Representation Subspace Distance for Domain Adaptation Regression

Xinyang Chen [1] **Sinan Wang** [1] **Jianmin Wang** [1] **Mingsheng Long** [1]

Abstract

Regression, as a counterpart to classification, is a major paradigm with a wide range of applications. Domain adaptation regression extends it by generalizing a regressor from a labeled source domain to an unlabeled target domain. Existing domain adaptation regression methods have achieved positive results limited only to the shallow regime. A question arises: Why learning invariant representations in the deep regime less pronounced? A key finding of this paper is that classification is robust to feature scaling but regression is not, and aligning the distributions of deep representations will alter feature scale and impede domain adaptation regression. Based on this finding, we propose to close the domain gap through *orthogonal bases* of the representation spaces, which are free from feature scaling. Inspired by Riemannian geometry of Grassmann manifold, we define a geometrical distance over representation subspaces and learn deep transferable representations by minimizing it. To avoid breaking the geometrical properties of deep representations, we further introduce the bases mismatch penalization to match the ordering of orthogonal bases across representation subspaces. Our method is evaluated on three domain adaptation regression benchmarks, two of which are constructed in this paper. Our method outperforms the state-of-the-art methods significantly, forming early positive results in the deep regime.

1. Introduction

The regression paradigm is proposed to formalize the tasks of predicting continuous values from each instance. As one of the main paradigms in machine learning, regression tasks attract parallel attention as classification tasks. Problems attributed to regression tasks arise widely in real applications, such as object localization, image registration, human pose estimation, to name a few (Lathuilière et al., 2019).

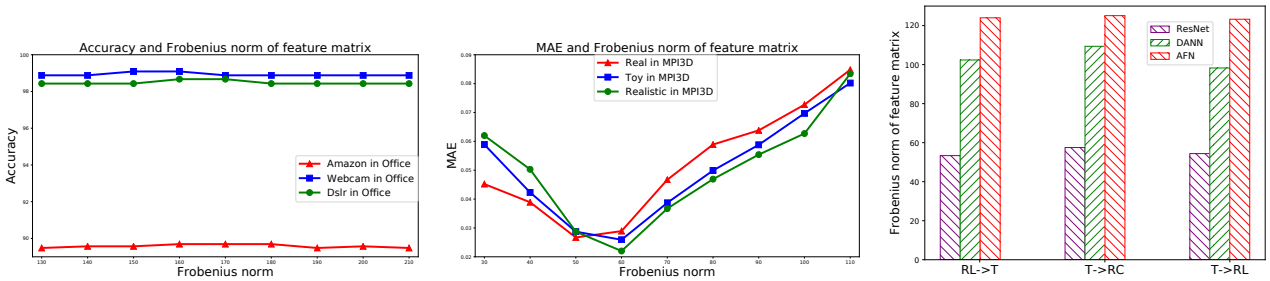
Deep learning has made remarkable changes in diverse regression applications across many fields. Nonetheless, training high-quality deep models relies on large-scale labeled datasets. And in many real-world regression applications, precisely annotating abundant training instances is time-consuming and laborious. A solution to this problem is leveraging off-the-shelf labeled data from a relevant domain and applying domain adaptation approaches to overcome the domain shift or dataset bias (Shimodaira et al., 2009). There are many pioneers hammering at tackling domain adaptation regression (**DAR**) problem at the theoretical or algorithmic level. Mansour et al. (2009) and Cortes & Mohri (2011) conducted extensive theoretical analyses of the DAR problem. Meanwhile, a series of algorithms are proposed for the DAR problem. These methods focus on importance weighting (Geng et al., 2007; Guo et al., 2008; Yamada et al., 2014) or learning invariant representations (Cao et al., 2010; Pan et al., 2011) in the shallow regime, achieving impressive improvements and bringing inspiration for future works. However, most methods rely on labeled target data. And there are rare DAR approaches in the deep regime.

Recent studies reveal that deep networks are able to learn representations generically useful across a variety of tasks (Yosinski et al., 2014). Inspired by this, deep domain adaptation methods have achieved remarkable advances in domain adaptation classification (**DAC**) problems. In light of these developments in DAC, a natural question arises: Why learning invariant representations in the deep regime for DAR less pronounced? An essential question is: what is the vital difference between classification and regression during the representation learning process?

Intuitively, their essential difference lies in the *loss function*. A commonly used loss function in regression is squared loss (L2), while in classification it is cross-entropy loss (CE) with softmax activation function. Softmax allows the activation values of different categories to compete with each other. Introducing this competition mechanism can make classifiers quickly adapt to change in feature scales. However, in regression tasks, regressors may not have such adaptability. To further explore this property, we conduct an exploratory study to verify the effect of feature scaling on

[1]School of Software, BNRist, Tsinghua University.
Xinyang Chen <chenxinyang95@gmail.com>. Correspondence to: Mingsheng Long <mingsheng@tsinghua.edu.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.



(a) Classification (b) Regression (c) Frobenius norm

Figure 1. (a) Accuracy and feature scale in Office-31. (b) MAE and feature scale in MPI3D. (c) Change of feature scale by two methods.

classifiers or regressors.

We first test the impact of the Frobenius norm of feature matrix on task performances in both classification and regression settings. For classification tasks, we use the most popular domain adaptation dataset: **Office-31** (Saenko et al., 2010), and for regression tasks, we use **MPI3D** (Gondal et al., 2019). First, we test the robustness of experimental performances to the change of feature scale. Specifically, we use L2 regularization to change the Frobenius norm of feature matrix. Results in classification are shown in Figure 1(a) and those in regression are shown in Figure 1(b). An important observation is that *classification performances are more robust to feature scaling, while regression performances are not*. This means that performance degradation is a risk when feature scale in terms of Frobenius norm is affected by regularization terms other than the supervised learning loss in the representation learning process.

Most transferable representation learning methods for domain adaptation classification (DAC) are able to reduce the distribution discrepancy based on instance representations. Two popular methods, Domain Adversarial Neural Network (**DANN**) (Ganin et al., 2016) and Adaptive Feature Norm (**AFN**) (Xu et al., 2019) are used to test the change of Frobenius norm of feature matrix on three transfer regression tasks from the source domain. Results are shown in Figure 1(c). An important observation is that these domain adaptation classification methods significantly change the feature scale, leading to the risk of performance degradation in regression. More experimental details are included in Appendix.

Based on these observations, we investigate how to learn transferable representation for domain adaptation regression (DAR) from a new perspective of preserving the feature scale. Different from the previous instance-based distance reduction approaches, we try to solve this problem by exploring the Riemannian geometry of the Grassmann manifold. In the transferable representation learning process, each instance representation vector has its direction and magnitude. Note that matching distributions using instance representation has the risk of changing the feature scale. Importantly, each feature matrix is a point in the Grassmannian, whose bases are unit vectors. *Matching bases can reduce distance between subspaces but will not change feature scale*.

