Learning to Transfer with von Neumann Conditional Divergence

Ammar Shaker[1]*, **Shujian Yu**[2,3]*, **Daniel Oñoro-Rubio**[1]

[1] NEC Laboratories Europe, Heidelberg, Germany
[2] UiT - The Arctic University of Norway, Tromsø, Norway
[3] Xi'an Jiaotong University, Xi'an, Shaanxi, China
ammar.shaker@neclab.eu, yusj9011@gmail.com, daniel.onoro@neclab.eu

Abstract

The similarity of feature representations plays a pivotal role in the success of problems related to domain adaptation. Feature similarity includes both the invariance of marginal distributions and the closeness of conditional distributions given the desired response y (e.g., class labels). Unfortunately, traditional methods always learn such features without fully taking into consideration the information in y, which in turn may lead to a mismatch of the conditional distributions or the mix-up of discriminative structures underlying data distributions. In this work, we introduce the recently proposed von Neumann conditional divergence to improve the transferability across multiple domains. We show that this new divergence is differentiable and eligible to easily quantify the functional dependence between features and y. Given multiple source tasks, we integrate this divergence to capture discriminative information in y and design novel learning objectives assuming those source tasks are observed either simultaneously or sequentially. In both scenarios, we obtain favorable performance against state-of-the-art methods in terms of smaller generalization error on new tasks and less catastrophic forgetting on source tasks (in the sequential setup).

Introduction

Deep learning has achieved remarkable successes in diverse machine learning problems and applications (Pouyanfar, Sadiq et al. 2018). However, most of deep learning applications are limited to a single or isolated task, in which a network is usually trained from scratch based on a large scale labeled dataset (Donahue, Jia et al. 2014). As a result, the training of deep neural networks becomes frustrating when labeled data is scarce or expensive to obtain. In these scenarios, the efficient transfer of information from one or multiple tasks to another and the prevention of negative transfer amongst all tasks become fundamental techniques for the successful deployment of a deep learning system (Yosinski et al. 2014; Riemer et al. 2019).

Different problems arise depending on the number of tasks and how tasks arrive (e.g., concurrently or sequentially). These problems range from the standard domain adaptation from a single source domain to a target domain (Pan et al. 2010), up to the continual learning which trains a single network on a series of interrelated tasks (Parisi, Kemker et al. 2019; Delange et al. 2021), with the goal of improving positive transfer and mitigating negative interference (Riemer et al. 2019).

Tremendous efforts have been made to improve transferability across multiple domains (Ganin, Ustinova et al. 2016; Zhao, Zhang et al. 2018; Zhao et al. 2019). Most of the works aim to learn domain-invariant features $\mathbf{t}$ without the knowledge of class label or desired response y. Common techniques to match feature marginal distributions include the maximum mean discrepancy (MMD) (Pan et al. 2010; Zhu, Zhuang, and Wang 2019), the moment matching (Zellinger et al. 2017), the $\mathcal{H}$ divergence (Zhao, Zhang et al. 2018), the Wasserstein distance (Wang et al. 2019), etc. For classification, $p(y|\mathbf{t})$ can be modeled with a multinomial distribution (Pei et al. 2018; Zhao, Gong et al. 2020). However, it is still an open problem to explicitly capture the functional dependence between $\mathbf{t}$ and y for regression.

Let us consider a network that consists of a feature extractor $f_\theta : \mathcal{X} \to \mathcal{T}$ (parametrized by θ) and a predictor $h_\varphi : \mathcal{T} \to \mathcal{Y}$ (parameterized by φ); the similarity of latent representation $\mathbf{t}$ includes two aspects: the invariance of marginal distributions (i.e., $p(f_\theta(\mathbf{x}))$) across different domains and the functional closeness of using $\mathbf{t}$ to predict y. The predictive power of h_φ can be characterized by the conditional distribution $p(y|\mathbf{t})$. From an information-theoretic perspective, the conditional entropy $H(y|\mathbf{t}) = -\mathbb{E}(\log(p(y|\mathbf{x})))$ also measures the dependence between y and $\mathbf{t}$.

Our main contributions are summarized as follows:

- We introduce the von Neumann conditional divergence D_{vN} (Yu et al. 2020) to the problems of domain adaptation. This new divergence can easily quantify the functional dependence between latent features $\mathbf{t}$ and the desired response y, in both classification and regression.
- We show the utility of D_{vN} in a standard domain adaptation setup in which multiple source tasks are observed either simultaneously (*a.k.a.*, multi-source domain adaptation) or sequentially (*a.k.a.*, continual learning).
- For multi-source domain adaptation (MSDA),
  - Given a hypothesis set $\mathcal{H}$ and the new loss function induced by D_{vN}, we define a new domain discrepancy distance $\mathcal{D}_{\text{M-disc}}(P, Q)$ to measure the closeness of two distributions P and Q.

*A. Shaker and S. Yu are the corresponding authors

  - By generating a weighted source domain D_α with probability $P_\alpha = \sum_{i=1}^{K} w_i P_{s_i}$ (subject to $\sum_{i=1}^{K} w_i = 1$), in which P_{s_i} denotes the distribution of the i-th source domain, we derive a new generalization bound based on $\mathcal{D}_{\text{M-disc}}$ for MSDA.
  - We design a new objective based on the derived bound and optimize it as a min-max game. Compared to four state-of-the-art (SOTA) methods, our approach reduces the generalization error and identifies meaningful strength of "relatedness" from each source to the target domain.
- For the problem of continual learning (CL),
  - We show that the functional similarity of latent features $\mathbf{t}$ to the desired response y is able to quantify the importance of network parameters to previous tasks. Based on this observation, we develop a new regularization-based CL approach by network modularization (Watanabe, Hiramatsu, and Kashino 2018).
  - We compare our approach with the baseline elastic weight consolidation (EWC) (Kirkpatrick, Pascanu et al. 2017) and three other SOTA methods on five benchmark datasets. Empirical results demonstrate that our approach reduces catastrophic forgetting and is less sensitive to the choice of hyper-parameters.

Background Knowledge

Problem Setup

Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and the desired response (e.g., class labels) spaces. Given K source domains (or tasks) $\{D_i\}_{i=1}^{K}$, we obtain N_i training samples $\{\mathbf{x}_i^j, y_i^j\}_{j=1}^{N_i}$ in the i-th source D_i, which follows a distribution $P_i(\mathbf{x}, y)$ (defined over $\mathcal{X} \times \mathcal{Y}$).

In a typical (unsupervised) domain adaptation setup, the goal is to generalize a parametric model learned from data samples in $\{D_i\}_{i=1}^{K}$ to a different, but related, target domain D_{K+1} following a new distribution $P_{K+1}(\mathbf{x}, y)$, in which we assume no access to the true response y in the data sampled from $P_{K+1}(\mathbf{x}, y)$, i.e., minimizing the objective

$$\mathbb{E}_{(\mathbf{x},y)\sim D_{K+1}} \left[\ell(w; \mathbf{x}, y)\right], \tag{1}$$

where $\ell(w; \mathbf{x}, y) : \mathcal{W} \to \mathbb{R}$ is the loss function of w associated with sample $(\mathbf{x}, y)$, and $\mathcal{W} \subseteq \mathbb{R}^d$ is the model parameter space.

