The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

# Unsupervised Domain Adaptation with Distribution Matching Machines

**Yue Cao, Mingsheng Long,* Jianmin Wang**
KLiss, MOE; NEL-BDS; TNList; School of Software, Tsinghua University, China
caoyue10@gmail.com mingsheng@tsinghua.edu.cn jimwang@tsinghua.edu.cn

## Abstract

Domain adaptation generalizes a learning model across source domain and target domain that follow different distributions. Most existing work follows a two-step procedure: first, explores either feature matching or instance reweighting independently, and second, train the transfer classifier separately. In this paper, we show that either feature matching or instance reweighting can only reduce, but not remove, the cross-domain discrepancy, and the knowledge hidden in the relations between the data labels from the source and target domains is important for unsupervised domain adaptation. We propose a new Distribution Matching Machine (DMM) based on the structural risk minimization principle, which learns a transfer support vector machine by extracting invariant feature representations and estimating unbiased instance weights that jointly minimize the cross-domain distribution discrepancy. This leads to a robust transfer learner that performs well against both mismatched features and irrelevant instances. Our theoretical analysis proves that the proposed approach further reduces the generalization error bound of related domain adaptation methods. Comprehensive experiments validate that the DMM approach significantly outperforms competitive methods on standard domain adaptation benchmarks.

## Introduction

Standard supervised learning machines will encounter poor generalization performance with limited training data, while manual labeling of sufficient training data for emerging application domains is prohibitive. Hence there is motivation to design effective algorithms to reduce the labeling cost by leveraging rich labeled data from relevant source domains to the target domains. Domain adaptation (Quionero-Candela et al. 2009; Pan and Yang 2010) addresses the problem that we have data from two related domains but under different distributions. The domain discrepancy poses a major obstacle for adapting predictive models across domains. Domain adaptation establishes knowledge transfer from the labeled source domain to the unlabeled target domain by exploring domain-invariant knowledge structures that manifest the similarity between domains under substantial discrepancy.

*Corresponding author: Mingsheng Long


The major computational problem of domain adaptation is how to reduce the distribution difference between domains. One successful approach to this problem is learning domain-invariant features by jointly minimizing a distance metric that well characterizes the cross-domain discrepancy (Pan, Kwok, and Yang 2008; Pan et al. 2011; Duan, Xu, and Tsang 2012; Duan, Tsang, and Xu 2012; Long et al. 2013; Zhang et al. 2013; Long et al. 2015a; Ganin and Lempitsky 2015; Sun, Feng, and Saenko 2016). However, when the cross-domain discrepancy is substantially large, there will always be some source instances that are irrelevant to the target domain even using domain-invariant features, which may introduce large bias to the transfer-classifier (Long et al. 2014; Aljundi et al. 2015). Another principled strategy is to estimate the weights (importance) of the source instances such that the distribution discrepancy can be minimized for the empirical risk minimization learning (Huang et al. 2006; Bruzzone and Marconcini 2010; Chen, Weinberger, and Blitzer 2011; Yu and Szepesvári 2012; Chu, De la Torre, and Cohn 2013). However, when the cross-domain discrepancy is substantially large, a large number of source instances will be down-weighed, leading to a smaller set of effective instances for training the transfer-classifier. This will result in a domain-unbiased but high-variance transfer-classifier, which is not robust to large cross-domain discrepancy.

In the aforementioned challenging scenario, either feature matching or instance reweighting can only reduce, but not remove, the cross-domain discrepancy, and it is inevitable to perform feature matching and instance reweighting jointly for robust unsupervised domain adaptation (Long et al. 2014; Aljundi et al. 2015). FurthSRMore, the knowledge hidden in the relations between the data labels from the source and target domains is important for learning a transfer classifier that is coherent with discriminative structure underlying the data distributions. However, most existing work follows a two-step procedure: first, explores either feature matching or instance reweighting independently, and second, train the transfer classifier separately. All in all, the knowledge that is safely transferable across domains should be (1) invariant to feature representations, (2) unbiased to irrelevant instances, and (3) consistent with the discriminative structure. To our best knowledge, there is no previous work that can optimize all the three inevitable learning criteria in a unified learning model for unsupervised domain adaptation.

In this paper, we tackle a more challenging domain adaptation scenario where the source and target are different due to both mismatched features and irrelevant instances. We propose a new Distribution Matching Machine (DMM) based on the structural risk minimization principle (Vapnik 1998), which learns a transfer support vector machine by extracting invariant feature representations and estimating unbiased instance weights that jointly minimize the cross-domain distribution discrepancy. Specifically, DMM is implemented by jointly minimizing the structural risk and a nonparametric cross-domain distribution discrepancy, the *Maximum Mean Discrepancy* (MMD) (Gretton et al. 2012). From the DMM model, we can jointly learn a transfer support vector machine, an invariant feature transformation and an unbiased instance reweighting. Our theoretical analysis based on (Ben-David et al. 2007) proves that the proposed approach further reduces the generalization error bound of related domain adaptation methods. Extensive experiments validate that DMM can outperform state of the art methods.

## Related Work

According to the survey (Pan and Yang 2010), existing domain adaptation methods can be roughly organized into two categories: *feature matching* and *instance reweighting*. Feature matching methods aim to reduce the distribution difference by learning a new feature representation, which can be learned via (1) extracting domain-invariant latent factors (Jhuo et al. 2012; Qiu et al. 2012; Fernando et al. 2013), (2) minimizing proper distance measures (Pan, Kwok, and Yang 2008; Pan et al. 2011; Long et al. 2013), and (3) reweighting relevant features with sparsity-promoting regularization (Argyriou and Evgeniou 2006; Masaeli, Fung, and Dy 2010). Instance reweighting methods aim to reduce the distribution difference by reweighting the source instances according to their relevance to the target instances (Huang et al. 2006; Bruzzone and Marconcini 2010; Chen, Weinberger, and Blitzer 2011; Chu, De la Torre, and Cohn 2013). However, these methods explored feature matching and instance reweighting independently, which is ineffective when the domain difference is substantially large.

Recent advances show that deep networks can learn abstract feature representations that can only reduce, but not remove, the cross-domain discrepancy (Glorot, Bordes, and Bengio 2011), resulting in unbounded risk for target tasks (Mansour, Mohri, and Rostamizadeh 2009; Ben-David et al. 2010). Some recent work bridges deep learning and domain adaptation (Long et al. 2015a; Ganin and Lempitsky 2015; Tzeng et al. 2015; Long et al. 2016; 2017; Tzeng et al. 2017), which extends deep convolutional networks (CNNs) to domain adaptation by adding adaptation layers through which the mean embeddings of distributions are matched (Long et al. 2015a; 2016; 2017), or by adding a subnetwork as domain discriminator while the deep features are learned to confuse the discriminator in a domain-adversarial training paradigm (Ganin and Lempitsky 2015; Tzeng et al. 2015; 2017). While performance was significantly improved, these state of the art methods may be restricted by the assumption that all the source instances may be useful for the target task. This assumption is violated in some difficult transfer setting where there may be some outlier source instances that are irrelevant to the target domain.