In light of this, we explore the principal angles, a subspace similarity in the Riemannian geometry of Grassmann manifold. Based on that, we define a new geometrical distance, called Representation Subspace Distance (**RSD**). The satisfaction to all axioms for a general metric is proved. Thus RSD can be exploited to enable transferable representation learning without changing feature scale. Meanwhile, the definition of RSD has a disadvantage under the scenarios of deep representation learning: it does not take the ordering of the importance of each orthonormal basis into consideration. Our observation is that equally important bases in different subspaces tend to have similar semantic information. In other words, it is unreasonable to match an important basis in one subspace with an unimportant basis in another subspace. To this end, we propose Bases Mismatch Penalization (**BMP**) to constrain similarly ranked bases in each subspace to match together when calculating the new distance. By minimizing both RSD and BMP, transferable representation learning can be enabled to improve DAR performance.

This paper has the following three contributions:

- We identify two key findings: regression performances are not robust to feature scaling, and transferable representation learning in domain adaptation classification (DAC) is at the risk of changing feature scale.
- We propose to match orthogonal bases to close domain shift without changing feature scale. Specifically, we propose Representation Subspace Distance (RSD), a new geometrical distance satisfying all key axioms for a general metric. We propose a novel regularizer, Bases Mismatch Penalization (BMP), to constrain similarly ranked bases in each subspace to match together when calculating the distance across subspaces. These two loss functions facilitate transferable representation learning to boost domain adaptation regression (DAR).
- We construct two new benchmarks for deep DAR. Our method outperforms state-of-the-arts by huge margins on both two benchmarks and a pose estimation dataset.

2. Related Work

Domain Adaptation Regression (DAR). Mansour et al. (2009) discuss domain adaptation theory in both classification and regression settings, and Cortes & Mohri (2011) conduct an extensive theoretical analysis for DAR. Meanwhile, a series of algorithms are proposed for the DAR problem. These methods focus on importance weighting (Geng et al., 2007; Guo et al., 2008; Yamada et al., 2014; de Mathelin et al., 2020) or learning invariant representations (Cao et al., 2010; Pan et al., 2011; Courty et al., 2017; Nikzad-Langerodi et al., 2020) in the shallow regime. Nevertheless, most of them need to use a small number of target domain labels to boost performances. Besides, Teshima et al. (2020) solve few-shot supervised domain adaptation problem by causal mechanism transfer. Under the framework of deep representation learning, Singh & Chakraborty (2020) leverage unlabeled data and adopt the widely used Maximum Mean Discrepancy (MMD) metric as well as a semi-supervised loss to strengthen the smoothness of the prediction function in the deep regime. However, it depends on labeled target data, and the semi-supervised loss contributes a considerable part of the improvement. Without using labeled target data, there is still a lack of explicit solutions to the core problem of DAR when dealing with complex regression problems under the framework of deep representation learning.

Domain Adaptation Classification (DAC). Domain adaptation, as a basic problem to learn from non-iid data, addresses the problem of learning a model that reduces the distribution discrepancy between training and testing distributions (Shimodaira et al., 2009). Early domain adaptation methods in the shallow regime learn invariant representations across domains (Pan et al., 2011; Gong et al., 2012) or reweigh source instances based on their correlation to the target domain (Huang et al., 2007; Gong et al., 2013). It is worth noting that Gopalan et al. (2011), Gong et al. (2012) and Zheng et al. (2012) propose geometrical appoarches to learn invariant representations. They all use the geodesic distance as the subspace distance and achieved good results, but their flaw is that the geodesic distance cannot be minimized because the subspaces are fixed. Recent studies reveal that deep networks are able to learn representations generically useful across a variety of tasks (Yosinski et al., 2014). Inspired by that, recent DAC methods in the deep regime simultaneously explore two approaches for learning transferable representations across domains. One approach is moment matching, which reduces the distribution discrepancy by matching statistics from two different distributions (Long et al., 2015; Li et al., 2016; Long et al., 2017; Maria Carlucci et al., 2017). The other approach is adversarial learning, inspired by generative adversarial nets (Goodfellow et al., 2014). A minimax game is built to directly close the domain gap, which takes the supremum of a proper function over the hypothesis space as distribution discrepancy, while feature representations are learned to reduce the discrepancy simultaneously (Ganin & Lempitsky, 2015; Tzeng et al., 2015; Ganin et al., 2016; Tzeng et al., 2017; Luo et al., 2017; Long et al., 2018; Zhang et al., 2019; Peng et al., 2019). Remarkable performance gains are yielded by these transferable representation learning methods in DAC. And many of the mentioned DAC methods can be naturally extended to DAR problems, we will verify their effectiveness in experiments. As a conclusion, deep learning lays strong foundations to solve complex domain adaptation problems.

Inspired by (Chen et al., 2019), we use spectral analysis methods to explore this problem in depth. We find that regression performances are not robust to feature scaling. And deep domain adaptation approaches closing domain gap based on instance representation have the risk of changing feature scale. Thus we tackle this challenge from a new perspective and make use of subspace similarity, a kind of Riemannian geometry, to define a geometrical distance for learning transferable representations. Further, we preserve the geometrical structure of deep representations by a new regularizer. To our knowledge, this work sheds the first light on designing effective deep DAR algorithms.

3. Approach

In this paper, we study the unsupervised domain adaptation problem in the scenario of regression (DAR). During training, we are given n^s labeled examples from a source domain $\mathcal{D}^s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}$ and n^t unlabeled examples from a target domain $\mathcal{D}^t = \{(\mathbf{x}_i^t)\}$. Examples from different domains are sampled from different distributions P and Q respectively, and the *i.i.d.* (independent and identically distributed) assumption of standard learning is violated, *i.e.* $P \neq Q$.

3.1. Motivation

As we stated in Section 1, to lower the generalization error in the target regression tasks, learning transferable representations by closing the domain shift without changing feature scale is the key idea to make DAR work in the deep regime.