In an online scenario where tasks arrive sequentially, lifelong learning searches for models minimizing the population loss over all seen $(K+1)$ tasks, where access to previous tasks $\{D_i\}_{i=1}^{K}$ is either limited or prohibited:

$$\sum_{i=1}^{K+1} \mathbb{E}_{(\mathbf{x},y)\sim D_i} \left[\ell(w; \mathbf{x}, y)\right]. \tag{2}$$

Obviously, this poses new challenges, as the network is required to ensure positive transfer from $\{D_i\}_{i=1}^{K}$ to D_{K+1}, and, at the same time, avoid negative interference to its performance on $\{D_i\}_{i=1}^{K}$.

In this work, we consider multi-source domain adaptation for regression (i.e., $y \in \mathbb{R}$) and a standard continual learning setup on image classification (i.e., y contains m unique categories $\{c_1, \ldots, c_m\}$).

von Neumann Conditional Divergence

Let us draw N samples from two joint distributions $P_1(\mathbf{x}, y)$ and $P_2(\mathbf{x}, y)$, i.e., $\{\mathbf{x}_1^i, y_1^i\}_{i=1}^{N}$ and $\{\mathbf{x}_2^i, y_2^i\}_{i=1}^{N}$. Here, y refers to the response variable, and $\mathbf{x}$ can be either the raw input variable or the feature vector $\mathbf{z} = f_\theta(\mathbf{x})$ after a feature extractor $f_\theta : \mathcal{X} \to \mathcal{Z}$ parameterized by θ.

Yu et al. (Yu et al. 2020) define the relative divergence from $P_1(y|\mathbf{x})$ to $P_2(y|\mathbf{x})$ as:

$$D(P_1(y|\mathbf{x}) \| P_2(y|\mathbf{x})) = D_{vN}(\sigma_{\mathbf{x}y} \| \rho_{\mathbf{x}y}) - D_{vN}(\sigma_{\mathbf{x}} \| \rho_{\mathbf{x}}), \tag{3}$$

where $\sigma_{\mathbf{x}y}$ and $\rho_{\mathbf{x}y}$ denote the sample covariance matrices evaluated on $\{\mathbf{x}_1^i, y_1^i\}_{i=1}^{N}$ and $\{\mathbf{x}_2^i, y_2^i\}_{i=1}^{N}$, respectively. Similarly, σ_x and ρ_x refer to the sample covariance matrices evaluated on $\{\mathbf{x}_1^i\}_{i=1}^{N}$ and $\{\mathbf{x}_2^i\}_{i=1}^{N}$, respectively. D_{vN} is the von Neumann divergence (Nielsen and Chuang 2002; Kulis, Sustik, and Dhillon 2009), $D_{vN}(\sigma \| \rho) = \operatorname{tr}(\sigma \log \sigma - \sigma \log \rho - \sigma + \rho)$, which operates on two symmetric positive definite (SPD) matrices, σ and ρ. Eq. (3) is not symmetric. To achieve symmetry, one can simply take the form:

$$\begin{aligned} D(P_1(y|\mathbf{x}) : P_2(y|\mathbf{x})) = \; & \frac{1}{2}(D(P_1(y|\mathbf{x}) \| P_2(y|\mathbf{x})) \\ + \; & D(P_2(y|\mathbf{x}) \| P_1(y|\mathbf{x}))). \end{aligned} \tag{4}$$

As a complement to (Yu et al. 2020), we additionally provide the convergence behavior analysis of the matrix-based von Neumann divergence on sample covariance matrix to the true distributional distance (see supplementary material), although this is not the main contribution of this work.

Note that, aligning distributions or conditional distributions always plays a pivotal role in different domain adaptation related problems. Before our work, the MMD has been extensively investigated. However, there is no universal agreement on the definition of conditional MMD (Park and Muandet 2020), and most of existing operator-based approaches on conditional MMD depend on stringent assumptions which are usually violated in practice (e.g., (Ren et al. 2016)). This unfortunate fact urges the need for exploring the possibility of a new divergence measure that is both simple to compute and differentiable. Moreover, compared to MMD that relies on a kernel function with width σ which is always hard to tune in practice, Eqs. (3) and (4) defined over sample covariance matrix are hyper-parameter free.

Interpreting the von Neumann Conditional Divergence as a Loss Function

In case $P_1(\mathbf{x}, y)$ and $P_2(\mathbf{x}, y)$ have the same marginal distribution $P(\mathbf{x})$ or share the same input variable $\mathbf{x}$ (i.e., $\sigma_{\mathbf{x}} = \rho_{\mathbf{x}}$), the symmetric von Neumann conditional divergence (Eq. (4)) reduces to:

$$D(P_1(y|\mathbf{x}) : P_2(y|\mathbf{x})) = \frac{1}{2} \operatorname{tr}\left((\sigma_{\mathbf{x}y} - \rho_{\mathbf{x}y})(\log \sigma_{\mathbf{x}y} - \log \rho_{\mathbf{x}y})\right). \tag{5}$$

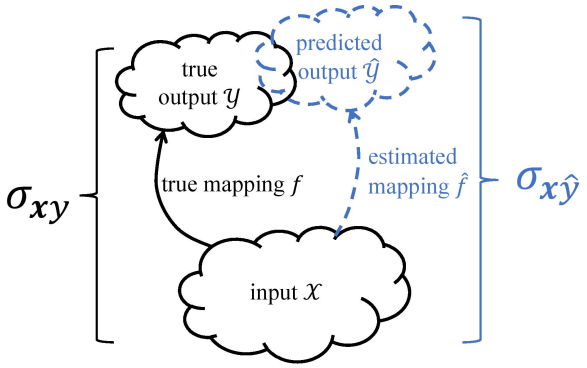


Figure 1: The geometry of loss $\mathcal{L} : \mathbb{S}^p_{++} \times \mathbb{S}^p_{++} \to \mathbb{R}_+$: our $\sqrt{J_{vN}}$ searches for an "optimal" predictor $\hat{f}$ that minimizes the discrepancy between the two covariance matrices $\sigma_{\mathbf{x},f(\mathbf{x})}$ and $\sigma_{\mathbf{x},\hat{f}(\mathbf{x})}$.

We term the r.h.s. of Eq. (5) as the Jeffery von Neumann divergence on $\sigma_{\mathbf{x}y}$ and $\rho_{\mathbf{x}y}$, and denote it as $J_{vN}(\sigma_{\mathbf{x}y} : \rho_{\mathbf{x}y})$.