The most similar work to the proposed DMM approach is Transfer Joint Matching (TJM) (Long et al. 2014) and Landmarks Selection-based Subspace Alignment (LSSA) (Aljundi et al. 2015). However, DMM clearly contrasts from TJM and LSSA (Long et al. 2014; Aljundi et al. 2015) in two key aspects: (1) DMM jointly learns the transfer classifier and the transferable knowledge (invariant feature representations and unbiased instance weights) in an end-to-end learning paradigm, while TJM and LSSA require a two-step procedure where the first step learns the transferable knowledge and the second step learns the transfer classifier. This is suboptimal as the learned transferable knowledge may not be consistent with the discriminative structure. (2) DMM learns the unbiased instance reweighting by the principled density ratio estimation (Huang et al. 2006) that is more theoretically guaranteed, while TJM and LSSA learn the reweighting by landmarks selection heuristics. All in all, the proposed DMM approach can jointly learn the transfer classifier and transferable knowledge with statistical guarantees.

## Distribution Matching Machine

In unsupervised domain adaptation, we are given a source domain $\mathcal{X}_s = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ with $n$ labeled examples, and a target domain $\mathcal{X}_t = \{\mathbf{x}_j\}_{j=n+1}^{n+n'}$ with $n'$ unlabeled examples, where the source domain and target domain follow different probability distributions $p$ and $q$, respectively. In this paper, we propose a new Distribution Matching Machine (DMM) based on the structural risk minimization principle, which learns a transfer support vector machine $y = f(\mathbf{x})$ that minimizes target risk $R_{\mathcal{X}_t}(f) = \Pr_{(\mathbf{x},y)\sim q}[f(\mathbf{x}) \neq y]$ from source and target data by jointly extracting invariant feature representations and estimating unbiased instance weights. Notations and their descriptions are summarized in Table 1.

Table 1: Notations and their descriptions used in this paper.

| Notation | Description | Notation | Description |
|---|---|---|---|
| $\mathcal{X}_s, \mathcal{X}_t$ | source/target domain | $k, \kappa$ | kernel functions |
| $n, n'$ | #source/target examples | $\phi, \psi$ | kernel feature maps |
| $d$ | feature dimension | $\mathbf{A}$ | feature transformation |
| $r$ | subspace dimension | $\boldsymbol{\alpha}$ | instance weights |
| $C$ | SVM penalty parameter | $\lambda$ | MMD penalty parameter |

### Model

**Structural Risk Minimization** Domain adaptation is challenging since the target domain has no (or only limited) labeled information while the source domain follows different distributions with the target domain. To approach this problem, many existing methods aim to bound the target error by the source error plus a discrepancy metric between the source and the target, which is theoretically supported by domain adaptation learning bounds (Ben-David et al. 2007; Mansour, Mohri, and Rostamizadeh 2009). In this paper, we propose to jointly learn a transfer support vector machine with invariant feature transformation $\mathbf{A} \in \mathbb{R}^{d\times r}$ and unbiased instance reweighting $\boldsymbol{\alpha} \in \mathbb{R}^{n}$ under the structural risk

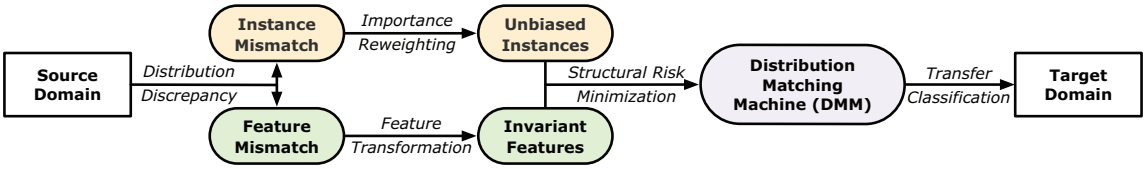



Figure 1: An illustration of the proposed Distribution Matching Machine (DMM) model for unsupervised domain adaptation.

minimization (SRM) framework, such that both mismatched features and irrelevant instances can be adapted by minimizing the distribution discrepancy. Based on the SRM principle (Vapnik 1998), we propose to learn a domain-transfer support vector machine (SVM) $f$ so that the structural target risk $\hat{R}_{\mathcal{X}_t}(f) = \sum_{(\mathbf{x},y)\in\mathcal{X}_t} q(\mathbf{x})\,\ell(y, f(\mathbf{x}))$ is minimized as

$$\begin{aligned}\hat{R}_{\mathcal{X}_t}(f) &= \sum_{(\mathbf{x},y)\in\mathcal{X}_t} q(\mathbf{Ax})\,\ell(y, f(\mathbf{Ax})) + \frac{1}{C}\Omega(f) \\ &\approx \sum_{(\mathbf{x},y)\in\mathcal{X}_s} p(\mathbf{Ax})\,\alpha(\mathbf{Ax})\,\ell(y, f(\mathbf{Ax})) + \frac{1}{C}\Omega(f) \\ &\approx \sum_{(\mathbf{x},y)\in\mathcal{X}_s} \alpha(\mathbf{Ax})\,\ell(y, f(\mathbf{Ax})) + \frac{1}{C}\Omega(f),\end{aligned} \tag{1}$$

where $\ell$ and $\Omega$ are loss function and complexity term of classifier $f$, $C$ is the tradeoff parameter, and $\alpha(\mathbf{Ax}) \triangleq \frac{q(\mathbf{Ax})}{p(\mathbf{Ax})}$ is the *importance weight* of source instance $\mathbf{x}$ after feature transformation $\mathbf{A}$. Consistent with previous work (Pan et al. 2011), we utilize the assumption that the conditional distribution remains unchanged in the new feature space transformed by matrix $\mathbf{A}$, i.e. $p(y|\mathbf{Ax}) = q(y|\mathbf{Ax})$. This is more general than most importance reweighting methods (Huang et al. 2006), which requires the assumption that the conditional distribution remains unchanged in the original feature space, i.e. $p(y|\mathbf{x}) = q(y|\mathbf{x})$. In the structural risk minimization (SRM) framework (1), by adopting the Hinge loss $\ell(y, f(\mathbf{x})) = \max(0, 1 - yf(\mathbf{x}))$ and integrating feature transformation $\mathbf{A}$ and instance weights $\boldsymbol{\alpha}$, we obtain transfer support vector machine (SVM) for domain adaptation as