In common deep domain adaptation models, there exists a feature extractor denoted by G_f. Deep representation is learned by the feature extractor, formalized as $\mathbf{f}_i = G_f(\mathbf{x}_i)$. A superscript is adopted to distinguish the feature vector $\mathbf{f}_i^s$ from the source domain and that $\mathbf{f}_i^t$ from the target domain. During training, each batch feature matrix $\mathbf{F}^s = [\mathbf{f}_1^s \dots \mathbf{f}_b^s]$ is composed of a batch size b of feature vectors. The Frobenius norm of feature matrix $||\mathbf{F}^s||_F = \sqrt{\mathrm{Tr}((\mathbf{F}^s)^\mathsf{T}\mathbf{F}^s)} = \sqrt{\sum_{i=1}^{b} \sigma_i^s}$, where Tr is trace, σ_i^s is the i-th singular value. This means that the Frobenius norm of feature matrix is only

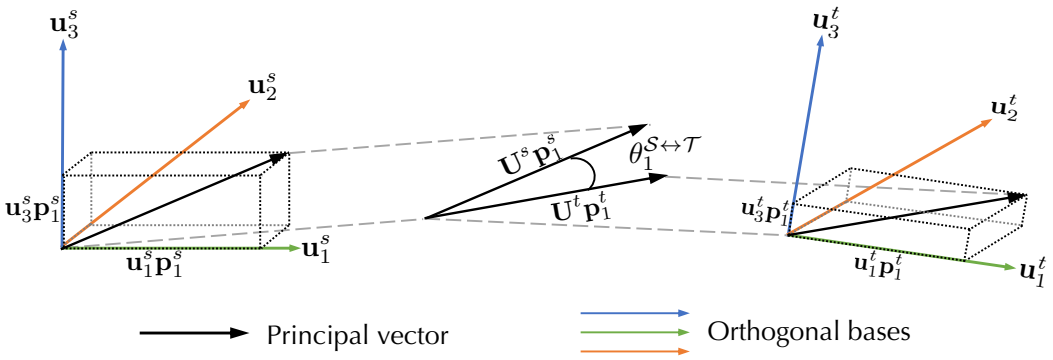


Figure 2. The calculation process of principal angles. Principal angle $\theta_1^{\mathcal{S}\leftrightarrow\mathcal{T}}$ is the angle between principal vector $\mathbf{U}^s\mathbf{p}_1^s$ and $\mathbf{U}^t\mathbf{p}_1^t$. Principal vector $\mathbf{U}^s\mathbf{p}_1^s$ is the *weighted sum* of orthogonal bases $\mathbf{U}^s$, where $\mathbf{p}_1^s$ is corresponding weight vector.

influenced by its singular values. Thus we can use a general geometrical tool, Singular Value Decomposition (**SVD**), to obtain the *orthonormal bases* $\mathbf{U}$ and *singular values* $\mathbf{\Sigma}$ of the feature matrices in both source and target domain:

$$\mathbf{F}^s = \mathbf{U}^s\mathbf{\Sigma}^s(\mathbf{V}^s)^\mathsf{T}, \quad \mathbf{F}^t = \mathbf{U}^t\mathbf{\Sigma}^t(\mathbf{V}^t)^\mathsf{T}. \tag{1}$$

Here, matrix $\mathbf{U}^s$ is composed of orthonormal bases in source representation subspace $\mathcal{S}$ and matrix $\mathbf{U}^t$ is composed of orthogornal bases in target representation subspace $\mathcal{T}$. Our aim is to close the domain gap *without using singular values*, thereby not changing the Frobenius norm of feature matrix. Inspired by Riemannian geometry of Grassmann manifold, we can only use orthogonal bases to achieve this goal.

3.2. Principal Angles

To measure the similarity of two b-dimensional subspaces $\mathcal{S}$ and $\mathcal{T}$ of $\mathbb{R}^n$ on the Grassmannian $\mathbf{Gr}(b,n)$, the basic concept is to calculate the similarity of orthonormal bases from two subspaces globally (Golub & Van Loan, 1996). A direct measurement is to use the principal angles, whose definition is as follows:

$$\begin{aligned}
\theta_1^{\mathcal{S}\leftrightarrow\mathcal{T}} &= \min_{\mathbf{u}_1^s\in\mathcal{S},\mathbf{u}_1^t\in\mathcal{T}} \arccos\left(\frac{(\mathbf{u}_1^s)^\mathsf{T}\mathbf{u}_1^t}{\|\mathbf{u}_1^s\|\,\|\mathbf{u}_1^t\|}\right),\\
\theta_2^{\mathcal{S}\leftrightarrow\mathcal{T}} &= \min_{\substack{\mathbf{u}_2^s\in\mathcal{S},\mathbf{u}_2^t\in\mathcal{T}\\ \mathbf{u}_2^s\perp\mathbf{u}_1^s,\mathbf{u}_2^t\perp\mathbf{u}_1^t}} \arccos\left(\frac{(\mathbf{u}_2^s)^\mathsf{T}\mathbf{u}_2^t}{\|\mathbf{u}_2^s\|\,\|\mathbf{u}_2^t\|}\right),\\
&\vdots\\
\theta_b^{\mathcal{S}\leftrightarrow\mathcal{T}} &= \min_{\substack{\mathbf{u}_b^s\in\mathcal{S},\mathbf{u}_b^t\in\mathcal{T}\\ \mathbf{u}_b^s\perp\mathbf{u}_1^s,\ldots,\mathbf{u}_{b-1}^s\\ \mathbf{u}_b^t\perp\mathbf{u}_1^t,\ldots,\mathbf{u}_{b-1}^t}} \arccos\left(\frac{(\mathbf{u}_b^s)^\mathsf{T}\mathbf{u}_b^t}{\|\mathbf{u}_b^s\|\,\|\mathbf{u}_b^t\|}\right),
\end{aligned} \tag{2}$$

where $\mathbf{U}^s = [\mathbf{u}_1^s,\ldots,\mathbf{u}_b^s]$ and $\mathbf{U}^t = [\mathbf{u}_1^t,\ldots,\mathbf{u}_b^t]$ are the orthonormal bases in the b-dimensional subspaces $\mathcal{S}$ and $\mathcal{T}$, respectively. When the principal angles $\mathbf{\Theta} = [\theta_1,\ldots,\theta_b]$ are all zero, the subspaces spanned by the two groups of orthogonal bases are exactly the same.

3.3. Representation Subspace Distance

It is straightforward to define a geometrical distance on the Grassmann manifold with the above similarity measurement. As an effective metric on the manifold, the new distance should satisfy all necessary axioms for a general metric.

Definition 1 (Representation Subspace Distance, RSD). *The Representation Subspace Distance (RSD) between two m-dimensional subspaces $\mathcal{S}$ and $\mathcal{T}$ is the sum of the sine values of all principal angles,*

$$\mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}}(\mathbf{U}^s,\mathbf{U}^t) = \left\|\sin\mathbf{\Theta}^{\mathcal{S}\leftrightarrow\mathcal{T}}\right\|_1 = \sum_{i=1}^{m}\sin\theta_i^{\mathcal{S}\leftrightarrow\mathcal{T}}. \tag{3}$$

Further, we prove that the new geometrical distance satisfies the three axioms for a general metric:

Theorem 1. *The Representation Subspace Distance (RSD) satisfies the three axioms for a general metric, to be specific, it satisfies the following conditions:*

(1) $\mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}} \geq 0$, *and* $\mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}} = 0$ *if and only if* $\mathcal{S} = \mathcal{T}$;

(2) $\mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}} = \mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}}$ *(symmetric);*

(3) $\mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}} \leq \mathrm{dis}_{\mathrm{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{A}} + \mathrm{dis}_{\mathrm{RSD}}^{\mathcal{T}\leftrightarrow\mathcal{A}}$ *(triangle inequality).*

Due to space limitation, we will defer proofs in Appendix. With the guarantee of these three properties, Representation Subspace Distance (RSD) can be used to measure the discrepancy between subspaces on the Grassmann manifold.