Taking $X = \sigma_{\mathbf{x},f(\mathbf{x})}$ and $Y = \sigma_{\mathbf{x},\hat{f}(\mathbf{x})}$, $\sqrt{J_{vN}(\sigma_{\mathbf{x},f(\mathbf{x})} : \sigma_{\mathbf{x},\hat{f}(\mathbf{x})})}$ can be interpreted and used as a loss function to train a deep neural network. Here, $\mathbf{x}$ refers to the input variable, $f : \mathbf{x} \to y$ is the true labeling or mapping function, $\hat{f}$ is the estimated predictor, $f(\mathbf{x}) = y$ is the true label or response variable, and $\hat{f}(\mathbf{x}) = \hat{y}$ is the predicted output. $\sigma_{\mathbf{x},f(\mathbf{x})}$ and $\sigma_{\mathbf{x},\hat{f}(\mathbf{x})}$ denote the covariance matrices for the pairs of variables $\{\mathbf{x}, f(\mathbf{x})\}$ and $\{\mathbf{x}, \hat{f}(\mathbf{x})\}$, respectively. Fig. 1 depicts an illustrative explanation.

Before presenting our methodology in both multi-source domain adaptation and continual learning, we show three appealing properties associated with $\sqrt{J_{vN}}$ (see supplementary material for proofs and empirical justifications[1]):

- $\sqrt{J_{vN}}$ has an analytical gradient and is automatically differentiable;
- Compared with the mean square error (MSE) loss, $\sqrt{J_{vN}(\sigma_{\mathbf{x},f(\mathbf{x})} : \sigma_{\mathbf{x},\hat{f}(\mathbf{x})})}$ enjoys improved robustness.
- Compared with the cross-entropy (CE) loss, $\sqrt{J_{vN}(\sigma_{\mathbf{x},f(\mathbf{x})} : \sigma_{\mathbf{x},\hat{f}(\mathbf{x})})}$ satisfies the triangle inequality. That is, given three models f_1, f_2 and f_3, we have: $\sqrt{J_{vN}(\sigma_{\mathbf{x},f_1(\mathbf{x})} : \sigma_{\mathbf{x},f_2(\mathbf{x})})} \leq \sqrt{J_{vN}(\sigma_{\mathbf{x},f_1(\mathbf{x})} : \sigma_{\mathbf{x},f_3(\mathbf{x})})} + \sqrt{J_{vN}(\sigma_{\mathbf{x},f_3(\mathbf{x})} : \sigma_{\mathbf{x},f_2(\mathbf{x})})}$.

MSDA by Matrix-based Discrepancy Distance

Bounding the von Neumann Conditional Divergence in Target Domain

Motivated by the discrepancy distance D_{disc} (Cortes and Mohri 2014) based on a loss function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$, we first present our matrix-based discrepancy distance $D_{\text{M-disc}}$ to quantify the discrepancy between two distributions P and Q over $\mathcal{X}$ based on our new loss $\mathcal{L} : \mathbb{S}^p_{++} \times \mathbb{S}^p_{++} \to \mathbb{R}_+$ (i.e., $\sqrt{J_{vN}(\sigma_{\mathbf{x},f(\mathbf{x})} : \sigma_{\mathbf{x},\hat{f}(\mathbf{x})})}$).

Definition 1. *The matrix-based discrepancy distance ($D_{\text{M-disc}}$) measures the longest distance between two domains (with respect to the hypothesis space $\mathcal{H}$) in a metric space equipped with the square root of Jeffery von Neumann divergence J_{vN} as a distance function. Given domains D_s and D_t and their corresponding distributions P_s and P_t, for any two hypotheses $h, h' \in \mathcal{H}$, $D_{\text{M-disc}}$ takes the form:*

$$D_{\text{M-disc}}(P_s, P_t) = \max_{h,h' \in \mathcal{H}} \Big| \sqrt{J_{vN}(\sigma^s_{x,h(x)} : \sigma^s_{x,h'(x)})} - \sqrt{J_{vN}(\sigma^t_{x,h(x)} : \sigma^t_{x,h'(x)})} \Big|, \tag{6}$$

with $a \in \{s, t\}$ and $g \in \{h, h'\}$, the matrix $\sigma^a_{\mathbf{x},g(\mathbf{x})}$ is the covariance matrix for the pair of variable $\mathbf{x}, g(\mathbf{x})$ in domain D_a.

Same to the notable $\mathcal{H}\Delta\mathcal{H}$ divergence in binary classification (Ben-David et al. 2010), $D_{\text{M-disc}}$ reaches the maximum value if a predictor h' is very close to h on the source domain but far on the target domain (or vice-versa). When fixing h, $D_{\text{M-disc}}(P_s, P_t; h)$ simply searches only for $h' \in \mathcal{H}$ maximizing Eq. (6). The following theorem presents a new generalization upper bound for the square root of J_{vN} on the target domain with respect to that of multiple sources.

Theorem 2. *Let $S = \{D_{s_1}, \ldots, D_{s_K}\}$ be the a set of K source domains, and denote the ground truth mapping function in D_{s_i} as f_{s_i}. Assign the weight w_i to source D_{s_i} (subject to $\sum_{i=1}^K w_i = 1$) and generate a weighted source domain D_α, such that the source distribution $P_\alpha = \sum_{i=1}^K w_i P_{s_i}$ and the mapping function $f_\alpha : x \to \left(\sum_{i=1}^K w_i P_{s_i}(x) f_{s_i}(x)\right) / \left(\sum_{i=1}^K w_i P_{s_i}(x)\right)$. For any hypothesis $h \in \mathcal{H}$, the square root of J_{vN} on the target domain D_t is bound in the following way:*

$$\sqrt{J_{vN}(\sigma^t_{x,h(x)} : \sigma^t_{x,f_t(x)})} \leq \sum_{i=1}^K w_i \left(\sqrt{J_{vN}(\sigma^{s_i}_{x,h(x)} : \sigma^{s_i}_{x,f_{s_i}(x)})}\right) + D_{\text{M-disc}}(P_t, P_\alpha; h) + \eta_Q(f_\alpha, f_t), \tag{7}$$

where $\eta_Q(f_\alpha, f_t) = \min_{h^* \in \mathcal{H}} \sqrt{J_{vN}(\sigma^t_{x,h^*(x)} : \sigma^t_{x,f_t(x)})} + \sqrt{J_{vN}(\sigma^\alpha_{x,h^*(x)} : \sigma^\alpha_{x,f_\alpha(x)})}$ is the minimum joint empirical losses on the combined source D_α and the target D_t, achieved by an optimal hypothesis h^*.