$$\begin{aligned}\min_{\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\alpha},\mathbf{A}} \; & \frac{1}{2}\|\boldsymbol{\theta}\|_\psi^2 + C\sum_{i=1}^{n}\alpha_i\xi_i \\ \text{s.t.} \quad & y_i\left(\boldsymbol{\theta}^\mathsf{T}\psi(\mathbf{Ax}_i) + b\right) \geqslant 1 - \xi_i \\ & \xi_i \geqslant 0, i = 1, \ldots, n,\end{aligned} \tag{2}$$

where $\boldsymbol{\theta}$, $b$ and $\boldsymbol{\xi}$ are the weight, bias parameters and slack variables of SVM, respectively, and $\psi(\cdot)$ is the kernel map, $\kappa(\mathbf{x}, \mathbf{x}') = \langle\psi(\mathbf{x}), \psi(\mathbf{x}')\rangle$. Based on the SRM principle, transfer SVM (2) trained on source labeled data can perform accurate prediction for target unlabeled data, given invariant feature transformation $\mathbf{A}$ and unbiased instance weights $\boldsymbol{\alpha}$.

**Dual Distribution Matching** However, only minimizing transfer SVM (2) is not enough to extract invariant feature transformation $\mathbf{A}$ and estimate unbiased instance weights $\boldsymbol{\alpha}$, since the cross-domain distribution discrepancy is not taken into account. To this end, we propose a new dual distribution matching strategy by minimizing a distance metric of cross-domain distribution discrepancy. Let $\mathcal{H}_k$ be the reproducing kernel Hilbert space (RKHS) induced with a characteristic kernel $k$. The *kernel mean embedding* of distribution $p$ in $\mathcal{H}_k$ is a unique element $\mu_k(p) = \mathbb{E}_p[\phi(\mathbf{x})]$ so that expectation satisfies $\mathbb{E}_{\mathbf{x}\sim p} f(\mathbf{x}) = \langle f(\mathbf{x}), \mu_k(p)\rangle_{\mathcal{H}_k}$ for all $f \in \mathcal{H}_k$. That is, all important information in distribution $p$ is encoded into embedding $\mu_k(p)$ and thus we can learn through $\mu_k(p)$ instead of $p$, which removes the nontrivial density estimation. The Maximum Mean Discrepancy (MMD) (Gretton et al. 2012) that measures the discrepancy between distributions $p$ and $q$ is defined as the squared RKHS-distance between the kernel mean embeddings of $p$ and $q$ as follows

$$D_k(p, q) \triangleq \|\mathbb{E}_p[\phi(\mathbf{x})] - \mathbb{E}_q[\phi(\mathbf{x}')]\|_{\mathcal{H}_k}^2, \tag{3}$$

where $\phi(\cdot)$ is a nonlinear feature map that induces RKHS $\mathcal{H}_k$, and $k(\mathbf{x}, \mathbf{x}') = \langle\phi(\mathbf{x}), \phi(\mathbf{x}')\rangle$. The two-sample test ensures that $p = q$ iff $D_k(p, q) = 0$ (Gretton et al. 2012). By minimizing MMD, we can match distributions $p$ and $q$.

Using MMD (3) as the cross-domain distribution discrepancy, we can perform dual distribution matching that jointly minimizes MMD with respect to both feature transformation $\mathbf{A}$ and instance weights $\boldsymbol{\alpha}$, which can yield invariant feature transformation $\mathbf{A}$ and unbiased instance weights $\boldsymbol{\alpha}$ as

$$\min_{\mathbf{A},\boldsymbol{\alpha}} D_k(p, q) = \|\mathbb{E}_p[\alpha(\mathbf{x})\,\phi(\mathbf{Ax})] - \mathbb{E}_q[\phi(\mathbf{Ax}')]\|_{\mathcal{H}_k}^2. \tag{4}$$

As directly computing the expectations of $p$ and $q$ in the infinite-dimensional kernel space $\phi(\mathbf{x})$ is difficult, we adopt the empirical estimate of MMD on all training data $\mathcal{X}_s \cup \mathcal{X}_t$,

$$\min_{\mathbf{A},\boldsymbol{\alpha}} D_k(\mathcal{X}_s, \mathcal{X}_t) = \left\|\sum_{i=1}^{n}\frac{\alpha_i\phi(\mathbf{Ax}_i)}{n} - \sum_{j=n+1}^{n+n'}\frac{\phi(\mathbf{Ax}_j)}{n'}\right\|_{\mathcal{H}_k}^2, \tag{5}$$

where we denote $\alpha_i = \alpha(\mathbf{A}^\mathsf{T}\mathbf{x}_i)$ for notation conciseness. Using kernel trick $k(\mathbf{x}, \mathbf{x}') = \langle\phi(\mathbf{x}), \phi(\mathbf{x}')\rangle$ and Gaussian kernel $k(\mathbf{x}, \mathbf{x}') = e^{-\|\mathbf{x}-\mathbf{x}'\|^2/\sigma}$ with parameter $\sigma$, we obtain

$$\min_{\mathbf{A},\boldsymbol{\alpha}} D_k(\mathcal{X}_s, \mathcal{X}_t) = \sum_{i=1}^{n+n'}\sum_{j=1}^{n+n'}\alpha_i'\alpha_j' e_i e_j k(\mathbf{Ax}_i, \mathbf{Ax}_j), \tag{6}$$

where $\boldsymbol{\alpha}'$ is the vector of weights for all the source and target instances, $\alpha_i' = \alpha_i$ if $\mathbf{x}_i \in \mathcal{X}_s$ and $\alpha_i' = 1$ if $\mathbf{x}_i \in \mathcal{X}_t$. Also, $\mathbf{e}$ is the MMD indicator vector, $e_i = 1/n$ if $\mathbf{x}_i \in \mathcal{X}_s$ and $e_i = -1/n'$ if $\mathbf{x}_i \in \mathcal{X}_t$. We introduce these two notations to avoid cluttered summation forms as (Gretton et al. 2012).