Golub & Van Loan (1996) present a neat way to calculate all principal angles using those orthonormal bases and SVD:

$$(\mathbf{U}^s)^\mathsf{T}\mathbf{U}^t = \mathbf{P}^s\big(\mathrm{diag}(\cos\mathbf{\Theta}^{\mathcal{S}\leftrightarrow\mathcal{T}})\big)(\mathbf{P}^t)^\mathsf{T}, \tag{4}$$

where $\cos\mathbf{\Theta}^{\mathcal{S}\leftrightarrow\mathcal{T}}$ is all the cosine values of principal angles

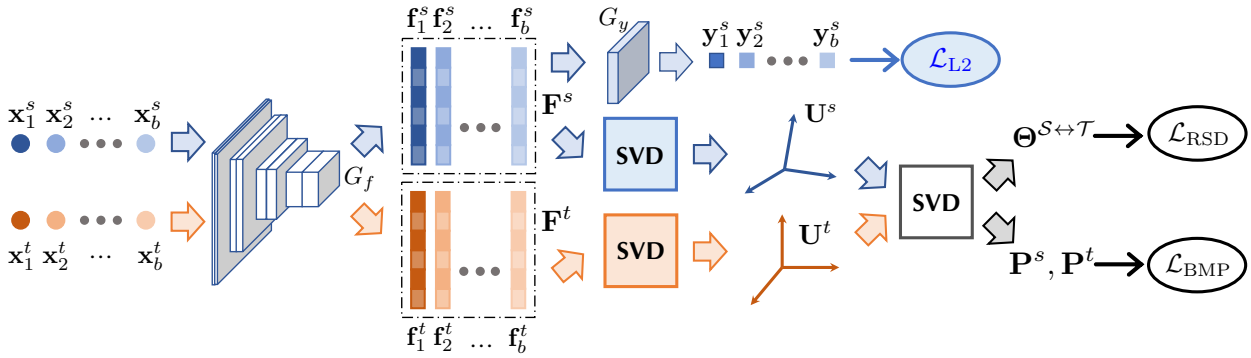


Figure 3. The architecture of our method. RSD is a novel distance to learn transferable representations for DAR problems. BMP is a novel regularizer to maintain geometrical structures of deep representations. They are end-to-end trainable by **differentiable SVD** in PyTorch.

between $\mathcal{S}$ and $\mathcal{T}$. With the principal angles, we can easily compute RSD between $\mathcal{S}$ and $\mathcal{T}$ using Equation (3).

3.4. Bases Mismatch Penalization

Directly minimizing RSD can reduce the distance between the source representation subspace $\mathcal{S}$ and the target representation subspace $\mathcal{T}$. However, another key property, *i.e.* the *ordering* of the orthogonal bases, should be considered to preserve the geometrical property of deep representations.

Foremost, we delve into the SVD process in Equation (4). For simplicity, we describe how the smallest principal angle $\theta_1^{\mathcal{S}\leftrightarrow\mathcal{T}}$ between two subspaces $\mathcal{S}$ and $\mathcal{T}$ is obtained when each subspace is composed of three orthogonal bases in Figure 2. The smallest principal angle $\theta_1^{\mathcal{S}\leftrightarrow\mathcal{T}}$ is the angle between principal vector $\mathbf{U}^s\mathbf{p}_1^s$ in $\mathcal{S}$ and principal vector $\mathbf{U}^t\mathbf{p}_1^t$ in $\mathcal{T}$. Principal vectors, which are used to calculate principal angles, are *weighted sums* of all orthogonal bases. The weight matrix is $\mathbf{P}^s$ in $\mathcal{S}$ and $\mathbf{P}^t$ in $\mathcal{T}$, which are easily obtained by SVD in Equation (4).

Based on the calculation process of principal angles in Figure 2, we can find the fact that in $\text{dis}_{\text{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}}(\mathbf{U}^s, \mathbf{U}^t)$, we do not consider the *ordering* of importance of each orthogonal basis (ordering of singular values in Equation (1)). An important observation is that most principal vectors are dominated by a specific orthogonal basis instead of more orthogonal bases. If the first principal vector $\mathbf{U}^s\mathbf{p}_1^s$ in $\mathcal{S}$ is dominated by a top-ranked orthogonal basis and the first principal vector $\mathbf{U}^t\mathbf{p}_1^t$ in $\mathcal{T}$ is dominated by a bottom-ranked orthogonal basis, then these two bases of different importance will be matched by minimizing $\text{dis}_{\text{RSD}}^{\mathcal{S}\leftrightarrow\mathcal{T}}(\mathbf{U}^s, \mathbf{U}^t)$. However, different orthogonal bases have different semantic meanings, and orthogonal bases with similar rankings in different domains are often more likely to represent similar semantics. It is unreasonable to match important components in one representation with unimportant components in another representation. To this end, we propose Bases Mismatch Penalization (**BMP**) to maintain this property during the representation learning process. For two principal vectors matched by Equation (4), an intuitive idea is that absolute values of their weight matrices $\mathbf{P}^s$ and $\mathbf{P}^t$ should be similar:

$$\text{reg}_{\text{BMP}}^{\mathcal{S}\leftrightarrow\mathcal{T}}(\mathbf{U}^s, \mathbf{U}^t) = \left\| |\mathbf{P}^s| - |\mathbf{P}^t| \right\|_F^2. \tag{5}$$

The reason why we use absolute values of $\mathbf{P}^s$ and $\mathbf{P}^t$ is that two subspaces formed by the bases in the opposite directions are the same one. By applying $\text{reg}_{\text{BMP}}^{\mathcal{S}\leftrightarrow\mathcal{T}}(\mathbf{U}^s, \mathbf{U}^t)$ throughout training, the orthogonal bases with similar rankings in their representation subspaces are more likely to be matched.