The result presented in Theorem 2 can be interpreted as bounding the square root of J_{vN} on the target domain D_t by quantities controlled by *(i)* a convex combination over the square root of J_{vN} in each of the sources, i.e., $\sqrt{J_{vN}(\sigma^{s_i}_{x,h(x)} : \sigma^{s_i}_{x,f_{s_i}(x)})}$; *(ii)* the mismatch between the weighted distribution P_α and the target distribution P_t, i.e., $D_{\text{M-disc}}(P_t, P_\alpha; h)$; and *(iii)* the optimal joint empirical risk on source and target, i.e., $\eta_Q(f_\alpha, f_t)$. The last term is irrelevant to the optimization and is expected to be small (Zhao et al. 2019). Notice that η_Q is constant and only depends on

[1] Supplementary material is available in our Arxiv version https://arxiv.org/abs/2108.03531

h^* in the case of a single source. For multiple source domains, the quantity η_Q does include the weights $\mathbf{w}$, yet it is constant for a given $\mathbf{w}$.

Optimization by Adversarial Min-Max Game

Similar to the notable Domain-Adversarial Neural Networks (DANN) (Ganin, Ustinova et al. 2016) that implicitly performs distribution matching by an adversarial min-max game, we explicitly implement the idea exhibited by Theorem 2 and combine a feature extractor $f_\theta : \mathcal{X} \to \mathcal{T}$ and a class of predictor $\mathcal{H} : \mathcal{T} \to \mathcal{Y}$ in a unified learning framework:

$$\min_{\substack{f_\theta, h \in \mathcal{H} \\ ||\mathbf{w}||_1 = 1}} \max_{h' \in \mathcal{H}} \left(\sum_{i=1}^{K} w_i \sqrt{J_{vN}(\sigma^{s_i}_{x,h(f_\theta(x))} : \sigma^{s_i}_{x,y})} + \left| \sqrt{J_{vN}(\sigma^{t}_{f_\theta(\mathbf{x}),h(f_\theta(\mathbf{x}))} : \sigma^{t}_{f_\theta(\mathbf{x}),h'(f_\theta(\mathbf{x}))})} - \sum_{i=1}^{K} w_k \sqrt{J_{vN}(\sigma^{s_i}_{f_\theta(\mathbf{x}),h(f_\theta(x))} : \sigma^{s_i}_{f_\theta(\mathbf{x}),h'(f_\theta(x))})} \right| \right). \quad (8)$$

The first term of Eq. (8) enforces h to be a good predictor on all source tasks[2]; the second term is an explicit instantiation of our $D_{\text{M-dist}}(P_t, P_\alpha)$. The general idea is to find a feature extractor $f_\theta(\mathbf{x})$ that for any given pair of hypotheses h and h', it is hard to discriminate the target domain P_t from P_α, the weighted combination of the source distributions.

We term our method the multi-source domain adaptation with matrix-based discrepancy distance (MDD) (pseudocode in the supplementary material). We also noticed that a similar min-max training strategy has been used in (Pei et al. 2018; Saito, Kim et al. 2019; Richard et al. 2020).

Comparison with State-of-the-Art Methods

We evaluate our MDD on four real-world datasets *(i)* Amazon review dataset,[3] *(ii)* TRANCOS which is a public benchmark for extremely overlapping vehicle counting, *(iii)* the YearPredictionMSD data (Bertin-Mahieux et al. 2011), and *(iv)* the relative location of CT slices on the axial axis dataset (Graf et al. 2011).

The following six methods are used for comparison: *(1)* DANN (Ganin, Ustinova et al. 2016) is used by merging all sources into a single one; *(2)* MDAN-Max and *(3)* MDAN-Dyn, where MDAN refers to the multisource domain adversarial networks by (Zhao, Zhang et al. 2018). It also applies a weighting scheme to all sources. *(4)* Adversarial Hypothesis-Discrepancy Multi-Source Domain Adaptation (AHD-MSDA) (Richard et al. 2020) and its baseline *(5)* AHD-1S that merges all sources into one and then applies AHD-MSDA between the single combined source and the target domain. (6) Domain AggRegation Network (DARN) (Wen, Greiner, and Schuurmans 2020) after implementing the automatically differentiable maximum eigenvalue computation for the discrepancy computation.

[2]In practice, one can replace the J_{vN} loss with the root mean square error (RMSE) loss.

[3]https://www.cs.jhu.edu/~mdredze/datasets/sentiment/

In the first experiment, following (Richard et al. 2020), we employ a shallow neural network with two fully-connected hidden layers of size 500 with ReLU activation, and a dropout rate of 10%. The Adam optimizer is used with learning rate $lr = 0.001$, and batch size of 300. We use 30 training epochs, and perform 5 independent runs. Each domain is used once as target and the remaining as sources.

The Amazon review dataset is introduced in (Blitzer, Dredze, and Pereira 2007); it contains review texts and ratings of bought products. Products are grouped into categories. Following (Zhao, Zhang et al. 2018; Richard et al. 2020), we perform tf-idf transformation and select the top $1,000$ frequent words. Ratings are used as the target labels.

The TRaffic ANd COngestionS (TRANCOS) (Guerrero-Gómez-Olmedo et al. 2015) dataset is a public benchmark dataset for extremely overlapping vehicle counting with $1,244$ images and $46,700$ manually annotated vehicles via the dotting method (Lempitsky and Zisserman 2010). It contains images that were collected from 11 video surveillance cameras. We apply hierarchical clustering to formulate five domains over the cameras. The hourglass network (Newell, Yang, and Deng 2016) is used such that the encoder plays the role of the feature extractor, and the predictor and discriminator follow the decoder design. The predicted vehicle count is computed by integrating over the predicted density map after applying the ground truth mask, thereafter, the mean absolute error is computed on the predicted count. The quantitative results on these two datasets are summarized in Table 1 and Table 2, respectively. Our MDD always achieves the smallest mean absolute error on all target domains, except for "Dom2" of the counting problem. It is worth mentioning that DARN fails to generalize on source domains of TRANCOS and, hence, performs poorly on the target domain, as discussed in the supplementary material.

We also analyse the weights $\mathbf{w}$ learned by our MDD (plots and discussion in supplementary material). In general, our learned weights reflect the strength of relatedness from each source to the target. Moreover, we observe that our weights are much more stable across training epochs, whereas the weights learned by DARN always oscillate and are less linked in successive epochs.