**Distribution Matching Machine** To enable unsupervised domain adaptation, the safely-transferable knowledge across domains should be (1) invariant to feature representations, i.e. optimal $\mathbf{A}$, (2) unbiased to irrelevant instances, i.e. optimal $\boldsymbol{\alpha}$, and (3) consistent with the discriminative structure,

i.e. optimal $\boldsymbol{\theta}$ and $b$. In structural risk minimization (SRM) framework (1), by integrating the feature transformation $\mathbf{A}$ and instance weights $\boldsymbol{\alpha}$ learned by dual distribution matching (5) into transfer SVM (2), we can obtain the Distribution Matching Machine (DMM) for robust domain adaptation as

$$\begin{aligned}\min_{\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\alpha},\mathbf{A}} \frac{1}{2}\left\|\boldsymbol{\theta}\right\|_{\psi}^{2}+C\sum_{i=1}^{n}\alpha_i\xi_i+\lambda D_k\left(\mathcal{X}_s,\mathcal{X}_t\right)\\ \text{s.t.}\quad y_i\left(\boldsymbol{\theta}^{\mathsf{T}}\psi\left(\mathbf{A}\mathbf{x}_i\right)+b\right)\geqslant 1-\xi_i\\ \xi_i\geqslant 0, i=1,\ldots,n\\ 0\leqslant\boldsymbol{\alpha}\leqslant B, \mathbf{1}^{\mathsf{T}}\boldsymbol{\alpha}=n,\end{aligned} \tag{7}$$

where $\boldsymbol{\theta}$, $b$ and $\boldsymbol{\xi}$ are the weight, bias parameters and slack variables of SVM, respectively, and $\psi(\cdot)$ is the kernel feature map for SVM; $\mathbf{A}$ is the invariant feature transformation and $\boldsymbol{\alpha}$ is the unbiased instance weights; $C$ is the tradeoff parameter between empirical risk and complexity term, and $\lambda$ is the relative importance of the MMD penalty term (5). The bounding constraint on $\boldsymbol{\alpha}$ guarantees that the instance weights are within proper ranges, and the equality constraint on $\boldsymbol{\alpha}$ guarantees that the learned instance weights $\boldsymbol{\alpha}$ follow a valid probability distribution (Huang et al. 2006). All these variables can be jointly learned by minimizing the DMM optimization problem (7). Based on the SRM principle, DMM trained on the source labeled data can perform accurate classification for the target unlabeled data, and can be robust to substantially large cross-domain distribution discrepancy caused by the mismatched features and irrelevant instances.

## Algorithm

The DMM optimization problem in Equation (7) consists of three variables, $\boldsymbol{\theta}$, $\boldsymbol{\alpha}$ and $\mathbf{A}$. Hence we adopt an alternating optimization paradigm, a variant of Coordinate Descent, to iteratively update one variable with the rest variables fixed.

**Update $\boldsymbol{\theta}$ and $b$** We update $\boldsymbol{\theta}$ and $b$, the weight and bias parameters of SVM, by fixing $\boldsymbol{\alpha}$ and $\mathbf{A}$ and rewrite (7) as

$$\begin{aligned}\min_{\boldsymbol{\theta},\boldsymbol{\xi}} \frac{1}{2}\left\|\boldsymbol{\theta}\right\|_{\psi}^{2}+C\sum_{i=1}^{n}\alpha_i\xi_i\\ \text{s.t.}\quad y_i\left(\boldsymbol{\theta}^{\mathsf{T}}\psi\left(\mathbf{A}\mathbf{x}_i\right)+b\right)\geqslant 1-\xi_i\\ \xi_i\geqslant 0, i=1,\ldots,n,\end{aligned} \tag{8}$$

which is a standard SVM readily solvable via its dual form. By representation theorem, $\boldsymbol{\theta}=\sum_{i=1}^{n}\boldsymbol{\vartheta}_i y_i\psi\left(\mathbf{A}\mathbf{x}_i\right)$, we get

$$\begin{aligned}\max_{\boldsymbol{\vartheta}} \sum_{i=1}^{n}\vartheta_i-\sum_{i=1}^{n}\sum_{j=1}^{n}y_i y_j\vartheta_i\vartheta_j\kappa\left(\mathbf{A}\mathbf{x}_i,\mathbf{A}\mathbf{x}_j\right)\\ \sum_{i=1}^{n}\vartheta_i y_i=0, 0\leqslant\vartheta_i\leqslant C\alpha_i.\end{aligned} \tag{9}$$

This problem can be efficiently solved by LIBSVM package.

**Update $\boldsymbol{\alpha}$** We update $\boldsymbol{\alpha}$, the unbiased instance weights of DMM, by fixing $\boldsymbol{\theta}$, $b$ and $\mathbf{A}$ and rewrite (7) as

$$\begin{aligned}\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{K}\boldsymbol{\alpha}-\left(\mathbf{h}-C\ell_i\right)^{\mathsf{T}}\boldsymbol{\alpha}\\ 0\leqslant\alpha_i\leqslant B, \mathbf{1}^{\mathsf{T}}\boldsymbol{\alpha}=n,\end{aligned} \tag{10}$$

where $\ell_i=\max\left(0,1-y_i f\left(\mathbf{A}\mathbf{x}_i\right)\right)$ is the loss of point $\mathbf{x}_i$; quadratic coefficients $\mathbf{K}_{ij}=k\left(\mathbf{A}\mathbf{x}_i,\mathbf{A}\mathbf{x}_j\right)$ and linear coefficients $h_i=\frac{n}{n'}\sum_{j=n+1}^{n+n'}k\left(\mathbf{A}\mathbf{x}_i,\mathbf{A}\mathbf{x}_j\right)$. This is a standard convex quadratic program (QP) and can be efficiently solved by off-the-shelf QP solvers, e.g. `quadprog` in MATLAB.

**Update $\mathbf{A}$** We update $\mathbf{A}$, the invariant feature transformation of DMM, by fixing $\boldsymbol{\theta}$, $b$ and $\boldsymbol{\alpha}$ and rewrite (7) as

$$\begin{aligned}\min_{\mathbf{A}} \sum_{i=1}^{n}\sum_{j=1}^{n}y_i y_j\vartheta_i\vartheta_j\kappa\left(\mathbf{A}\mathbf{x}_i,\mathbf{A}\mathbf{x}_j\right)\\ +\lambda\sum_{i=1}^{n+n'}\sum_{j=1}^{n+n'}\alpha'_i\alpha'_j e_i e_j k\left(\mathbf{A}\mathbf{x}_i,\mathbf{A}\mathbf{x}_j\right).\end{aligned} \tag{11}$$

This is a non-convex optimization problem which can be solved by the (mini-batch) Gradient Descent algorithm[1].

Each iteration takes $O(n^{2.3})$ to solve the weighted kernel SVM in (9), $O(n^3)$ to solve the quadratic program in (10), and $O(Tr(n+n')^2)$ to solve the non-convex optimization problem in (11), where $T$ is the number of iterations in Gradient Descent. The overall complexity $O(n^3)$ is comparable to related methods (Huang et al. 2006; Pan et al. 2011).

