by rendering 3D models from different angles and under different lighting conditions, and validation domain, which contains natural images, as source domain and target domain, respectively. We adopt it in the closed set and open set UDA. For open set UDA, there are additional 33 background categories and 69 other categories aggregated as the unknown class of source domain and target domain, respectively.

**Digits** The MNIST [56], SVHN [57], and USPS [58] datasets are adopted in the closed set UDA. Following [19], we adopt the modified LeNet and evaluate on three adaptation tasks of SVHN $to$ MNIST, MNIST $\rightarrow$ USPS, and USPS $\rightarrow$ MNIST. Following [14], we sample $2,000$ images from MNIST and $1,800$ images from USPS for adaptation between MNIST and USPS, and use the full training sets for the SVHN $\rightarrow$ MNIST task.

**Modified LeNet Implementation** Following [19], we adopt the modified LeNet for the Digits datasets [56], [57], [58]. All parameters are updated with the Adam optimizer with a learning rate of 0.0002, a $\beta_1$ of 0.5, a $\beta_2$ of 0.999, and a batch size of 256 images. We convert all training images to greyscale and scale them to $28 \times 28$ pixels.

### E.2 Analysis

**Full Results of Different Implementations of McDalNets (21)** We present the full results of different implementations of McDalNets (21) on datasets of Office-31 [52], ImageCLEF-DA [54], Office-Home [53], VisDA-2017 [55], and Digits [56], [57], [58] in Table 12, Table 13, Table 14, Table 15, and Table 16, respectively.

**Visualization with Class Information** We visualize the network activations from the feature extractor of “DANN” and “SymmNets-V2” on the adaptation task of **A** $\rightarrow$ **W** by t-SNE [64] with class information in Figure E. The samples of the same class across domains are aligned intuitively with the features of SymmNets-V2.

### E.3 Results

**Results Based on the AlexNet Structure** To illustrate the generalization of our SymmNets-V2 to different network structures, we additionally implement SymmNets based on the AlexNet [2]. Given an AlexNet [2] pre-trained on the ImageNet dataset [61], the feature extractor $\psi$ is the AlexNet without $fc8$ layer, and an additional bottleneck layer added to $fc7$ layer with dimension of 256 following [13]. Other settings are the same as that for the ResNet. Results for the closed set and partial UDA tasks are respectively presented in Table 17 and Table 18, certifying the effectiveness and generalization of SymmNets-V2 on various model structures.

TABLE 12
Accuracies (%) of different instantiations of McDalNets on the Office-31 [52] dataset for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A → W | D → W | W → D | A → D | D → A | W → A | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [65] | 79.9±0.3 | 96.6±0.4 | 99.4±0.2 | 84.1±0.4 | 64.5±0.3 | 66.4±0.4 | 81.8 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | | | | |
| DANN [13], [51] (30) | 82.2±0.2 | 98.2±0.2 | 99.8±0.2 | 84.1±0.3 | 66.3±0.4 | 66.4±0.2 | 82.8 |
| MDD [18] variant (28) | 86.5±1.2 | 98.2±0.3 | 99.8±0.2 | 87.3±0.5 | 67.9±0.3 | 67.7±0.1 | 84.5 |
| McDalNets based on the following surrogates of MCSD (6) | | | | | | | |
| $L_1$/MCD [16] (23) | 84.8±0.1 | 98.2±0.3 | 99.8±0.2 | 86.8±0.3 | 69.8±0.1 | 68.6±0.4 | 84.7 |
| KL (24) | 85.3±0.5 | 98.5±0.1 | 99.8±0.2 | 86.2±0.3 | 69.6±0.6 | 68.3±0.1 | 84.6 |
| CE (25) | 88.0±0.2 | 98.5±0.2 | 100.0±.0 | 86.9±0.2 | 70.0±0.6 | 68.6±0.4 | 85.3 |
| SymmNets-V2 (37) | **94.2**±0.1 | **98.8**±.0 | **100.0**±.0 | **93.5**±0.3 | **74.4**±0.1 | **73.4**±0.2 | **89.1** |

