ICML2020 Paper Review

Presenter: Zhe Wang

https://qdata.github.io/deep2Read

Zhe Wang

201909

Content

Few-shot domain adaptation by causal mechanism transfer

Takeshi Teshima, Issei Sato, Masashi Sugiyama

The University of Tokyo, RIKEN

Setup

Task: Homogeneous, multi-source, few-shot, supervised domain adaptation

- multi-source: labeled data from multiple source domains are available
- homogeneous: all source domains are in the same data space $\mathbb{R}^d$
- few-shot and supervised: in target domain, there are only a few labeled data available

Background:
In most existing work, the assumptions are relied on the similarity or small discrepancy of representation distributions $P(\Phi(X))$, or conditional distributions $P(\Phi(X)|Y), P(Y|\Phi(X))$.
These distribution based assumptions may not fit for transfer learning from apparently different distributions.

Assumption:
Sharing data generating mechanism (causal SEM) across different domains.

Intuition:
Human cares about causal knowledge, because once discovered, it applies to different systems.

Motivating Example:

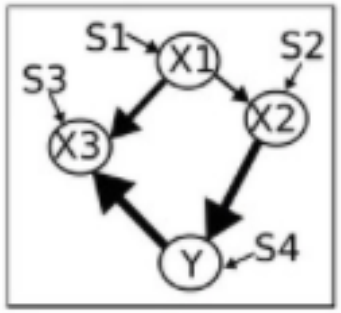
- Predict disease risk from medical records.
- Data distribution varies for different lifestyle.
- Common pathological mechanism across different regions.

SEM

SEM: The joint distribution can be factorized into the product of independent components.

$$P(X_1, X_2, \cdots, X_n) = \prod_{i=1}^{n} P(X_i | pa(X_i))$$

$$\begin{cases} X_1 &= f_1'(\mathrm{pa}_1, S_1) \\ X_2 &= f_2'(\mathrm{pa}_2, S_2) \\ X_3 &= f_3'(\mathrm{pa}_3, S_3) \\ Y &= f_4'(\mathrm{pa}_4, S_4) \end{cases}$$



Where S_i are independent variables.

Moreover, if the causal graph is acyclic, then it can be further reduced to:

$$\begin{cases} X_1 &= f_1'(\mathrm{pa}_1, S_1) \\ X_2 &= f_2'(\mathrm{pa}_2, S_2) \\ X_3 &= f_3'(\mathrm{pa}_3, S_3) \\ Y &= f_4'(\mathrm{pa}_4, S_4) \end{cases} \Rightarrow \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ Y \end{pmatrix} = f \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{pmatrix}$$

Structural equations Reduced form

The assumption of the paper is this structural equation f is shared across different domains.

Setup

Notation:

- input subspace $\mathcal{X} \in \mathbb{R}^{D-1}$,
- $\mathcal{Y} \in \mathbb{R}$,
- As a result, the overall data space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y} \in \mathbb{R}^D$, each labeled data is noted as $Z = (X, Y)$
- the set of distributions on $\mathbb{R}^D$ is noted as Q

Basic setup: p_{Tar} be a target distribution over $\mathcal{Z}$, $\mathcal{G}$ is a hypothesis set $G \subset \{g : \mathcal{R}^{D-1} \to R\}$, $l : \mathcal{G} \times \mathbb{R}^D \to [0, B_l]$ be the loss function. The goal is to find $g \in \mathcal{G}$ such that $R(g) = \mathbb{E}_{p_{Tar}} l(g, Z)$ is minimized.
Also suppose we have labeled data from K distinct source distributions $\{p_k\}_{k=1}^K$ over $\mathcal{Z}$, that is, we have iid samples $\mathcal{D}_k = \{Z_{k,i}^{Src}\}_{i=1}^{n_k} \sim p_k$

Key assumption

There exists a set of D dimensional IC distributions $q_{Tar}, q_k \in \mathcal{Q}$, and a smooth, invertible function $f : \mathbb{R}^D \rightarrow \mathbb{R}^D$ such that the data generation mechanism can be modeled as a two-step process:

- $S_{k,i}^{Src} \sim q_k$
- $Z_{k,i}^{Src} = f(S_{k,i}^{Src})$

And similarly for target domain: $Z_i = f(S_i), S_i \sim q_{Tar}$.