## Analysis

We analyze target risk by theory of domain adaptation (Ben-David et al. 2007; Mansour, Mohri, and Rostamizadeh 2009) and kernel embedding of distributions (Gretton et al. 2012).

**Theorem 1.** *Let $h\in\mathcal{H}$ be a hypothesis, $\epsilon_s(h)$ and $\epsilon_t(h)$ be the expected risks of the source and target respectively, then*

$$\epsilon_t(h)\leqslant\epsilon_s(h)+2D_k(\mathcal{X}_s,\mathcal{X}_t)+C, \tag{12}$$

*where $C$ is a constant for the complexity of hypothesis space and plus the risk of an ideal hypothesis for both domains.*

*Proof sketch:* The theoretical result from (Ben-David et al. 2007) shows that $\epsilon_t(h)\leqslant\epsilon_s(h)+d_{\mathcal{H}}(p,q)+C_0$, where $d_{\mathcal{H}}(p,q)$ is the $\mathcal{H}$-divergence between distributions $p$ and $q$,

$$d_{\mathcal{H}}(p,q)\triangleq 2\sup_{\eta\in\mathcal{H}}\left|\Pr_{\mathbf{x}^s\sim p}\left[\eta(\mathbf{x}^s)=1\right]-\Pr_{\mathbf{x}^t\sim q}\left[\eta(\mathbf{x}^t)=1\right]\right|. \tag{13}$$

The $\mathcal{H}$-divergence relies on the capacity of the hypothesis space $\mathcal{H}$ to distinguish distributions $p$ from $q$, and $\eta\in\mathcal{H}$ can be viewed as a *two-sample* classifier. By choosing $\eta$ as a one-class SVM classifier (Huang et al. 2006), $d_{\mathcal{H}}(p,q)$ can be bounded by its empirical estimate and thus by MMD (5),

$$\begin{aligned}d_{\mathcal{H}}(p,q)&\leqslant\hat{d}_{\mathcal{H}}(\mathcal{X}_s,\mathcal{X}_t)+C_1\\ &\leqslant 2\left(1-\inf_{\eta\in\mathcal{H}}\left[\sum_{i=1}^{n}\frac{\ell_{+1}[\eta(\mathbf{x}_i)]}{n}+\sum_{j=1}^{n'}\frac{\ell_{-1}\left[\eta(\mathbf{x}'_j)\right]}{n'}\right]\right)+C_1\\ &=2\left(1+D_k(\mathcal{X}_s,\mathcal{X}_t)\right)+C_1,\end{aligned} \tag{14}$$

where $\ell(\cdot)$ is the loss function for one-class SVM classifier $\eta$, and $\ell_{+1}[\eta]\triangleq\max(0,1-\eta), \ell_{-1}[\eta]\triangleq\max(0,1+\eta)$. DMM (7) decreases the target risk bound by (1) minimizing MMD w.r.t. feature transformation $\mathbf{A}$, which learns the

[1] http://www.manopt.org/

Table 2: Accuracy (%) on 12 transfer tasks of *Office10-Caltech10* with $\text{DeCAF}_6$ features.

| Dataset | SVM | KMM | TCA | TJM | LSSA | $\text{DMM}_\text{A}$ | $\text{DMM}_2$ | DMM |
|---|---|---|---|---|---|---|---|---|
| C→A | 91.6 | 91.5 | 90.3 | 91.4 | 91.6 | 92.2 | **92.7** | 92.4 |
| C→W | 80.7 | 81.0 | 83.7 | 84.7 | 85.2 | 83.7 | 85.4 | **87.5** |
| C→D | 86.0 | 85.4 | 88.5 | **91.7** | 90.4 | 88.5 | 91.1 | 90.4 |
| A→C | 82.2 | 82.2 | 82.0 | 82.1 | 82.5 | **85.1** | 84.8 | 84.8 |
| A→W | 71.9 | 72.2 | 75.6 | 76.3 | 79.4 | 79.7 | 84.1 | **84.7** |
| A→D | 80.9 | 82.2 | 85.4 | 81.5 | 86.6 | 89.8 | 89.8 | **92.4** |
| W→C | 67.9 | 67.3 | 68.5 | 70.2 | 73.2 | 78.4 | 78.4 | **81.7** |
| W→A | 73.4 | 74.4 | 74.5 | 79.5 | 81.4 | 85.7 | **89.7** | 86.5 |
| W→D | **100.0** | **100.0** | 99.4 | 99.4 | 99.4 | 98.7 | 99.4 | 98.7 |
| D→C | 72.8 | 72.0 | 76.9 | 77.1 | 80.4 | 80.9 | 82.3 | **83.3** |
| D→A | 78.7 | 79.6 | 82.7 | 83.9 | 83.4 | 88.3 | 90.3 | **90.7** |
| D→W | 98.3 | 98.3 | 98.0 | 98.3 | 98.0 | 98.0 | 98.0 | **99.3** |
| Average | 82.0 | 82.2 | 83.8 | 84.7 | 85.5 | 87.4 | 88.8 | **89.4** |

domain-invariant features; (2) minimizing MMD w.r.t. instance weights $\boldsymbol{\alpha}$, which learns the one-class SVM classifier parametrized by $\boldsymbol{\alpha}$ for optimal two-sample discrimination. As instance weights $\boldsymbol{\alpha}$ are the parameters of the one-class SVM classifier $\eta$, we use SVM-like constraints for $\boldsymbol{\alpha}$ in (7).

## Experiments

We perform extensive experiments to evaluate DMM against state of the art methods on standard domain adaptation benchmarks including both image and text datasets. Codes, datasets and configurations will be made available online.

### Setup

**Datasets** **Office-31** (Saenko et al. 2010) is the standard benchmark for domain adaptation, which consists of 4,652 images within 31 categories collected from three distinct domains: *Amazon* (**A**), which are images downloaded from amazon.com, *Webcam* (**W**) and *DSLR* (**D**), which are images taken by web camera and digital SLR camera under different environmental and photography variations. **Caltech-256** (Griffin, Holub, and Perona 2007) is a standard database for object recognition with 30,607 images and 256 classes.

In experiments, we adopt **Office-10 + Caltech-10** (Fernando et al. 2013; Long et al. 2013), which consists of the 10 common categories shared by the Office-31 and Caltech-256 datasets and is widely adopted in transfer learning methods (Long et al. 2013; 2014). From these two datasets, we can construct 12 transfer tasks for empirical evaluation: **A** → **C**, **W** → **C**, **D** → **C**, **C** → **A**, **C** → **W**, **C** → **D**, **A** → **W**, **D** → **W**, **W** → **D**, **A** → **D**, **D** → **A**, and **W** → **A**. The dataset is represented with the DeCAF features (Donahue et al. 2014), which are the 4,096-dimensional activations of the FC6-layer ($\text{DeCAF}_6$) or FC7-layer ($\text{DeCAF}_7$) extracted by the deep convolutional neural network (CNN) (Krizhevsky, Sutskever, and Hinton 2012). We use a common practice to normalize the input features by both z-score and $\ell_2$-norm.