Visualizing Domain Importance in Synthetic Data

We further evaluate the ability of MDD to discover the correct strength of relatedness from each source on a synthetic data, in which the "ground truth" of relatedness is known. We construct a synthetic data set with six domains each with features from $\mathbf{x} \in [-1, 1]^{12}$, and the Friedman target function (Friedman 1991) $y(\mathbf{x}) = 10 \sin(\pi x_1 x_3) + 20(x_5 - 0.5)^2 + 10x_7 + 5x_9 + \epsilon$, $\epsilon \sim \mathcal{N}(0, 1)$. The six generated domains are equally distributed in the diagonal of the space $[-1, 1]^{12}$. To this end, each domain $s_i \in \{s_1, \ldots, s_6\}$ is sampled from $\mathcal{N}(\mu^{(i)}, \Sigma^{(i)})$, such that $\mu^{(i)} = c_i \mathbf{1}_{12}$, where $c_i = (-1 + (2i - 2)/5)$ and $\mathbf{1}_{12}$ is the the all-one vector of size 12. The element of the covariance matrix $\Sigma^{(i)}$ are set to zero except for $\Sigma^{(i)}_{2i-1,2i} = \Sigma^{(i)}_{2i,2i-1} = 0.1$, $\Sigma^{(i)}_{2i,2i+1} = \Sigma^{(i)}_{2i+1,2i} = 0.07$ if $i < 6$, $\Sigma^{(i)}_{2i-1,2i-2} = \Sigma^{(i)}_{2i-2,2i-1} = 0.07$ if $i > 1$, and $\Sigma^{(i)}_{2k-1,2k} = \Sigma^{(i)}_{2k,2k-1} = 0.5$ where $k = i-1$ or

	AHD -1S	DANN -1S	AHD-MSDA	DARN	MDAN -Max	MDAN -Dyn	MDD
ba	0.627 (.003)	2.9 (1.3)	0.586 (.003)	0.755 (.001)	0.591 (.015)	0.711 (.006)	**0.581** (.003)
be	0.614 (.003)	1.1 (.2)	0.608 (.005)	0.69 (.001)	0.628 (.003)	0.656 (.004)	**0.588** (.003)
ca	0.559 (.003)	1.0 (.1)	0.534 (.006)	0.643 (.002)	0.522 (.005)	0.598 (.006)	**0.508** (.003)
co	0.617 (.005)	2.2 (.8)	0.61 (.004)	0.665 (.001)	0.682 (.016)	0.829 (.055)	**0.584** (.003)
el	0.669 (.002)	0.7 (.01)	0.657 (.002)	0.776 (.000)	0.654 (.001)	0.670 (.003)	**0.65** (.001)
go	0.585 (.002)	0.9 (.3)	0.566 (.003)	0.639 (.002)	0.552 (.003)	0.553 (.003)	**0.537** (.003)
gr	0.543 (.003)	1.5 (.8)	0.527 (.002)	0.627 (.002)	0.519 (.002)	0.538 (.003)	**0.513** (.009)

Table 1: Performance comparison in terms of mean absolute error (MAE) over five iterations on the Amazon rating data (with standard error in brackets). The best performance is marked in boldface. The categories are abbreviated as follows, ba:baby, be:beauty, ca:camera&photo, co:computer&video-games, al:electronics, go:gourmet-food, gr:grocery.

	AHD -1S	DANN -1S	AHD-MSDA	DARN	MDAN -Max	MDAN -Dyn	MDD
Dom1	46.87 (12.89)	16.19 (0.42)	57.19 (22.93)	—	32.17 (7.98)	29.35 (3.96)	**14.73** (0.52)
Dom2	27.39 (4.8)	21.7 (0.86)	33.8 (6.51)	—	18.02 (0.34)	**14.34**(0.24)	15.27 (0.92)
Dom3	63.69 (31.62)	28.43 (5.63)	63.27 (24.77)	—	38.5 (11.77)	26.81 (4.61)	**24.67** (3.43)
Dom4	23.02 (3.71)	21.54 (5.64)	88.07(52.72)	—	19.89 (3.83)	22.86 (1.04)	**14.25** (1.64)
Dom5	65.89 (22.71)	57.12 (29.74)	38.02 (11.7)	—	57.28 (36.24)	22.73(4.72)	**17.34** (1.43)

Table 2: Performance comparison in terms of mean absolute error (MAE) over three iterations on TRANCOS data (with standard error in brackets). The best performance is marked in boldface. DARN fails to generalize on the source domains, hence, performs very poorly on the target domains.

$i+1$. This way, the neighboring domains will have a gradual covariate shift in terms of both mean and covariance.

The distribution of the first two dimensions of $\mathbf{x}$ is depicted in Fig. 2a and the covariance matrix $\Sigma^{(i)}$ for domain i is illustrated in Fig. 2b. Fig. 2c to 2e show the weights learned by DARN, AHD-MSDA and MDD, respectively. The value in the (i, j)-th entry is the weight from source j, when the target is domain i. As can be seen, our MDD learns an almost symmetric weight matrix with high weights centered around the diagonal and smoothly fading weights in the anti-diagonal direction. AHD-MSDA seems to learn uniform weights. DARN learns sparse weights while often fails in ranking the sources in agreement with the ground truth.

Continual Learning by Representation Similarity Penalty

We demonstrate, in this section, that the von Neumann conditional divergence is also suitable to alleviate negative backward transfer or catastrophic forgetting in continual learning (CL). We exemplify our argument by proposing a new regularization-based CL approach.

Elastic Weight Consolidation (EWC) and its Extensions

Regularization approaches mitigate catastrophic forgetting by imposing penalties on the updates of the important neural weights (to previous tasks) (Parisi, Kemker et al. 2019; Delange et al. 2021). As a notable example in this category, EWC (Kirkpatrick, Pascanu et al. 2017) consists of a quadratic penalty on the difference between the parameters θ for the old and the new tasks. The objective to be minimized when observing task T_B after learning on task T_A is:

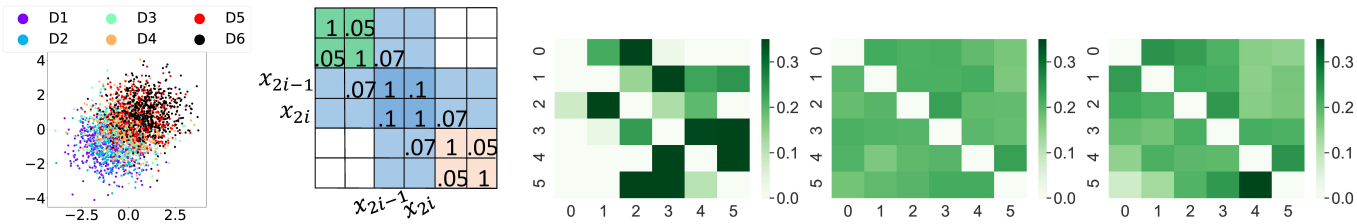
$$\mathcal{L}(\theta) = \mathcal{L}_B(\theta) + \sum_i \frac{\lambda}{2} \mathcal{F}_{\theta_i} (\theta_i - \theta^*_{A,i})^2, \qquad (9)$$

$\mathcal{L}_B(\theta)$ is the loss for task T_B, λ is the regularization strength, $\{\theta^*_{A,i}\}$ is the set of parameters after learning on task A, and F_θ is the diagonal Fisher information matrix (FIM). The i-th diagonal element of F_θ is computed as $F_{\theta_i} = \mathbb{E}[(\frac{\partial \mathcal{L}}{\partial \theta_i})^2]$. The supplementary material shows the derivation of Eq. (9).