TABLE 13
Accuracies (%) of different instantiations of McDalNets on the ImageCLEF [54] dataset for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | I → P | P → I | I → C | C → I | C → P | P → C | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [65] | 74.7±1.0 | 87.3±0.5 | 93.0±0.3 | 83.5±0.3 | 67.5±0.3 | 90.2±0.8 | 82.7 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | | | | |
| DANN [13], [51] (30) | 77.3±0.1 | 90.7±0.3 | 94.3±0.2 | 88.3±0.2 | 73.5±0.8 | 92.7±0.1 | 86.1 |
| MDD [18] variant (28) | 77.2±0.3 | 91.8±0.2 | 95.0±0.2 | 87.8±0.6 | 73.7±0.5 | 94.7±0.3 | 86.7 |
| McDalNets based on the following surrogates of MCSD (6) | | | | | | | |
| $L_1$/MCD [16] (23) | 77.8±0.2 | 91.8±0.3 | 94.8±0.1 | 89.7±0.3 | 75.2±0.5 | 93.2±0.4 | 87.0 |
| KL (24) | 77.7±0.2 | 91.3±0.1 | 95.3±0.2 | 91.0±0.2 | 76.0±0.3 | 94.2±0.2 | 87.6 |
| CE (25) | 78.2±0.1 | 91.7±0.5 | 95.8±0.4 | 91.5±0.3 | 75.3±0.1 | 94.5±0.2 | 87.8 |
| SymmNets-V2 (37) | **79.0**±0.3 | **93.5**±0.2 | **96.9**±0.2 | **93.4**±0.3 | **79.2**±0.3 | **96.2**±0.1 | **89.7** |

TABLE 14
Accuracies (%) of different instantiations of McDalNets on the Office-Home [53] dataset for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

| Methods | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Only [65] | 40.5 | 66.1 | 74.3 | 53.2 | 61.2 | 63.9 | 52.6 | 37.5 | 72.3 | 65.5 | 43.2 | 77.0 | 58.9 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | | | | | | | | | | |
| DANN [13], [51] (30) | 42.9 | 65.5 | 74.3 | 54.5 | 60.6 | 65.4 | 54.0 | 40.3 | 73.1 | 66.7 | 45.4 | 76.9 | 60.0 |
| MDD [18] variant (28) | 33.2 | 64.2 | 75.0 | 58.9 | 62.4 | 68.3 | 57.7 | 43.0 | 75.5 | 70.1 | 46.0 | 79.0 | 61.1 |
| McDalNets based on the following surrogates of MCSD (6) | | | | | | | | | | | | | |
| $L_1$/MCD [16] (23) | 45.4 | 67.2 | 75.2 | 58.3 | 62.9 | 68.2 | 56.7 | 42.8 | 73.9 | 67.5 | 47.9 | 78.0 | 62.0 |
| KL (24) | 46.6 | 69.2 | 75.2 | 59.9 | 65.1 | 68.2 | 60.2 | 45.6 | 73.8 | 67.3 | 50.4 | 77.7 | 63.3 |
| CE (25) | 46.6 | 69.2 | 75.6 | 59.9 | 65.1 | 68.8 | 61.4 | 45.8 | 74.8 | 68.8 | 52.1 | 79.6 | 64.0 |
| SymmNets-V2 (37) | **48.1** | **74.3** | **78.7** | **64.6** | **71.8** | **74.1** | **64.4** | **50.0** | **80.2** | **74.3** | **53.1** | **83.2** | **68.1** |

TABLE 15
Accuracies (%) of different instantiations of McDalNets on the VisDA-2017 [55] dataset for closed set UDA. Results are based on models adapted from a 50-layer ResNet.