**Reuters-21578** is a text dataset of Reuters news articles. The three top categories are *orgs* (**O**), *people* (**P**) and *place* (**Q**), each of which containing of many subcategories, making domain adaptation more difficult. We generate 6 transfer tasks **O** → **P**, **P** → **O**, **O** → **Q**, **Q** → **O**, **P** → **Q**, **Q** → **P**. For fair comparison, we directly adopt the preprocessed version of Reuters-21578 with TF-IDF features (Long et al. 2015b).

**Comparison Methods** We evaluate DMM against state of the art domain adaptation methods and the variants of DMM.

- **SVM** is used as base classifier for adaptation methods.
- **KMM** (Huang et al. 2006) is a seminal instance-based method that reweighs the importance of source instances to minimize the MMD between the source and target.
- **TCA** (Pan et al. 2011) is a seminal feature-based method that learns kernel-PCA components to minimize MMD.
- **TJM** (Long et al. 2014) jointly performs feature matching and instance reweighting through $\ell_{2,1}$-structural regularized Kernel PCA, which is the most relevant baseline.
- **LSSA** (Aljundi et al. 2015) selects landmarks to reduce the discrepancy between domains and then use kernel subspace alignment to perform feature-based adaptation.
- $\textbf{DMM}_\textbf{A}$ is the variant of DMM that only performs feature matching without instance reweighting, i.e. set $\boldsymbol{\alpha} = \mathbf{1}$.
- $\textbf{DMM}_2$ is the variant of DMM that performs distribution matching and weighted SVM (2) in a two-step procedure.

We adopt the same evaluation protocol for all comparison methods (Long et al. 2014; Aljundi et al. 2015). We use all source examples with labels and all target examples without labels for training, and report the average classification accuracy. For all comparison methods, we select their optimal hyper-parameters by cross-validation on labeled source data as (Pan et al. 2011). We give parameter sensitivity analysis for DMM, which will validate that DMM can achieve stable performance for a wide range of hyper-parameter settings.

### Results

**Image Classification** The classification accuracy of all comparison methods on the 12 transfer tasks of Office-10 + Caltech-10 using $\text{DeCAF}_6$ and $\text{DeCAF}_7$ features are shown in Tables 2 and 3, respectively. DMM significantly outperforms the five baseline methods and its two variants on most of the transfer tasks, setting a new state of the art record on this dataset. More specifically, DMM achieves the following

Table 3: Accuracy (%) on 12 transfer tasks of *Office10-Caltech10* with $\text{DeCAF}_7$ features.

| Dataset | SVM | KMM | TCA | TJM | LSSA | $\text{DMM}_\text{A}$ | $\text{DMM}_2$ | DMM |
|---|---|---|---|---|---|---|---|---|
| C→A | 92.0 | 91.0 | 90.9 | 91.5 | 91.9 | **92.7** | 91.9 | 92.6 |
| C→W | 84.4 | 81.0 | 85.8 | 88.8 | 88.8 | 88.8 | 88.8 | **90.5** |
| C→D | 86.6 | 83.4 | 87.3 | 91.1 | 90.4 | 90.4 | 91.1 | **91.7** |
| A→C | 82.4 | 82.5 | 80.3 | 80.8 | 82.4 | **84.3** | 84.0 | 83.3 |
| A→W | 84.1 | 84.4 | 82.4 | 83.4 | 86.4 | 91.5 | 86.4 | **92.2** |
| A→D | 86.6 | 86.6 | 81.5 | 88.5 | 86.4 | 91.1 | 91.7 | **93.0** |
| W→C | 73.0 | 73.2 | 78.6 | 78.7 | 81.6 | 85.5 | 85.1 | **85.8** |
| W→A | 79.4 | 81.4 | 85.6 | 87.2 | 88.4 | 90.8 | 90.7 | **92.5** |
| W→D | 99.4 | 99.4 | 98.7 | **100.0** | 99.4 | 89.8 | 98.7 | **100.0** |
| D→C | 76.0 | 78.3 | 80.5 | 80.4 | 82.5 | **85.0** | 84.1 | 84.3 |
| D→A | 83.1 | 86.7 | 87.2 | 86.8 | 86.7 | 91.4 | 92.2 | **93.2** |
| D→W | 96.9 | 98.3 | 97.6 | 97.3 | 97.3 | 91.9 | 99.0 | **99.7** |
| Average | 85.3 | 85.5 | 86.4 | 87.9 | 88.5 | 89.4 | 90.3 | **91.6** |

Table 4: Classification accuracy (%) on 6 transfer tasks of the *Reuters-21578* dataset.

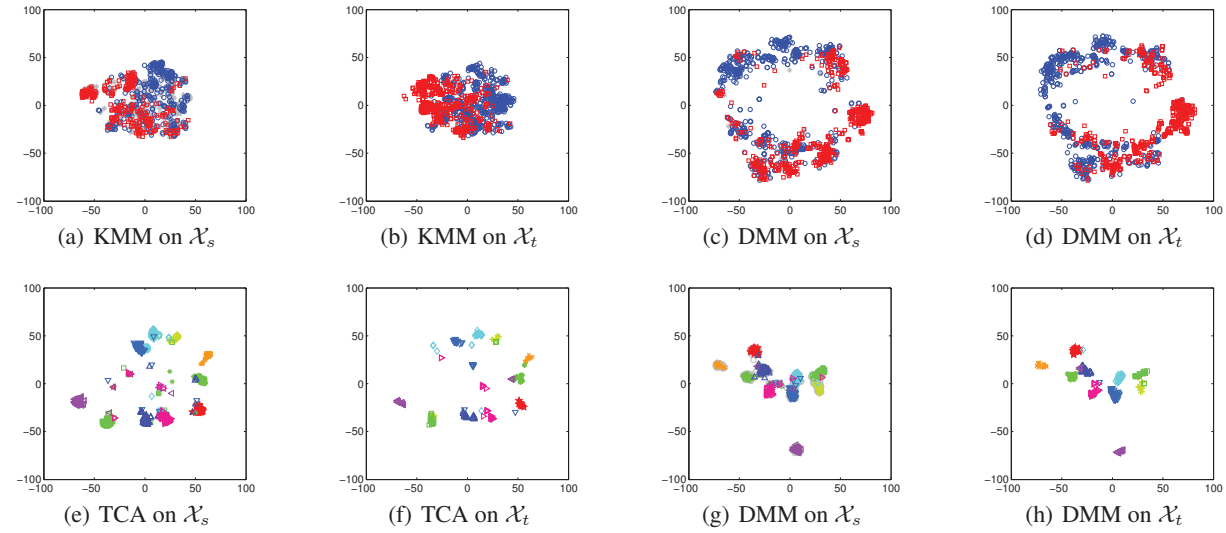
| Dataset | SVM | KMM | TCA | TJM | LSSA | $\text{DMM}_\text{A}$ | $\text{DMM}_2$ | DMM |
|---|---|---|---|---|---|---|---|---|
| O → P | 77.4 | 77.8 | 78.0 | **81.8** | 80.4 | 79.6 | 80.5 | 80.9 |
| O → Q | 75.7 | 69.5 | 71.8 | 73.8 | 75.3 | **77.8** | 77.3 | 77.6 |
| P → Q | 68.5 | 61.6 | 61.9 | 68.3 | 68.5 | 68.2 | 69.4 | **69.5** |
| P → O | 69.0 | 77.9 | 82.5 | 83.7 | 83.5 | 83.8 | 83.1 | **85.0** |
| Q → O | 61.6 | 67.9 | 70.8 | 75.0 | 76.2 | 79.5 | 77.3 | **79.9** |
| Q → P | 58.3 | 59.5 | 64.5 | 66.4 | 67.3 | **68.0** | 65.5 | 67.8 |
| Average | 68.4 | 69.0 | 71.6 | 74.8 | 75.2 | 76.1 | 75.5 | **76.8** |