3.5. Transferable Representation Learning

Our DAR model is trained in an end-to-end way based on deep learning architectures. Both feature extractor G_f and regressor G_y are trained by minimizing the supervised loss:

$$\mathcal{L}_{\text{L2}}(G_f, G_y) = \mathbb{E}_{(\mathbf{x}_i^s, \mathbf{y}_i^s)\sim P^b} \mathsf{loss}(G_y(G_f(\mathbf{x}_i^s)), \mathbf{y}_i^s), \tag{6}$$

where $\mathsf{loss}(\cdot,\cdot)$ is the squared loss ($\mathsf{L2}$), and $(\mathbf{x}_i^s, \mathbf{y}_i^s) \sim P^b$ means sampling a batch of b instances from source domain. With this batch and another batch of b instances sampled from target domain, we define the RSD loss $\mathcal{L}_{\text{RSD}}$ based on the geometrical distance between source representation subspace $\mathcal{S}$ and target representation subspace $\mathcal{T}$, and the BMP loss $\mathcal{L}_{\text{BMP}}$ based on matching the ordering of orthogonal bases of $\mathcal{S}$ and $\mathcal{T}$. The defined loss functions are:

$$\begin{aligned} \mathcal{L}_{\text{RSD}}(G_f) &= \mathbb{E}_{\mathbf{x}_i^s\sim P^b, \mathbf{x}_i^t\sim Q^b} \left\| \sin \mathbf{\Theta}^{\mathcal{S}\leftrightarrow\mathcal{T}} \right\|_1, \\ \mathcal{L}_{\text{BMP}}(G_f) &= \mathbb{E}_{\mathbf{x}_i^s\sim P^b, \mathbf{x}_i^t\sim Q^b} \left\| |\mathbf{P}^s| - |\mathbf{P}^t| \right\|_F^2. \end{aligned} \tag{7}$$

The final goal is to learn transferable deep representations through minimizing both RSD and BMP between domains. DAR is achieved with the following optimization problem:

$$\min_{G_f, G_y} \mathcal{L}_{\text{L2}}(G_f, G_y) + \beta\mathcal{L}_{\text{RSD}}(G_f) + \gamma\mathcal{L}_{\text{BMP}}(G_f), \tag{8}$$

where $\beta, \gamma > 0$ are trade-off hyper-parameters. The overall architecture of the proposed approach is shown in Figure 3.

4. Experiments

We evaluate our method with several state-of-the-art domain adaptation methods on three benchmarks. First, we carefully observe the behavior of our model on dSprites, an easy 2D synthetic dataset. Further, MPI3D, a challenging *simulation-to-real* 3D dataset, is employed to testify the ability of all methods to learn transferable representations for regression tasks. Finally, the Biwi Kinect Head Pose dataset, a real-world head pose estimation benchmark, is utilized to examine the practicability of all methods. The code is available at `github.com/thuml/Domain-Adaptation-Regression`.

4.1. Datasets

dSprites[1] (Higgins et al., 2017) is a standard 2D synthetic dataset for deep representation learning. It is composed of three domains each with 737,280 images: *Color* (**C**), *Noisy* (**N**) and *Scream* (**S**). The example images are shown in Figure 4. In every image, there are five factors of variations, details illustrated in Table 1.

Table 1. Factors of variations in dSprites

Factor	Possible Values	Task
Shape	square, ellipse, heart	recognition
Scale	6 values in $[0.5, 1]$	regression
Orientation	40 values in $[0, 2\pi]$	regression
Position X	32 values in $[0, 1]$	regression
Position Y	32 values in $[0, 1]$	regression

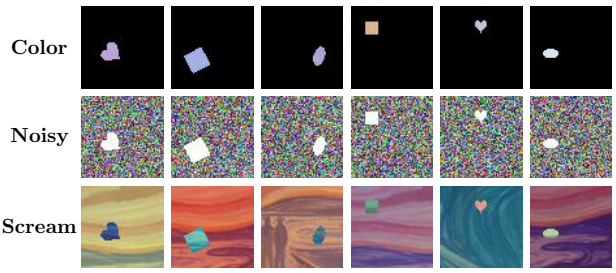


Figure 4. Examples of dSprites.

In dSprites, there are four factors that can be employed for regression tasks: *scale*, *orientation*, *position X and Y*. However, it is knotty to determine the value of orientation: (1) For a heart shape, possible values of orientation are 20 values in $[0, 2\pi]$; (2) For an ellipse shape, possible values of orientation are 20 values in $[0, \pi]$; (3) For a square shape, possible values of orientation are 20 values in $[0, \frac{1}{2}\pi]$. Consequently, the orientation regression task is excluded from consideration. We evaluate all methods on six transfer tasks: **C** $\rightarrow$ **N**, **C** $\rightarrow$ **S**, **N** $\rightarrow$ **C**, **N** $\rightarrow$ **S**, **S** $\rightarrow$ **C**, and **S** $\rightarrow$ **N**. And sum of MAE on three regression tasks (*scale*, *position X and Y*) is reported.

MPI3D[2] (Gondal et al., 2019) is a *simulation-to-real* dataset of 3D objects. It has three domains: *Toy* (**T**), *RealistiC* (**RC**) and *ReaL* (**RL**). Each domain contains 1,036,800 images, with mechanical platforms and example images shown in Figure 5. And every image has seven factors of variations, details shown in Table 2.

Table 2. Factors of variations in MPI3D

Factor	Possible Values	Task
Object Color	5 values	recognition
Object Shape	6 values	recognition
Object Size	2 values	recognition
Camera Height	3 values	recognition
Background Color	3 values	recognition
Horizontal Axis	40 values in $[0, 1]$	regression
Vertical Axis	40 values in $[0, 1]$	regression

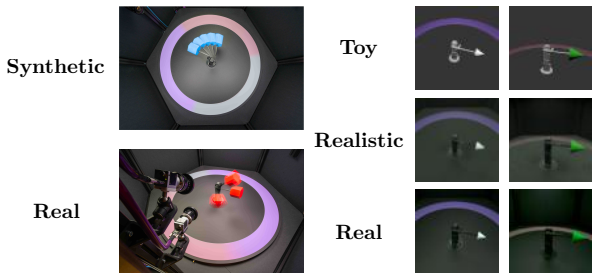


Figure 5. Mechanical platforms and examples of MPI3D.

In MPI3D, there are two factors that can be employed for regression tasks: a *rotation* about a vertical axis at the base and a *second rotation* about a horizontal axis. We evaluate all methods on six transfer tasks: **RL** $\rightarrow$ **RC**, **RL** $\rightarrow$ **T**, **RC** $\rightarrow$ **T**, **RC** $\rightarrow$ **RL**, **T** $\rightarrow$ **RL**, and **T** $\rightarrow$ **RC**. And sum of MAE on two regression tasks (*two rotations*) is reported.

Biwi Kinect (Fanelli et al., 2013) is a real-world dataset for head pose estimation. We divide the dataset into two domains according to gender: *Female* (**F**) (5874 images) and *Male* (**M**) (9804 images). Example images are shown in Figure 6. And every image has 3 factors of variations, details shown in Table 3.