Benefits:

Now, no constraint on the similarity of the distribution enable the model to better accommodate intricate distribution shift.

Algorithm

Algorithm:

- Performing ICA on the labeled data from source domains to estimate the shared transformation $\hat{f} = ICA(\mathcal{D}_1, \cdots, \mathcal{D}_K)$
- Using the learned f to extract IC of the target domains $\hat{s}_i = \hat{f}^{-1}(Z_i), i = 1, 2, n_{Tar}$
- Data augmentation on the IC space, and get $\bar{s}_i$
- Synthesize more target data samples: $\bar{z}_i = \hat{f}(\bar{s}_i)$
- fit the predictor with augmented data of the target domain. $g^* = \arg\min_{g \in \mathcal{G}} l(g, \bar{z})$

Estimating f using source domain data:

Nonlinear ICA, specifically generalized contrastive learning(GCL), will be used to estimate f, based on the identification of nonlinear ICA, an auxiliary variable u also has to be observed. So the domain indices will be used to train a binary classifier:
the classification task to be trained in GCL is

$$r_{f,\phi}(z,u) = \sum_{d=1}^{D} \psi_d(\hat{f}^{-1}(z)_d, u),$$

consisting of $(\hat{f}, \{\psi_d\}_{d=1}^{D})$, the classification task of GCL is logistic regression to classify (Z_k^{Src}, k) as positive and $(Z_k^{Src}, k' \neq k)$ as negative.
domain contrastive learning criterion to estimate f:

$$\arg \min_{\hat{f},\{\psi_d\}_{d=1}^{D}} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i=1}^{n_k} (\phi(r_{\hat{f},\psi}(Z_{k,i}^{Src}, k)) + \mathbb{E}_{k' \neq k} \phi(-r_{\hat{f},\psi}((Z_{k,i}^{Src}, k'))))$$

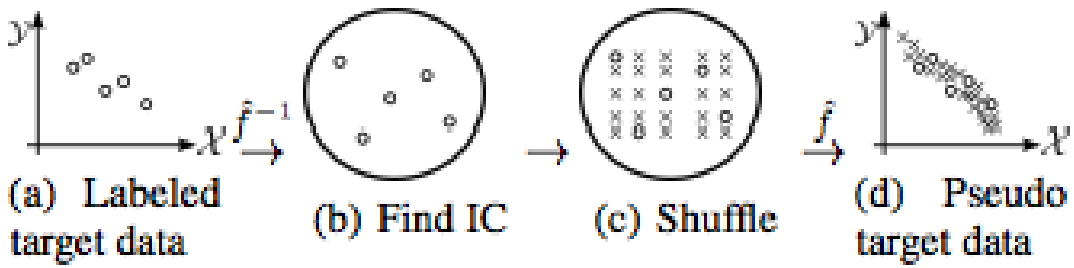
- Extract IC of the target domain:

$$\hat{s}_i = \hat{f}^{-1}(Z_i), |Z_i| = n_{Tar}$$

- Inflate the target IC

$$|\hat{s}| = n_{Tar},$$

where each $\hat{s}_i$ is of D dimension, every two dimensions are independent. To do augmentation, they inflate the set of IC values by taking all dimension-wise combinations of the estimated IC. Concretely, for each dimension, there are n_{Tar} choices, thus, the total number of augmented data is n_{Tar}^D



- Synthesis target data

$$\bar{z}_i = \hat{f}(\bar{s}_i), |\bar{z}| = n^D_{Tar}$$

- Fit a predictor with augmented data

$$R(g) = \frac{1}{n^D_{Tar}} \sum_i l(g, \bar{Z}) + \lambda ||g||^2$$

Experiment

Dataset

they use the gasoline consumption data, which is a panel data of gasoline usage in 18 countries over 19 years. Each country is considered as a domain.

The dataset contains four variables, and are widely-used for domain adapting regression tasks, especially for multi-source transfer learning.