performance gains compared against the best baselines: (1) **3.9%** on the 12 transfer tasks with $\text{DeCAF}_6$ features, and (2) **3.1%** on the 12 transfer tasks with $\text{DeCAF}_7$ features. Although DMM cannot perform the best on all tasks, it is desirable that (1) if DMM performs the best, then it usually outperforms the best baseline by a large margin; (2) otherwise, it performs only slightly worse than the best baseline. This verifies that DMM is more robust to both feature shift and instance bias for domain adaptation.

We can make more observations. (1) Domain adaptation methods generally outperform SVM, showing that reducing the distribution discrepancy is key to domain adaptation. (2) Feature-based adaptation methods TCA and $\text{DMM}_\text{A}$ perform much better than instance-based adaptation method KMM. This implies that reweighting source instances can remove the domain shift, but at the expense of larger estimation variance, as many source labeled examples are downweighed and are no longer effective for training the source classifier. (3) $\text{DMM}_\text{A}$ further outperforms TCA, validating the efficacy of minimizing the distribution discrepancy in the infinite-dimension reproducing kernel Hilbert space (DMM) instead of the dimension-reduced kernel PCA space (TCA). (4) By dual distribution matching of both features and instances, TJM outperforms TCA, while DMM performs the best in most cases. Only feature matching is not good enough for domain adaptation when the domain difference is substantially large, as there may be some source instances that are irrelevant to the target instances even using invariant features. TJM and DMM address this limitation by reweighting the source instances according to their relevance to the target instances in the new invariant feature space.

While TJM and LSSA perform distribution matching of features and instances, the superiority of DMM over TJM and LSSA are two important aspects. (1) DMM corrects the domain mismatch by reweighting the source instances in the structural risk minimization framework, where the instance importance is directly related to generalization error, as studied in our theoretical analysis. DMM further performs feature matching to guarantee more source instances are effective for classifying the target data. In TJM and LSSA, the instance reweighting is neither related to the empirical risk nor to the instance importance. (2) DMM jointly learns the transfer classifier and the transferable knowledge (invariant feature representations and unbiased instance weights) in an end-to-end learning paradigm, while TJM and LSSA require a two-step procedure where the first step learns the transferable knowledge and the second step learns the transfer classifier. This is suboptimal as the learned transferable knowledge may not be consistent with the discriminative structure. This is evidenced by the superiority of DMM over $\text{DMM}_2$.

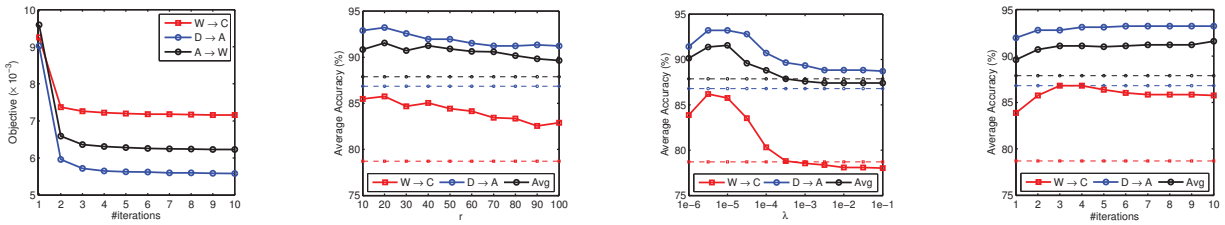
**Text Categorization** The average classification accuracy of the 6 transfer tasks of Reuters-21578 are reported in Table 4. The overall accuracy of DMM is **76.8%**, and the accuracy boost over the best baseline LSSA is **1.6%**. Both TJM and DMM significantly outperform the other baselines, which validates the effectiveness of dual distribution matching. An interesting observation is that $\text{DMM}_2$ outperforms $\text{DMM}_\text{A}$, implying that learning a joint transfer SVM is more important than dual distribution matching. DMM performs



(a) KMM on $\mathcal{X}_s$ (b) KMM on $\mathcal{X}_t$ (c) DMM on $\mathcal{X}_s$ (d) DMM on $\mathcal{X}_t$

(e) TCA on $\mathcal{X}_s$ (f) TCA on $\mathcal{X}_t$ (g) DMM on $\mathcal{X}_s$ (h) DMM on $\mathcal{X}_t$

Figure 2: t-SNE visualization of source and target data on tasks $\mathbf{A} \rightarrow \mathbf{W}$ and $\mathbf{Q} \rightarrow \mathbf{O}$: KMM (a)–(b), TCA (e)–(f), DMM (c)–(d) & (g)–(h). Color markers denote different classes, and gray circles denote source instances down-weighed by KMM or DMM.




(a) MMD (b) Dimension $r$ (c) Parameter $\lambda$ (d) Accuracy

Figure 3: Effectiveness analysis, parameter sensitivity and convergence analysis of the proposed DMM approach.

best, showing significance of optimizing all transfer criteria.