In Biwi Kinect, there are three factors that can be employed for regression tasks: *pitch*, *yaw*, and *roll*. We evaluate all

[1] `https://github.com/deepmind/dsprites-dataset`

[2] `https://github.com/rr-learning/disentanglement_dataset`

methods on two transfer tasks: $\mathbf{F} \rightarrow \mathbf{M}$ and $\mathbf{M} \rightarrow \mathbf{F}$. And sum of MAE on three regression tasks (*pitch*, *yaw*, and *roll*) is reported.

Table 3. Factors of variations in Biwi Kinect

Factor	Possible Values	Task
Pitch	values in $[-92.044, 231.352]$	regression
Yaw	values in $[-87.7066, 246.684]$	regression
Roll	values in $[754.182, 1297.45]$	regression

Figure 6. Examples of Biwi Kinect.

4.2. Implementation Details

We use PyTorch[3] with Titan V to implement our methods and fine-tune ResNet-18 (He et al., 2016) pre-trained on ImageNet (Russakovsky et al., 2015). On dSprites, there are three regression tasks: *scale, position X and position Y*. On MPI3D, there are two tasks: *a rotation about a vertical axis at the base and a second rotation about a horizontal axis*. And on Biwi Kinect, there are three tasks: *pitch, yaw and roll*. We treat all tasks equally on each dataset. Labels are all normalized to $[0, 1]$ to eliminate the effects of diverse scales in regression values, where the activation of the regressor is Sigmoid. Tasks on one dataset share the same feature extractor G_f and the same learning rate, respectively. For the baseline methods for domain adaptation regression in the shallow regime, we use the pretrained ResNet-18 to extract representations as input to those methods.

Following the standard protocols for unsupervised domain adaptation (Ganin et al., 2016), all labeled source samples and unlabeled target samples participate in training. All images are resized to 224×224, and data augmentation methods are not used. We employ IWCV (Sugiyama et al., 2007), a model selection method for domain adaptation, to determine the hyper-parameters and the number of iterations for all methods. The learning rates of layers trained from scratch are set to 10 times those of fine-tuned layers. The batch size is $b = 36$. We use mini-batch SGD with a momentum of 0.95 with the learning rate of 0.1 and the progressive training strategies of DANN (Ganin et al., 2016). All experiments run five times and average results are reported.

[3] http://pytorch.org

Baseline Methods. We compare with state-of-the-art unsupervised domain adaptation methods that can be applied to regression problems. Some are reliable methods for domain adaptation regression (DAR) in the shallow regime: Transfer Component Analysis (**TCA**) (Pan et al., 2011), Joint Distribution Optimal Transport (**JDOT**) (Courty et al., 2017). Some are excellent deep representation learning methods designed for domain adaptation classification (DAC): Deep Adaptation Network (**DAN**) (Long et al., 2015), Domain Adversarial Neural Network (**DANN**) (Ganin et al., 2016), Maximum Classifier Discrepancy (**MCD**) (Saito et al., 2018), Adaptive Feature Norm (**AFN**) (Xu et al., 2019).

4.3. Results

Results on 2D Synthetic Dataset. The aim we evaluate methods on the easy dataset, dSprites, is to test their ability to capture the key object. The Mean Absolute Error (MAE) of all tasks are summed up, that is, three regression tasks (scale, position X and position Y) on dSprites with results in Table 4. **RSD+BMP** substantially boosts the performance and yields state-of-the-art results. There are obvious gains on relatively difficult task $\mathbf{C} \rightarrow \mathbf{N}$ and $\mathbf{C} \rightarrow \mathbf{S}$.

Results on 3D Simulation-to-Real Dataset. This challenging simulation-to-real dataset can help us evaluate how well transferable representations are learned. MAE of two tasks (rotation about two axes) are summed up, with results in Table 5. **RSD+BMP** significantly outperforms existing methods and yields state-of-the-art results. It is obvious that on two difficult tasks $\mathbf{T} \rightarrow \mathbf{RL}$ and $\mathbf{T} \rightarrow \mathbf{RC}$ (synthetic to real), only **RSD** cannot achieve satisfactory performance. Further maintaining the geometrical properties of the deep representations using $\mathcal{L}_{\mathrm{BMP}}$ can boost performance.

Results on Real-World Dataset. This real-world dataset can help us evaluate the practicability of domain adaptation regression methods. MAE of three tasks (pitch, yaw and roll) are summed up, with results in Table 6. **RSD+BMP** improves the performance of real-world datasets sharply and yields state-of-the-art results.

4.4. Analyses

We perform extensive analytical experiments to further verify the effectiveness and practicability of our method.

Ablation Study. We observe the effectiveness of the two regularizers in Tables 4, 5 and 6. Without BMP, RSD can achieve great results in simple tasks. But when dealing with some challenging tasks, RSD may break the geometrical properties (the ordering of importance for each orthogonal basis) of deep representations. Thus BMP does a good job in the deep representation learning process. All two loss terms are effective in boosting performances.

Table 4. Sum of MAE across three regression tasks on dSprites: unsupervised domain adaptation (ResNet-18).

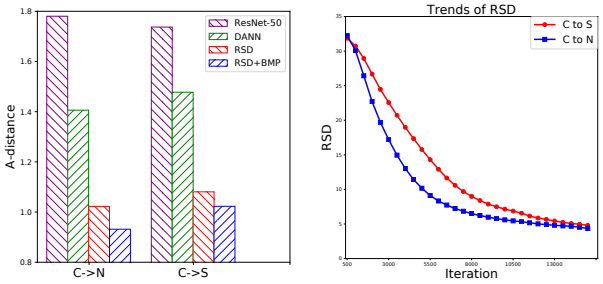
Method	C → N	C → S	N → C	N → S	S → C	S → N	Avg
ResNet-18 (He et al., 2016)	0.94 ± 0.06	0.90 ± 0.08	0.16 ± 0.02	0.65 ± 0.02	0.08 ± 0.01	0.26 ± 0.03	0.498
TCA (Pan et al., 2011)	0.94 ± 0.03	0.87 ± 0.02	0.19 ± 0.02	0.66 ± 0.05	0.10 ± 0.02	0.23 ± 0.04	0.498
DAN (Long et al., 2015)	0.70 ± 0.05	0.77 ± 0.09	**0.12** ± 0.03	0.50 ± 0.05	0.06 ± 0.02	0.11 ± 0.04	0.377
DANN (Ganin et al., 2016)	0.47 ± 0.07	0.46 ± 0.07	0.16 ± 0.02	0.65 ± 0.05	**0.05** ± 0.00	0.10 ± 0.01	0.315
JDOT (Courty et al., 2017)	0.86 ± 0.03	0.79 ± 0.02	0.19 ± 0.02	0.64 ± 0.05	0.10 ± 0.02	0.23 ± 0.04	0.468
MCD (Saito et al., 2018)	0.81 ± 0.09	0.81 ± 0.12	0.17 ± 0.12	0.65 ± 0.03	0.07 ± 0.02	0.19 ± 0.04	0.450
AFN (Xu et al., 2019)	1.00 ± 0.04	0.96 ± 0.05	0.16 ± 0.03	0.62 ± 0.04	0.08 ± 0.01	0.32 ± 0.06	0.523
RSD (ours)	0.32 ± 0.02	0.35 ± 0.02	0.16 ± 0.02	0.57 ± 0.01	0.08 ± 0.01	0.09 ± 0.02	0.258
RSD+BMP (ours)	**0.31** ± 0.03	**0.31** ± 0.03	**0.12** ± 0.02	**0.53** ± 0.01	0.07 ± 0.00	**0.08** ± 0.01	**0.237**