EWC assumes all weights in θ are independent, which leads to a diagonal FIM. To make this assumption more practical, R-EWC (Liu et al. 2018) takes a factorized rotation of parameter space that leads to the desired diagonal FIM. (Chaudhry et al. 2018) reformulates the objective of EWC by KL-divergence in the Riemannian Manifold and suggests an efficient and online version of EWC. As an alternative to computing FIM, synaptic intelligence (SI) (Zenke, Poole, and Ganguli 2017) measures each parameter's importance by its accumulative contribution to the loss changes.

Measuring Weight Significance by Representation Similarity

In this section, we introduce a new form of regularization that measures the significance of a group of weights (rather than individual ones) to T_A by the (dis)similarity of local representations between T_A and T_B induced by these weights. Our method's essence comes from observing that



(a) Data (b) Submatrix of $\Sigma^{(i)}$ (c) DARN (d) AHD-MSDA (e) MDD

Figure 2: (a) The data used for the weight learning over synthetic data. (b) The submatrix of the covariance matrix $\Sigma^{(i)}$ of domain i. Figures (c), (d), and (e) show heatmaps of the learned weights of DARN, AHD-MSDA, and MDD, respectively. The rows and columns represent the target and source domains, respectively.

tasks with similar representations are more prone to overwrite or negatively affect each other. A similar observation has been recently discovered by (Ramasesh, Dyer, and Raghu 2020).

Specifically, in the d-th hidden layer, suppose we identified K groups of neurons $(g_1^d, g_2^d, \cdots, g_K^d)$ that are functionally mutually independent. Each group can be viewed as a module that operates independently. Therefore, changes to parameters belonging to the same module should be regularized together taking into account *(i)* their relatedness to the different tasks (through the von Neumann conditional divergence), and *(ii)* the parameter's interdependence through the network modularization. Taking these two aspects into consideration, we define a new regularization-based CL objective as:

$$\mathcal{L}(\theta) = \mathcal{L}_{T_B}(\theta) + \sum_{T_A \in \mathbb{T} \setminus \{T_B\}} \sum_{k,d} r_{k,d}^A \sum_{\theta_i \in g_k^d} (\theta_i - \theta^*_{T_A,i})^2, \tag{10}$$

$$r_{k,d}^A = \frac{1}{Z} \frac{\lambda}{2} \exp(-D(P_{T_A}(y|g_k^d(x)) : P_{T_B}(y|g_k^d(x)))). \tag{11}$$

Objective (10) iterates over each group g_k^d (second sum), and computes the representation similarity (11), induced by the sub-network associated by the group of neurons g_k^d, between the current task T_B and each previous task $T_A \in \mathbb{T} \setminus \{T_B\}$. This similarity takes the form of the softmax of the negative divergence with Z being the normalization term, and D is the symmetric von Neumann conditional divergence, i.e., Eq. (4). Based on this similarity, the change in the parameters of each group g_k^d is penalized by the representation indifference between the two tasks caused by that group. Hence, we call our method representation similarity penalty (RSP). For an architecture with R layers, RSP computes the groups for layers $d \in \{2, \ldots, R-1\}$, which leaves the parameters and bias of the first layer without assigned groups; for these parameters the Fisher index is used to weight the penalty.

Implementation Details and Empirical Evaluation

RSP employs the modularization strategy in (Watanabe, Hiramatsu, and Kashino 2018) to construct groups of neurons in each layer that are mutually independent. In our experiments, we fix the number of groups to be $K_d = 20$.

Setting, Datasets and Performance Measures The following empirical evaluations follow the continual learning setting described in (Riemer et al. 2019), where each sample of each task is observed in a single pass sequence. As for the neural network architecture, we use a single head fully-connected neural network with two hidden layers, each with 100 neurons, a 28×28 input layer, and an output layer with a single head with 10 units. This architecture is similar to the one used in (Lopez-Paz and Ranzato 2017). The hidden layers employ the ReLU activation, and SGD is used to minimize the softmax cross-entropy on the online training data.

We evaluate on the following datasets: *(i)* MNIST Permutations **(mnistP)** (Kirkpatrick, Pascanu et al. 2017), *(ii)* MNIST Rotations **(mnistR)** (Lopez-Paz and Ranzato 2017), *(iii)* Permuted Fashion-MNIST **(fashionP)** (Han, Kashif, and Roland 2017), and *(iv)* Permuted notMNIST **(notmnistP)**.[4] All these datasets contain images of size 28×28 pixels. Additionally, we also perform a comparison on the Omniglot dataset (Lake et al. 2011) using the first ten alphabets and a convolutional neural network; the setting and results are explained in the supplementary material.

To measure the learnability and resistance to forgetting, we compute the performance measures: *(i)* Learning accuracy (LA): the average accuracy on each task after learning it. *(ii)* Retained accuracy (RA): the average performance on all tasks after observing the last one. *(iii)* Backward transfer (BT): the loss in performance due to forgetting, i.e., the difference between LA and RA (Chaudhry et al. 2018).

Comparison Protocol and Results We compare the performance of our RSP against that of EWC, R-EWC, and two popular replay-based CL methods, namely the Averaged Gradient Episodic Memory (AGEM) (Chaudhry et al. 2019), and the Meta-Experience Replay (MER) (Riemer et al. 2019). A grid-based hyperparameter search is carried on for each method on each dataset as explained in the supplementary material. The ten datasets form a stream of ten tasks, each of which contains a sequence of only 1000 samples. Every time an evaluation is performed on a task, it is done on its test data of $10,000$ samples.

We employ the online setting with a restricted memory budget of ten samples per task. Table 3 shows that RSP out-

[4] http://yaroslavvb.blogspot.com/2011/09/notmnist-dataset.html

	AGEM			MER			R-EWC			EWC			RSP		
	RA	LA	BT	RA	LA	BT	RA	LA	BT	RA	LA	BT	RA	LA	BT
D1	66.6	78.5	-12	50.6	55.1	-4.6	69.8	**83.8**	-14	68.7	81	-12	**72.3**	79.5	-7.2
	(1.5)	(0.6)	(1.5)	(0.7)	(0.8)	(0.7)	(0.5)	(0.1)	(0.5)	(0.3)	(0.1)	(0.2)	(0.3)	(0.1)	(0.2)
D2	59.5	65.4	-5.9	53.3	61.2	-7.8	58.5	64.0	-5.4	42.2	56.2	-14	**62.5**	**66.6**	-4.2
	(0.5)	(0.3)	(0.5)	(0.1)	(0.8)	(0.9)	(0.8)	(0.1)	(0.7)	(2.1)	(1.4)	(0.8)	(0.3)	(0.1)	(0.3)
D3	75.0	85.6	-11	**81.2**	81.3	-0.2	60.9	**87.8**	-27	62.1	85.6	-24	62.9	83.6	-21
	(0.3)	(0.1)	(0.3)	(0.2)	(0.2)	(0.2)	(0.8)	(0.1)	(0.8)	(0.3)	(0.1)	(0.3)	(0.2)	(0.1)	(0.2)
D4	67	78.7	-12	68.9	75.9	-7.0	64.8	79.1	-14	66.1	77	-12	**71.8**	**80.8**	-9
	(0.4)	(0.3)	(0.6)	(0.3)	(0.2)	(0.3)	(0.5)	(0.2)	(0.4)	(1.9)	(0.7)	(1.3)	(0.2)	(0.1)	(0.2)

Table 3: Performance comparison between RSP, AGEM, MER, R-EWC and EWC. D1: not-mnistP, D2: fashionP, D3: mnistR, D4: mnistP. BT is rounded to the nearest integer when it is larger than 10.

performs all other methods in terms of RA on all data sets, except for mnistP. RSP shows the highest LA on fashionP and mnistP. Only on mnistR, RSP performs worse than MER on RA, and worse than R-EWC on LA.