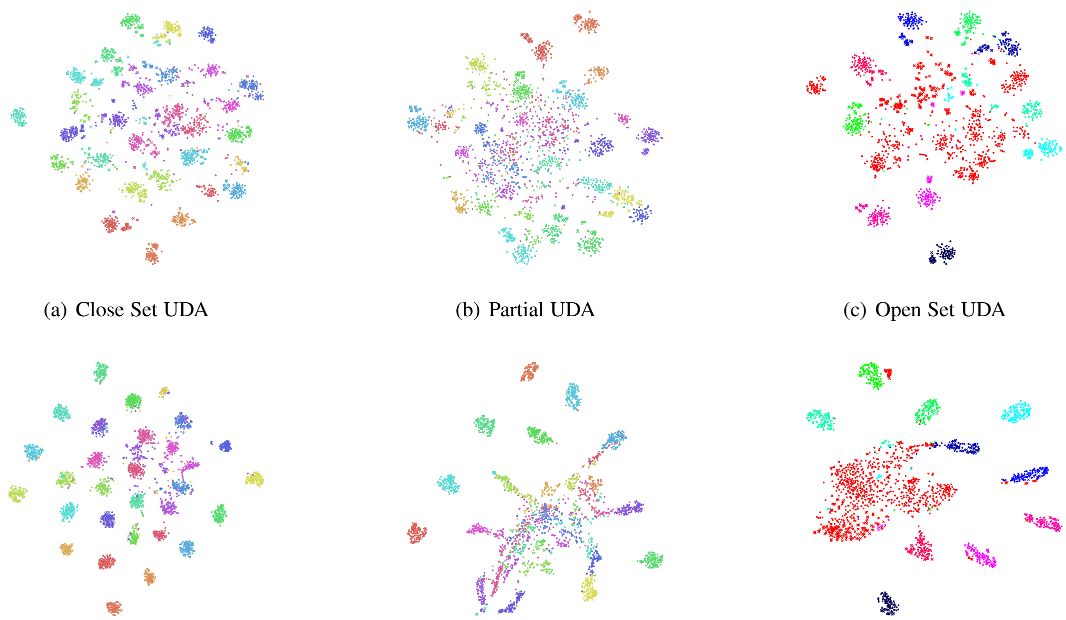
| Methods | plane | bcycle | bus | car | horse | knife | mcycl | person | plant | sktbrd | train | trunk | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Only [65] | 68.2 | 10.9 | 35.3 | 75.7 | 53.6 | 2.7 | 74.1 | 4.7 | 61.8 | 18.9 | 90.5 | 4.3 | 41.8 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | | | | | | | | | | |
| DANN [13], [51] (30) | 77.1 | 35.7 | 68.0 | 59.0 | 75.8 | 20.1 | 89.3 | 42.1 | 86.3 | 38.8 | 85.9 | 22.5 | 58.4 |
| MDD [18] variant (28) | did not converge | | | | | | | | | | | | |
| McDalNets based on the following surrogates of MCSD (6) | | | | | | | | | | | | | |
| $L_1$/MCD [16] (23) | 84.8 | 60.0 | 75.6 | 75.5 | 82.5 | 76.5 | 93.0 | 73.1 | 92.8 | 28.2 | **90.9** | 10.4 | 70.4 |
| KL (24) | **89.3** | **62.9** | 70.6 | 70.4 | **83.5** | **83.1** | 92.5 | 68.9 | 91.5 | 6.6 | 91.0 | 18.3 | 69.0 |
| CE (25) | 86.5 | 56.7 | 78.0 | 72.9 | 80.8 | 81.3 | **93.7** | **76.5** | **94.1** | 20.0 | 87.6 | 16.7 | 70.5 |
| SymmNets-V2 (37) | 87.3 | 62.2 | 79.1 | 66.7 | 80.3 | 79.7 | 87.8 | 75.6 | 88.9 | **31.4** | 90.7 | **25.8** | **71.3** |

TABLE 16
Accuracies (%) of different instantiations of McDalNets on the Digits [56], [57], [58] dataset for closed set UDA. Results are based on models adapted from a modified LeNet.

| Methods | S → M | U → M | M → U | Avg |
|---|---|---|---|---|
| Source Only | 62.7±1.1 | 77.5±2.2 | 71.2±0.7 | 70.5 |
| McDalNets based on the following surrogates of degenerate MCSD (13) (12) | | | | |
| DANN [13], [51] (30) | 74.2±1.0 | 73.0±2.9 | 70.3±1.5 | 72.5 |
| MDD [18] variant (28) | | did not converge | | did not converge |
| McDalNets based on the following surrogates of MCSD (6) | | | | |
| $L_1$/MCD [16] (23) | 90.4±0.4 | 95.8±0.6 | 85.7±1.9 | 90.6 |
| KL (24) | 76.6±1.3 | 94.5±0.7 | 77.5±0.6 | 82.9 |
| CE (25) | **97.8**±0.2 | 96.6±0.6 | 90.3±0.8 | 94.9 |
| SymmNets-V2 (37) | 96.3±1.2 | **96.8**±0.3 | **94.8**±0.6 | **96.0** |