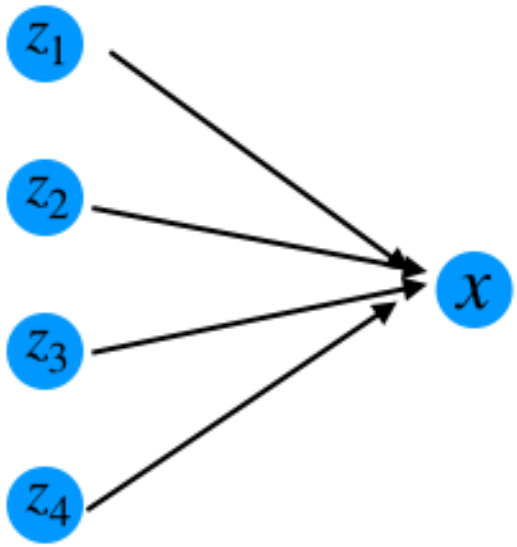
Target	(LOO)	TrgOnly	**Prop**	SrcOnly	S&TV	TrAda	GDM	Copula	IW(.0)	IW(.5)	IW(.95)
AUT	1	5.88 (1.60)	**5.39 (1.86)**	9.67 (0.57)	9.84 (0.62)	5.78 (2.15)	31.56 (1.39)	27.33 (0.77)	39.72 (0.74)	39.45 (0.72)	39.18 (0.76)
BEL	1	10.70 (7.50)	**7.94 (2.19)**	8.19 (0.68)	9.48 (0.91)	8.10 (1.88)	89.10 (4.12)	119.86 (2.64)	105.15 (2.96)	105.28 (2.95)	104.30 (2.95)
CAN	1	5.16 (1.36)	**3.84 (0.98)**	157.74 (8.83)	156.65 (10.69)	51.94 (30.06)	516.90 (4.45)	406.91 (1.59)	592.21 (1.87)	591.21 (1.84)	589.87 (1.91)
DNK	1	3.26 (0.61)	**3.23 (0.63)**	30.79 (0.93)	28.12 (1.67)	25.60 (13.11)	16.84 (0.85)	14.46 (0.79)	22.15 (1.10)	22.11 (1.10)	21.72 (1.07)
FRA	1	2.79 (1.10)	**1.92 (0.66)**	4.67 (0.41)	3.05 (0.11)	52.65 (25.83)	91.69 (1.34)	156.29 (1.96)	116.32 (1.27)	116.54 (1.25)	115.29 (1.28)
DEU	1	16.99 (8.04)	**6.71 (1.23)**	229.65 (9.13)	210.59 (14.99)	341.03 (157.80)	739.29 (11.81)	929.03 (4.85)	817.50 (4.60)	818.13 (4.55)	812.60 (4.57)
GRC	1	3.80 (2.21)	**3.55 (1.79)**	5.30 (0.90)	5.75 (0.68)	11.78 (2.36)	26.90 (1.89)	23.05 (0.53)	47.07 (1.92)	45.50 (1.82)	45.72 (2.00)
IRL	1	**3.05 (0.34)**	4.35 (1.25)	135.57 (5.64)	12.34 (0.58)	23.40 (17.50)	3.84 (0.22)	26.60 (0.59)	6.38 (0.13)	6.31 (0.14)	6.16 (0.13)
ITA	1	**13.00 (4.15)**	14.05 (4.81)	35.29 (1.83)	39.27 (2.52)	87.34 (24.05)	226.95 (11.14)	343.10 (10.04)	244.25 (8.50)	244.84 (8.58)	242.60 (8.46)
JPN	1	10.55 (4.67)	12.32 (4.95)	**8.10 (1.05)**	8.38 (1.07)	18.81 (4.59)	95.58 (7.89)	71.02 (5.08)	135.24 (13.57)	134.89 (13.50)	134.16 (13.43)
NLD	1	3.75 (0.80)	3.87 (0.79)	**0.99 (0.06)**	0.99 (0.05)	9.45 (1.43)	28.35 (1.62)	29.53 (1.58)	33.28 (1.78)	33.23 (1.77)	33.14 (1.77)
NOR	1	2.70 (0.51)	2.82 (0.73)	1.86 (0.29)	**1.63 (0.11)**	24.25 (12.50)	23.36 (0.88)	31.37 (1.17)	27.86 (0.94)	27.86 (0.93)	27.52 (0.91)
ESP	1	5.18 (1.05)	6.09 (1.53)	5.17 (1.14)	**4.29 (0.72)**	14.85 (4.20)	33.16 (6.99)	152.59 (6.19)	53.53 (2.47)	52.56 (2.42)	52.06 (2.40)
SWE	1	6.44 (2.66)	5.47 (2.63)	2.48 (0.23)	**2.02 (0.21)**	2.18 (0.25)	15.53 (2.59)	2706.85 (17.91)	118.46 (1.64)	118.23 (1.64)	118.27 (1.64)
CHE	1	3.51 (0.46)	**2.90 (0.37)**	43.59 (1.77)	7.48 (0.49)	38.32 (9.03)	8.43 (0.24)	29.71 (0.53)	9.72 (0.29)	9.71 (0.29)	9.79 (0.28)
TUR	1	1.65 (0.47)	1.06 (0.15)	1.22 (0.18)	**0.91 (0.09)**	2.19 (0.34)	64.26 (5.71)	142.84 (2.04)	159.79 (2.63)	157.89 (2.63)	157.13 (2.69)
GBR	1	5.95 (1.86)	**2.66 (0.57)**	15.92 (1.02)	10.05 (1.47)	7.57 (5.10)	50.04 (1.75)	68.70 (1.25)	70.98 (1.01)	70.87 (0.99)	69.72 (1.01)
USA	1	4.98 (1.96)	**1.60 (0.42)**	21.53 (3.30)	12.28 (2.52)	2.06 (0.47)	308.69 (5.20)	244.90 (1.82)	462.51 (2.14)	464.75 (2.08)	465.88 (2.16)
#Best	-	2	10	2	4	0	0	0	0	0	0