Compared only to EWC, RSP improves RA by 20% on the fashionP, and around 6% and 4% on notmnistP and mnistP, respectively. In terms of LA, both methods perform similarly on notmnistP and mnistR, whereas RSP shows substantial improvement on fashionP and mnistP. This result indicates that RSP performs better than EWC in encouraging positive forward transfer under the circumstances of limited memory. The gain in both LA and RA that our modification causes to EWC is accompanied by less negative backward transfer (BT) on all datasets. Under the setting adopted in this experiment, R-EWC performs similarly or slightly better than EWC, but it is still worse than RSP in most cases.

Related Work

Multi-Source Domain Adaptation (MSDA) Existing domain adaptation methods mainly focus on the single-source scenario. (Mansour, Mohri, and Rostamizadeh 2009) assumes that the target distribution can be approximated by a mixture of given source distributions, which also partially motivated our MDD. There are other theoretical analyses to the design of MSDA methods, with the purpose of either developing more accurate measures of domain discrepancy or deriving tighter generalization bounds (Redko et al. 2019; Zhao et al. 2020). Most existing bounds are based on the seminal work (Blitzer et al. 2007; Ben-David et al. 2010). For example, (Zhao, Zhang et al. 2018) extends the generalization bound in (Blitzer et al. 2007) to multiple sources. (Li et al. 2018) considered the relationship between pairwise sources and derived a tighter bound on weighted multi-source discrepancy based on a Wasserstein-like metric. Calculating such pairwise weights can be computationally demanding when the number of sources is large. Recently, (Wen, Greiner, and Schuurmans 2020) extends the upper-bound on the target domain loss, developed by (Cortes, Mohri, and Medina 2019), to MSDA. The new bound depends on the discrepancy distance between two domains (Mansour, Mohri, and Rostamizadeh 2009). (Richard et al. 2020) uses the hypothesis distance for regression (Cortes and Mohri 2014) and derives a similar bound.

Distinct from these methods, our discrepancy measure does not align the distribution of feature $p(\mathbf{t})$. Rather, it aims to match the dependence between $\mathbf{t}$ and y across domains, such that the conditional distributions $p(y|\mathbf{t})$ remain similar. To the best of our knowledge, we are also the first to derive a new generalization bound based on the matrix-based divergence (Kulis, Sustik, and Dhillon 2009; Yu et al. 2020).

Regularization-based Continual Learning and Network Modularizaton The general idea and popular regularization-based continual learning methods have been discussed in the previous section. Recently, network modularization is becoming a popular paradigm for efficient network training (Hadsell et al. 2020; Duan, Yu, and Príncipe 2021). Indeed, biological brains are modular, with distinct yet interacting subsystems. Introducing modularization to prevent forgetting dates back to (Pape et al. 2011) on the training of deep belief networks (DBN) (Hinton, Osindero, and Teh 2006). Recently, (Veniat, Denoyer, and Ranzato 2021) suggests a modular solution by identifying the trained modules (groups of neurons) to be re-used and extending the network with new modules for each new task.

Conclusion

We introduced von Neumann conditional divergence D_{vN} to align the dependence between latent representation $\mathbf{t}$ and response variable y across different domains and exemplified this idea in domain adaptation, assuming multiple source tasks are observed either simultaneously or sequentially. For the former, we consider multi-source domain adaptation (MSDA) and developed a new generalization bound as well as a new learning objective based on the loss induced by D_{vN}. For the latter, we focus on continual learning (CL) and demonstrated that such dependence can be formulated as a penalty to regularize the changes of network parameters. Empirical results justify the superiority of our methods.

Our point of departure is how learning, in general, can benefit from the conditional von Neumann divergence. At the same time, more than promoting a specific method, we aim at investigating a suitable distance measure for aligning representations. The perfect testbed for this is MSDA and CL. While the techniques we propose are deeply rooted and shaped by these domains, we hope them to be seen as an example of how the divergence can be beneficial.

Acknowledgments

This work is supported in part by the Research Council of Norway (RCN) under grant 309439.

References

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

Bertin-Mahieux, T.; Ellis, D. P.; Whitman, B.; and Lamere, P. 2011. The Million Song Dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Wortman, J. 2007. Learning bounds for domain adaptation. In *Conference on Neural Information Processing Systems, NeurIPS 2007*, 129–136.

Blitzer, J.; Dredze, M.; and Pereira, F. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Annual Meeting of the Association for Computational Linguistics, ACL 2007*, 440–447.

Chaudhry, A.; Dokania, P. K.; Ajanthan, T.; and Philip, P. H. 2018. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 532–547.

Chaudhry, A.; Ranzato, M.; Rohrbach, M.; and Elhoseiny, M. 2019. Efficient Lifelong Learning with A-GEM. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net.

Cortes, C.; and Mohri, M. 2014. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, 519: 103–126.

Cortes, C.; Mohri, M.; and Medina, A. M. 2019. Adaptation based on generalized discrepancy. *The Journal of Machine Learning Research*, 20(1): 1–30.

Delange, M.; Aljundi, R.; Masana, M.; Parisot, S.; Jia, X.; Leonardis, A.; Slabaugh, G.; and Tuytelaars, T. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Donahue, J.; Jia, Y.; et al. 2014. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning, ICML 2014*, 647–655.

Duan, S.; Yu, S.; and Príncipe, J. C. 2021. Modularizing deep learning via pairwise learning with kernels. *IEEE Transactions on Neural Networks and Learning Systems*.

Friedman, J. H. 1991. Multivariate adaptive regression splines. *The annals of statistics*, 1–67.

Ganin, Y.; Ustinova, E.; et al. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1): 2096–2030.

Graf, F.; Kriegel, H.-P.; Schubert, M.; Pölsterl, S.; and Cavallaro, A. 2011. 2d image registration in ct images using radial image descriptors. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 607–614. Springer.