DANN [13], [51]

SymmNets-V2



(a) Close Set UDA (b) Partial UDA (c) Open Set UDA

(d) Close Set UDA (e) Partial UDA (f) Open Set UDA

Fig. E. The t-SNE visualization of class-labeled feature representations learned by DANN (top row) and SymmNets-V2 (bottom row) under the settings of closed set, partial, and open set UDA. The point marks ("·") represent features of samples from the source domain **A** whereas the cross marks ("x") represent features of samples from the target domain **W**, where different colors represent different classes. In open set UDA, the red color indicates the unknown class. In partial UDA, we illustrate the feature representations learned by SymmNets-V2 (With active $\omega_k$), where we focus on the domain-shared classes and leave the source classes exclusive to the target domain as an indistinguishable cluster via the soft class weighting scheme, as discussed in Section 4.

TABLE 17
Accuracy (%) on the Office-Home dataset [53] for *closed set* UDA. Results are based on models adapted from a AlexNet.

| Methods | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Only [2] | 26.4 | 32.6 | 41.3 | 22.1 | 41.7 | 42.1 | 20.5 | 20.3 | 51.1 | 31.0 | 27.9 | 54.9 | 34.3 |
| DAN [36] | 31.7 | 43.2 | 55.1 | 33.8 | 48.6 | 50.8 | 30.1 | 35.1 | 57.7 | 44.6 | 39.3 | 63.7 | 44.5 |
| DANN [13], [51] | 36.4 | 45.2 | 54.7 | 35.2 | 51.8 | 55.1 | 31.6 | 39.7 | 59.3 | 45.7 | 46.4 | 65.9 | 47.3 |
| CDAN+E [15] | **38.1** | 50.3 | 60.3 | 39.7 | 56.4 | 57.8 | 35.5 | **43.1** | 63.2 | 48.4 | **48.5** | 71.1 | 51.0 |
| **SymmNets-V1** [17] | 37.4 | **53.9** | 60.9 | **40.0** | 56.3 | **58.5** | 34.7 | 40.1 | 64.0 | **49.6** | 46.7 | **71.6** | **51.1** |
| **SymmNets-V2** | 36.5 | 53.8 | **61.2** | **40.0** | **57.0** | 58.1 | **36.2** | 39.8 | **64.2** | 48.8 | 46.1 | 71.2 | **51.1** |
| GCAN [74] | 36.4 | 47.3 | 61.1 | 37.9 | 58.3 | 57.0 | 35.8 | **42.7** | 64.5 | **50.1** | **49.1** | **72.5** | 51.1 |
| **SymmNets-V2-SC** | **38.6** | **61.4** | **65.8** | **41.2** | **59.6** | **63.4** | **37.7** | 39.4 | **66.4** | 49.2 | 47.1 | 71.4 | **53.4** |

TABLE 18
Accuracy (%) on the Office-31 dataset [52] for *partial* UDA. Results are based on models adapted from a AlexNet.

| Methods | A → W | D → W | W → D | A → D | D → A | W → A | Avg |
|---|---|---|---|---|---|---|---|
| Source Only [2] | 58.51 | 95.05 | 98.08 | 71.23 | 70.60 | 67.74 | 76.87 |
| DAN [36] | 56.58 | 71.86 | 86.78 | 51.86 | 50.42 | 52.29 | 61.62 |
| DANN [13], [51] | 49.49 | 93.55 | 90.44 | 49.68 | 46.72 | 48.81 | 63.11 |
| SAN [23] | 80.02 | 98.64 | **100.00** | 81.28 | 80.58 | 83.09 | 87.27 |
| Zhang *et al.* [75] | 76.27 | **98.98** | **100.00** | 78.98 | **89.46** | 81.73 | 87.57 |
| **SymmNets-V2** | 76.62 | 79.30 | 99.37 | 82.83 | 71.33 | 83.19 | 82.11 |
| **SymmNets-V2** (With active $\omega_k$) | **82.71** | 94.90 | 98.72 | **85.35** | 83.50 | **93.00** | **89.70** |