Challenging common assumptions in the unsupervised disentangled representations

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, Olivier Bachem

ETH Zurich, MaxPlanck Institute for Intelligent Systems, Google Brain

The key idea behind the unsupervised learning of disentangled representation is that real word is generated by a few explanatory factors of variation.



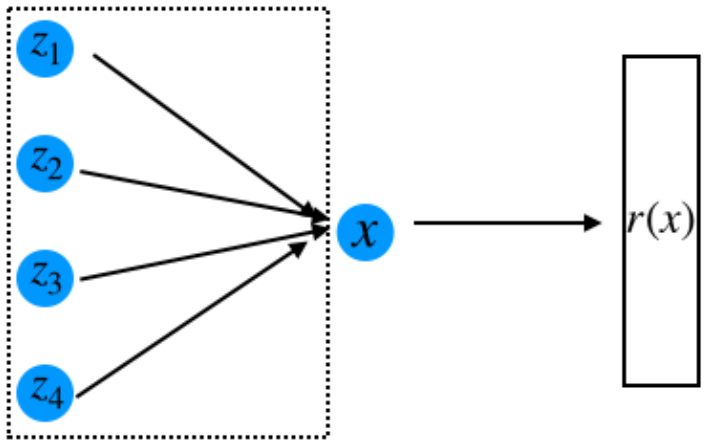
Data generating process:

- sample a multivariate latent random variable z from a distribution $P(z)$. Z corresponds to semantically meaningful factors of variation of the observations.
- The observation x is sampled from the conditional distribution $P(x|z)$

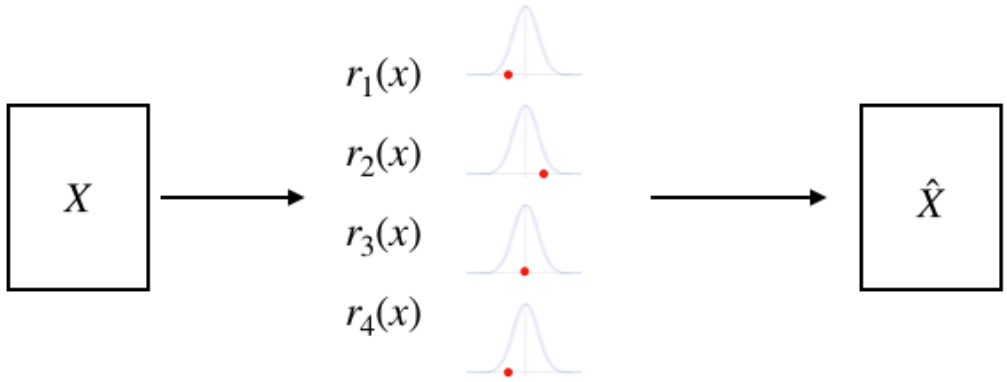
The goal of representation learning is to find $r(x)$, which satisfies:

- contain all the information present in x in a compact and interpretable structure,
- being independent from the task at hand,
- be useful for downstream tasks,
- enable to preform interventions and to answer counterfactual questions.

Learning of disentangled representations is an important step towards the goal.



This is no single formalized definition of disentanglement, the key intuition is that a disentangled representation should separate the distinct informative factors of variations in the data.
State-of-the-art approaches for unsupervised disentanglement learning are based on VAE:



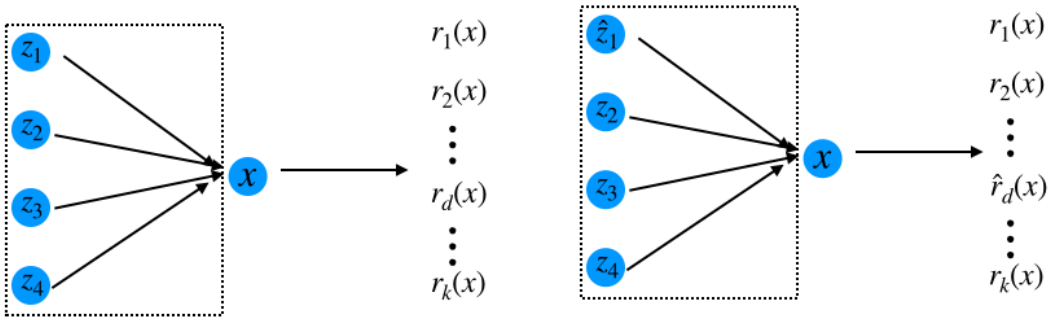
Assumptions of VAE:

- A specific prior $P(\mathrm{Z})$,
- use a DNN to parameterize the conditional probability $P(\mathrm{x}|\mathrm{z})$,
- the posterior is approximated by $Q(\mathrm{z}|\mathrm{x})$

The common theme is to enforce a factorized aggregated posterior $\int_x Q(\mathrm{z}|\mathrm{x})P(\mathrm{x})d\mathrm{x}$

In a similar spirit to disentanglement, ICA studies the problem of recovering independent components of a signal.
And the identification problem in ICA is a special case of disentanglement representation learning.

What is unsupervised disentanglement learning?



A change in a single ground-truth factor should lead to a single change in the representation.

Whether unsupervised disentanglement learning is possible?

Theorem 1. *For $d > 1$, let $(\mathbf{z}, \mathbf{x}) \sim P$ denote any generative model which admits a density $p(\mathbf{z}) = \prod_{i=1}^{d} p(\mathbf{z}_i)$ and where $\mathbf{z}$ denotes the independent latent variables and $\mathbf{x}$ the data observations. Then, there exists an infinite family of bijective functions $f : \mathrm{supp}(\mathbf{z}) \to \mathrm{supp}(\mathbf{z})$ such that $P(\mathbf{z} \leq \boldsymbol{u}) = P(f(\mathbf{z}) \leq \boldsymbol{u})$ for all $\boldsymbol{u} \in \mathrm{supp}(\mathbf{z})$ and $\frac{\partial f_i(\boldsymbol{u})}{\partial u_j} \neq 0$ almost everywhere for all i and j.*