Guerrero-Gómez-Olmedo, R.; Torre-Jiménez, B.; López-Sastre, R.; Maldonado-Bascón, S.; and Onoro-Rubio, D. 2015. Extremely overlapping vehicle counting. In *Iberian Conference on Pattern Recognition and Image Analysis*, 423–431. Springer.

Hadsell, R.; Rao, D.; Rusu, A. A.; and Pascanu, R. 2020. Embracing Change: Continual Learning in Deep Neural Networks. *Trends in Cognitive Sciences*.

Han, X.; Kashif, R.; and Roland, V. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv preprint*.

Hinton, G. E.; Osindero, S.; and Teh, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7): 1527–1554.

Kirkpatrick, J.; Pascanu, R.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521–3526.

Kulis, B.; Sustik, M. A.; and Dhillon, I. S. 2009. Low-Rank Kernel Learning with Bregman Matrix Divergences. *The Journal of Machine Learning Research*, 10(2).

Lake, B.; Salakhutdinov, R.; Gross, J.; and Tenenbaum, J. 2011. One shot learning of simple visual concepts. In *Proceedings of the annual meeting of the cognitive science society*.

Lempitsky, V.; and Zisserman, A. 2010. Learning to count objects in images. *Advances in neural information processing systems*, 23: 1324–1332.

Li, Y.; Murias, M.; Major, S.; Dawson, G.; and Carlson, D. E. 2018. Extracting relationships by multi-domain matching. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 6799–6810.

Liu, X.; et al. 2018. Rotate your networks: Better weight consolidation and less catastrophic forgetting. In *International Conference on Pattern Recognition (ICPR)*, 2262–2268. IEEE.

Lopez-Paz, D.; and Ranzato, M. 2017. Gradient episodic memory for continual learning. In *Conference on Neural Information Processing Systems, NeurIPS 2017*, 6467–6476.

Mansour, Y.; Mohri, M.; and Rostamizadeh, A. 2009. Domain Adaptation: Learning Bounds and Algorithms. In *COLT 2009 - The 22nd Conference on Learning Theory*.

Newell, A.; Yang, K.; and Deng, J. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, 483–499. Springer.

Nielsen, M. A.; and Chuang, I. 2002. *Quantum computation and quantum information*. American Association of Physics Teachers.

Pan, S. J.; Tsang, I. W.; Kwok, J. T.; and Yang, Q. 2010. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2): 199–210.

Pape, L.; Gomez, F.; Ring, M.; and Schmidhuber, J. 2011. Modular deep belief networks that do not forget. In *The 2011 International Joint Conference on Neural Networks*, 1191–1198. IEEE.

Parisi, G. I.; Kemker, R.; et al. 2019. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113: 54–71.

Park, J.; and Muandet, K. 2020. A measure-theoretic approach to kernel conditional mean embeddings. *Advances in Neural Information Processing Systems*, 33.

Pei, Z.; Cao, Z.; Long, M.; and Wang, J. 2018. Multi-adversarial domain adaptation. In *Conference on Artificial Intelligence, AAAI 2018*.

Pouyanfar, S.; Sadiq, S.; et al. 2018. A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)*, 51(5): 1–36.

Ramasesh, V. V.; Dyer, E.; and Raghu, M. 2020. Anatomy of catastrophic forgetting: Hidden representations and task semantics. *arXiv preprint arXiv:2007.07400*.

Redko, I.; Morvant, E.; Habrard, A.; Sebban, M.; and Bennani, Y. 2019. *Advances in domain adaptation theory*. Elsevier.

Ren, Y.; Zhu, J.; Li, J.; and Luo, Y. 2016. Conditional generative moment-matching networks. *Advances in Neural Information Processing Systems*, 29: 2928–2936.

Richard, G.; de Mathelin, A.; Hébrail, G.; Mougeot, M.; and Vayatis, N. 2020. Unsupervised Multi-Source Domain Adaptation for Regression. In *European Conference on Machine Learning, ECML 2020*.

Riemer, M.; Cases, I.; Ajemian, R.; Liu, M.; Rish, I.; Tu, Y.; and Tesauro, G. 2019. Learning to Learn without Forgetting by Maximizing Transfer and Minimizing Interference. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net.

Saito, K.; Kim, K.; et al. 2019. Semi-supervised domain adaptation via minimax entropy. In *IEEE International Conference on Computer Vision, ICCV 2019*, 8050–8058.

Veniat, T.; Denoyer, L.; and Ranzato, M. 2021. Efficient Continual Learning with Modular Networks and Task-Driven Priors. In *7th International Conference on Learning Representations, ICLR 2021*.

Wang, H.; Yang, W.; Lin, Z.; and Yu, Y. 2019. TMDA: Task-specific multi-source domain adaptation via clustering embedded adversarial training. In *2019 IEEE International Conference on Data Mining (ICDM)*, 1372–1377. IEEE.

Watanabe, C.; Hiramatsu, K.; and Kashino, K. 2018. Modular representation of layered neural networks. *Neural Networks*, 97: 62–73.

Wen, J.; Greiner, R.; and Schuurmans, D. 2020. Domain aggregation networks for multi-source domain adaptation. In *International Conference on Machine Learning*, 10214–10224. PMLR.

Yosinski, J.; Clune, J.; Bengio, Y.; and Lipson, H. 2014. How transferable are features in deep neural networks? In *Conference on Neural Information Processing Systems, NeurIPS 2014*, 3320–3328.

Yu, S.; Shaker, A.; Alesiani, F.; and Principe, J. C. 2020. Measuring the Discrepancy between Conditional Distributions: Methods, Properties and Applications. In *International Joint Conference on Artificial Intelligence, IJCAI 2020*, 2777–2784.

Zellinger, W.; Grubinger, T.; Lughofer, E.; Natschläger, T.; and Saminger-Platz, S. 2017. Central moment discrepancy (cmd) for domain-invariant representation learning. In *International Conference on Learning Representations, ICLR 2017*.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual Learning Through Synaptic Intelligence. In *International conference on machine learning, ICML 2017*, 3987–3995.

Zhao, H.; Combes, R. T. D.; Zhang, K.; and Gordon, G. 2019. On Learning Invariant Representations for Domain Adaptation. In *International conference on machine learning, ICML 2019*, 7523–7532.

Zhao, H.; Zhang, S.; et al. 2018. Adversarial multiple source domain adaptation. In *Conference on Neural Information Processing Systems, NeurIPS 2018*, volume 31, 8559–8570.

Zhao, S.; Gong, M.; et al. 2020. Domain Generalization via Entropy Regularization. In *Conference on Neural Information Processing Systems, NeurIPS 2020*, volume 33.

Zhao, S.; Li, B.; Xu, P.; and Keutzer, K. 2020. Multi-source domain adaptation in the deep learning era: A systematic survey. *arXiv preprint arXiv:2002.12169*.

Zhu, Y.; Zhuang, F.; and Wang, D. 2019. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 5989–5996.