Table 5. Sum of MAE across two regression tasks on MPI3D: unsupervised domain adaptation (ResNet-18).

Method	RL → RC	RL → T	RC → RL	RC → T	T → RL	T → RC	Avg
ResNet-18 (He et al., 2016)	0.17 ± 0.02	0.44 ± 0.04	0.19 ± 0.02	0.45 ± 0.03	0.51 ± 0.01	0.50 ± 0.03	0.377
TCA (Pan et al., 2011)	0.17 ± 0.02	0.42 ± 0.01	0.19 ± 0.02	0.42 ± 0.02	0.50 ± 0.02	0.50 ± 0.02	0.373
DAN (Long et al., 2015)	0.12 ± 0.03	0.35 ± 0.02	0.12 ± 0.02	0.27 ± 0.02	0.40 ± 0.02	0.41 ± 0.04	0.278
DANN (Ganin et al., 2016)	**0.09** ± 0.01	0.24 ± 0.04	0.11 ± 0.03	0.41 ± 0.03	0.48 ± 0.02	0.37 ± 0.04	0.283
JDOT (Courty et al., 2017)	0.16 ± 0.02	0.41 ± 0.01	0.16 ± 0.02	0.41 ± 0.02	0.47 ± 0.02	0.47 ± 0.02	0.353
MCD (Saito et al., 2018)	0.13 ± 0.02	0.40 ± 0.04	0.15 ± 0.02	0.45 ± 0.01	0.52 ± 0.02	0.50 ± 0.03	0.358
AFN (Xu et al., 2019)	0.18 ± 0.03	0.45 ± 0.02	0.20 ± 0.03	0.46 ± 0.03	0.53 ± 0.02	0.52 ± 0.04	0.390
RSD (ours)	0.10 ± 0.01	0.23 ± 0.03	0.11 ± 0.01	0.17 ± 0.02	0.41 ± 0.01	0.42 ± 0.01	0.242
RSD+BMP (ours)	**0.09** ± 0.01	**0.19** ± 0.02	**0.08** ± 0.00	**0.15** ± 0.03	**0.36** ± 0.01	**0.36** ± 0.02	**0.205**

Table 6. Sum of MAE across three regression tasks on Biwi Kinect

Method	F → M	M → F
ResNet-18 (He et al., 2016)	0.38 ± 0.02	0.29 ± 0.01
TCA (Pan et al., 2011)	0.39 ± 0.01	0.31 ± 0.01
DAN (Long et al., 2015)	0.37 ± 0.01	0.28 ± 0.01
DANN (Ganin et al., 2016)	0.37 ± 0.02	0.30 ± 0.01
JDOT (Courty et al., 2017)	0.39 ± 0.01	0.29 ± 0.02
MCD (Saito et al., 2018)	0.37 ± 0.02	0.31 ± 0.02
AFN (Xu et al., 2019)	0.41 ± 0.02	0.32 ± 0.02
RSD (ours)	0.33 ± 0.02	0.27 ± 0.01
RSD+BMP (ours)	**0.30** ± 0.02	**0.26** ± 0.01



(a) A-distance (b) Trends of RSD

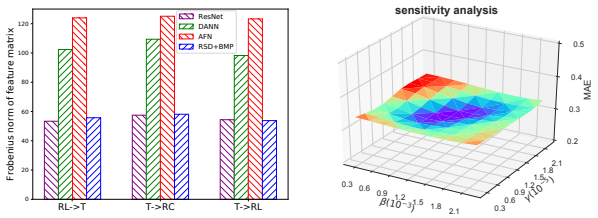
Figure 7. (a) A-distance on **C** → **N** and **C** → **S**. (b) The trends of Representation Subspace Distance (RSD) on **C** → **N** and **C** → **S**.

Representation Transferability. The A-distance (Ben-David et al., 2010) is a measure for distribution discrepancy, and thus we can use it to evaluate the transferability of representations. A-distance is defined as $\mathsf{dis}(A) = 1 - 2\epsilon$, where ϵ is the test error of a classifier trained to discriminate the source from the target. Results of A-distance of representations trained with different methods are shown in Figure 7(a). It is verified that our method has a smaller A-distance and can help learn transferable representations.

Representation Subspace Distance. We plot the trends of RSD on **C** → **S** and **C** → **N** in Figure 7(b). We observe that RSD in **C** → **S** is harder to be reduced than in **C** → **N**, which coherently implies their transfer difficulties.

Feature Scale. We plot the average Frobenius norm of source domain feature matrix in Figure 8(a). A key observation is that **our method does not change feature scale**.

Hyperparameter Sensitivity. MAE error on **C** → **S** with respect to different values of hyperparameter β and γ is shown in Figure 8. Results confirm that our method is not



(a) Feature Scale (b) $\mathbf{C} \rightarrow \mathbf{S}$

Figure 8. (a) Feature scale of different methods; (b) Hyperparameter sensitivity on transfer task $\mathbf{C} \rightarrow \mathbf{S}$.

sensitive to hyperparameters.

Time Complexity. Time consumption of SVD in a batch (of size $b = 36$) is acceptable (Chen et al., 2019). In one iteration of our method, three SVD operations are conducted. The training time of one iteration for source only model and our model are 0.203 seconds and 0.229 seconds, respectively. The extra time overhead is acceptable.

5. Conclusion

This paper studies the domain adaptation regression (DAR) problem in the deep representation learning regime. We find that regression performances are not robust to feature scaling. To tackle this challenge, we close the domain gap based on orthogonal bases in representation subspace instead of instance representations by exploring the Riemannian geometry of Grassmann manifold. Two novel regularizers are proposed to achieve the goal. RSD, a general metric on the Grassmannian, is proposed to assist in learning transferable representation. BMP, a crucial regularizer for every component in representation matrices, is proposed to maintain the geometrical structures of deep representations. With these regularizers, transferable representations are learned during training, and sharp gains are observed on regression tasks.