## Discussion

**Feature Visualization:** We visualize in Figures 2(a)–2(b), Figures 2(e)–2(f), and Figures 2(c)–2(d), 2(g)–2(h) the t-SNE embeddings (Donahue et al. 2014) of images on transfer tasks $\mathbf{A} \rightarrow \mathbf{W}$ and $\mathbf{Q} \rightarrow \mathbf{O}$ with features of KMM, TCA, and DMM, respectively. We make interesting observations. (1) KMM does not learn the invariant features, thus the discrepancy between source and target are still large. (2) TCA does not learn the unbiased weights of source instances, thus the source instances that are dissimilar to the target instances will not be down-weighed, leading to large domain bias. These observations explain the inferior performance of KMM and TCA, and highlight the superiority of DMM.

**Distribution Discrepancy:** We show the MMD of SVM, KMM, TCA, and DMM on transfer task $\mathbf{A} \rightarrow \mathbf{W}$ in Figure 3(a). Smaller distribution discrepancy lower generalization error. (1) Without performing distribution matching, the distribution discrepancy of SVM in original feature space is largest. (2) KMM and TCA explicitly reduce the distribution difference, thus they can outperform SVM. (3) DMM performs dual distribution matching of features and instances, hence it can maximally reduce the distribution discrepancy.

**Parameter Sensitivity:** We check sensitivity of subspace dimension $r$ and penalty parameter $\lambda$. Figures 3(b) and 3(c) show DMM outperforms baselines for wide ranges of parameters $r \in [10, 70]$, $\lambda \in [10^{-6}, 10^{-4}]$. The convergence of DMM on $\mathbf{W} \rightarrow \mathbf{C}$, $\mathbf{D} \rightarrow \mathbf{A}$ in Figure 3(d) shows the accuracy increases with iterations and converges in 10 iterations.

## Conclusion

We proposed a new Distribution Matching Machine (DMM) for domain adaptation. Using the structural risk minimization principle, DMM learns a transfer learner by extracting invariant feature representations and estimating unbiased instance weights that jointly minimize the cross-domain distribution discrepancy. Extensive experiments show that DMM significantly outperforms state of the art adaptation methods.

## Acknowledgments

This work was supported by the National Key Research and Development Program of China (2016YFB1000701),

National Natural Science Foundation of China (61502265, 61325008, 61772299, 61672313) and TNList Fund.

## References


Aljundi, R.; Emonet, R.; Muselet, D.; and Sebban, M. 2015. Landmarks-based kernelized subspace alignment for unsupervised domain adaptation. In *CVPR*.

Argyriou, A., and Evgeniou, T. 2006. Multi-task feature learning. In *NIPS*.

Ben-David, S.; Blitzer, J.; Crammer, K.; and Pereira, F. 2007. Analysis of representations for domain adaptation. In *NIPS*.

Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan, J. W. 2010. A theory of learning from different domains. *MLJ* 79(1-2):151–175.

Bruzzone, L., and Marconcini, M. 2010. Domain adaptation problems: A dasvm classification technique and a circular validation strategy. *TPAMI* 32(5):770–787.

Chen, M.; Weinberger, K. Q.; and Blitzer, J. C. 2011. Co-training for domain adaptation. In *NIPS*.

Chu, W.-S.; De la Torre, F.; and Cohn, J. F. 2013. Selective transfer machine for personalized facial action unit detection. In *CVPR*. IEEE.

Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; and Darrell, T. 2014. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*. ACM.

Duan, L.; Tsang, I. W.; and Xu, D. 2012. Domain transfer multiple kernel learning. *TPAMI* 34(3):465–479.

Duan, L.; Xu, D.; and Tsang, I. W. 2012. Domain adaptation from multiple sources: A domain-dependent regularization approach. *TNNLS* 23(3):504–518.

Fernando, B.; Habrard, A.; Sebban, M.; and Tuytelaars, T. 2013. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*.

Ganin, Y., and Lempitsky, V. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*.

Glorot, X.; Bordes, A.; and Bengio, Y. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*.

Gretton, A.; Borgwardt, K.; Rasch, M.; Schölkopf, B.; and Smola, A. 2012. A kernel two-sample test. *JMLR* 13:723–773.

Griffin, G.; Holub, A.; and Perona, P. 2007. Caltech-256 object category dataset. Technical report, California Institute of Technology.

Huang, J.; Smola, A. J.; Gretton, A.; Borgwardt, K. M.; and Scholkopf, B. 2006. Correcting sample selection bias by unlabeled data. In *NIPS*.

Jhuo, I.-H.; Liu, D.; Lee, D.-T.; and Chang, S.-F. 2012. Robust visual domain adaptation with low-rank reconstruction. In *CVPR*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*.

Long, M.; Wang, J.; Ding, G.; Sun, J.; and Yu, P. S. 2013. Transfer feature learning with joint distribution adaptation. In *ICCV*. IEEE.

Long, M.; Wang, J.; Ding, G.; Sun, J.; and Yu, P. S. 2014. Transfer joint matching for unsupervised domain adaptation. In *CVPR*. IEEE.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. I. 2015a. Learning transferable features with deep adaptation networks. In *ICML*. ACM.

Long, M.; Wang, J.; Sun, J.; and Yu, P. S. 2015b. Domain invariant transfer kernel learning. *TKDE* 27(6).

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

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2017. Deep transfer learning with joint adaptation networks. In *ICML*.

Mansour, Y.; Mohri, M.; and Rostamizadeh, A. 2009. Domain adaptation: Learning bounds and algorithms. In *COLT*.

Masaeli, M.; Fung, G.; and Dy, J. G. 2010. From transformation-based dimensionality reduction to feature selection. In *ICML*.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *TKDE* 22:1345–1359.

Pan, S. J.; Tsang, I. W.; Kwok, J. T.; and Yang, Q. 2011. Domain adaptation via transfer component analysis. *TNNLS* 22(2):199–210.

Pan, S. J.; Kwok, J. T.; and Yang, Q. 2008. Transfer learning via dimensionality reduction. In *AAAI*.

Qiu, Q.; Patel, V. M.; Turaga, P.; and Chellappa, R. 2012. Domain adaptive dictionary learning. In *ECCV*.

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

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. In *ECCV*.

Sun, B.; Feng, J.; and Saenko, K. 2016. Return of frustratingly easy domain adaptation. In *AAAI*.

Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2015. Simultaneous deep transfer across domains and tasks. In *ICCV*.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *CVPR*.

Vapnik, V. 1998. *Statistical Learning Theory*. John Wiley.

Yu, Y., and Szepesvári, C. 2012. Analysis of kernel mean matching under covariate shift. In *ICML*. ACM.

Zhang, K.; Schölkopf, B.; Muandet, K.; and Wang, Z. 2013. Domain adaptation under target and conditional shift. In *ICML*. ACM.