Acknowledgements

We thank Yuchen Zhang at Tsinghua University for insightful discussions. This work was supported by National Key R&D Program of China (2020AAA0109201), NSFC grants (62022050, 62021002, 61772299), Beijing Nova Program (Z201100006820041), and MOE Innovation Plan of China.

References

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

Cao, B., Pan, S. J., Zhang, Y., Yeung, D.-Y., and Yang, Q. Adaptive transfer learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

Chen, X., Wang, S., Long, M., and Wang, J. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, pp. 1081–1090, 2019.

Cortes, C. and Mohri, M. Domain adaptation in regression. In *International Conference on Algorithmic Learning Theory (ALT)*, pp. 308–323. Springer, 2011.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3730–3739, 2017.

de Mathelin, A., Richard, G., Mougeot, M., and Vayatis, N. Adversarial weighting for domain adaptation in regression. *arXiv preprint arXiv:2006.08251*, 2020.

Fanelli, G., Dantone, M., Gall, J., Fossati, A., and Gool, L. Random forests for real time 3d face analysis. *Int. J. Comput. Vision*, 101(3):437–458, February 2013.

Ganin, Y. and Lempitsky, V. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning (ICML)*, pp. 1180–1189, 2015.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research (JMLR)*, 17(1): 2096–2030, 2016.

Geng, X., Zhou, Z.-H., and Smith-Miles, K. Automatic age estimation based on facial aging patterns. *IEEE Transactions on pattern analysis and machine intelligence (TPAMI)*, 29(12):2234–2240, 2007.

Golub, G. H. and Van Loan, C. F. *Matrix computations (3rd ed.)*. DBLP, 1996.

Gondal, M., Wuthrich, M., Miladinovic, D., Locatello, F., Breidt, M., Volchkov, V., Akpo, J., Bachem, O., Schölkopf, B., and Bauer, S. On the transfer of inductive biasfrom simulation to the real world: a new disentanglement dataset. In *Neural Information Processing Systems (NeurIPS)*, 2019.

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

Gong, B., Grauman, K., and Sha, F. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In

International Conference on Machine Learning (ICML), pp. 222–230, 2013.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In *Neural Information Processing Systems (NeurIPS)*, 2014.

Gopalan, R., Li, R., and Chellappa, R. Domain adaptation for object recognition: An unsupervised approach. In *International conference on computer vision (ICCV)*, pp. 999–1006. IEEE, 2011.

Guo, G., Fu, Y., Dyer, C. R., and Huang, T. S. Image-based human age estimation by manifold learning and locally adjusted robust regression. *IEEE Transactions on Image Processing (TIP)*, 17(7):1178–1188, 2008.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 770–778, 2016.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017.

Huang, J., Gretton, A., Borgwardt, K. M., Schölkopf, B., and Smola, A. J. Correcting sample selection bias by unlabeled data. In *Advances in neural information processing systems (NeurIPS)*, pp. 601–608, 2007.

Lathuilière, S., Mesejo, P., Alameda-Pineda, X., and Horaud, R. A comprehensive analysis of deep regression. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2019.

Li, Y., Wang, N., Shi, J., Liu, J., and Hou, X. Revisiting batch normalization for practical domain adaptation. *arXiv preprint arXiv:1603.04779*, 2016.

Long, M., Cao, Y., Wang, J., and Jordan, M. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning (ICML)*, pp. 97–105, 2015.

Long, M., Zhu, H., Wang, J., and Jordan, M. I. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning (ICML)*, pp. 2208–2217. JMLR. org, 2017.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1647–1657, 2018.

Luo, Z., Zou, Y., Hoffman, J., and Fei-Fei, L. F. Label efficient learning of transferable representations acrosss domains and tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 165–177, 2017.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation: Learning bounds and algorithms. In *Conference on Learning Theory (COLT)*, 2009.

Maria Carlucci, F., Porzi, L., Caputo, B., Ricci, E., and Rota Bulo, S. Autodial: Automatic domain alignment layers. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 5067–5075, 2017.

Nikzad-Langerodi, R., Zellinger, W., Saminger-Platz, S., and Moser, B. A. Domain adaptation for regression under beer–lambert's law. *Knowledge-Based Systems (KBS)*, 210:106447, 2020.

Pan, S. J., Tsang, I. W., Kwok, J. T., and Yang, Q. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks (TNN)*, 22(2):199–210, 2011.

Peng, X., Huang, Z., Sun, X., and Saenko, K. Domain agnostic learning with disentangled representations. In *International Conference on Machine Learning (ICML)*, pp. 5102–5112, 2019.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

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

Saito, K., Watanabe, K., Ushiku, Y., and Harada, T. Maximum classifier discrepancy for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3723–3732, 2018.

Shimodaira, H., Sugiyama, M., Storkey, A., Gretton, A., David, S.-B., QuinoneroCandela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. *Dataset Shift in Machine Learning*, pp. 201–205. Neural Information Processing Series. Yale University Press in association with the Museum of London, 2009. ISBN 978-0-26217-005-5.

Singh, A. and Chakraborty, S. Deep domain adaptation for regression. In *Development and Analysis of Deep Learning Architectures*, pp. 91–115. Springer, 2020.

Sugiyama, M., Krauledat, M., and Müller, K.-R. Covariate shift adaptation by importance weighted cross validation.

Journal of Machine Learning Research (JMLR), 8:1027–1061, 2007.

Teshima, T., Sato, I., and Sugiyama, M. Few-shot domain adaptation by causal mechanism transfer. In *International Conference on Machine Learning (ICML)*, pp. 9458–9469. PMLR, 2020.

Tzeng, E., Hoffman, J., Darrell, T., and Saenko, K. Simultaneous deep transfer across domains and tasks. In *International Conference on Computer Vision (ICCV)*, pp. 4068–4076, 2015.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pp. 4, 2017.

Xu, R., Li, G., Yang, J., and Lin, L. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *International Conference on Computer Vision (ICCV)*, pp. 1426–1435, 2019.

Yamada, M., Sigal, L., and Chang, Y. Domain adaptation for structured regression. *International journal of computer vision (IJCV)*, 109(1-2):126–145, 2014.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems (NeurIPS)*, pp. 3320–3328, 2014.

Zhang, Y., Liu, T., Long, M., and Jordan, M. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning (ICML)*, pp. 7404–7413, 2019.

Zheng, J., Liu, M.-Y., Chellappa, R., and Phillips, P. J. A grassmann manifold-based domain adaptation approach. In *International Conference on Pattern Recognition (ICPR)*, pp. 2095–2099. IEEE, 2012.