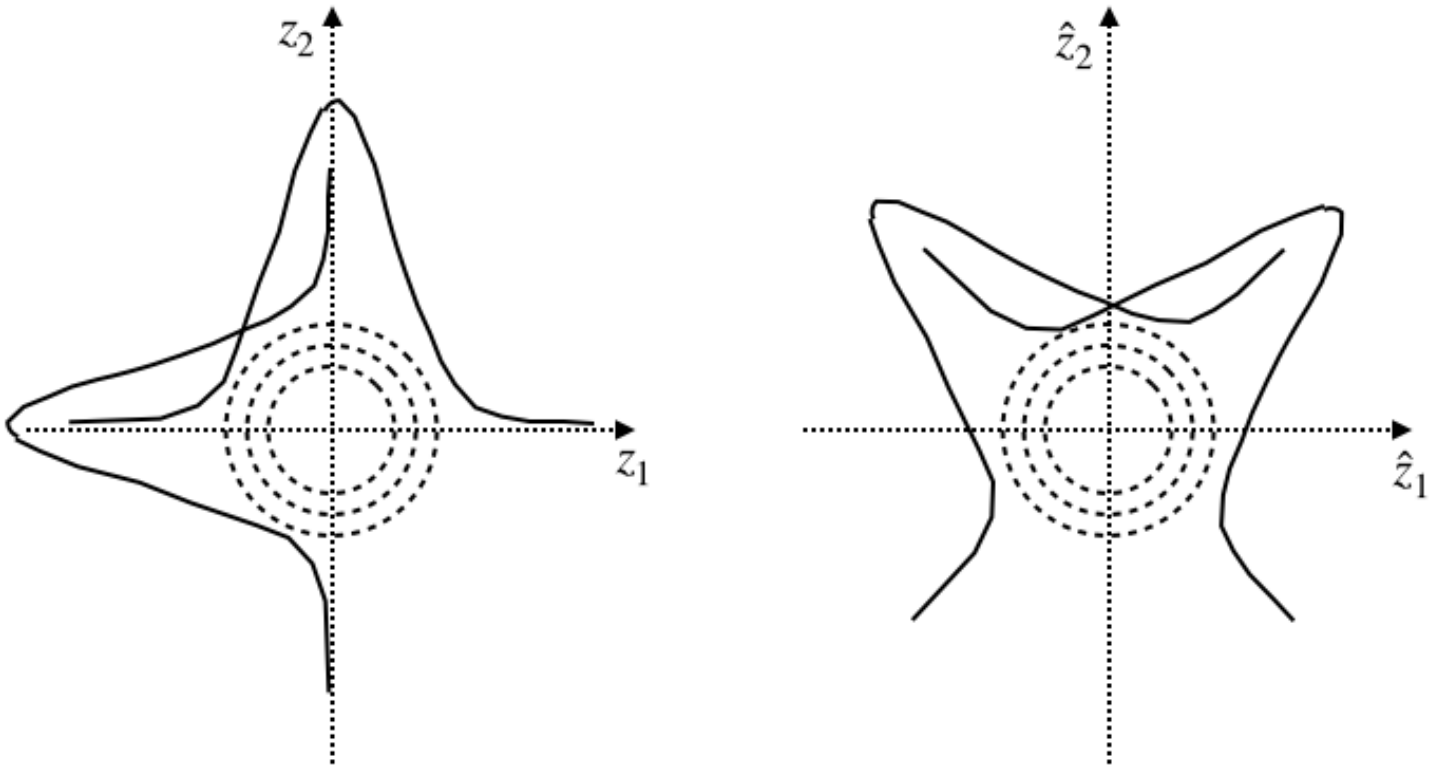
z and $f(z)$ are completely entangled and they have the same marginal distributions.

Conclusion: without inductive biases both on models and data sets, unsupervised disentanglement learning is impossible for arbitrary generative model with a factorized prior.

- Assume we have $p(z)$ and some $P(x|z)$ defining a generative model.
- Consider any unsupervised disentanglement method and assume that it finds a representation $r(x)$ that is perfectly disentangled with respect to z in the generative model.
- Theorem 1 implies that there is an equivalent generative model with the latent variable $\hat{z} = f(z)$ where $\hat{z}$ is completely entangled with respect to z and thus also $r(x)$.
- since f is deterministic and $p(z) = p(\hat{z})$ almost everywhere, both generative models have the same marginal distribution of the observations x by $p(x) = \int p(x|z)p(z)dz = \int p(x|\hat{z})p(\hat{z})dz$

Since the (unsupervised) disentanglement method only has access to observations x, it hence cannot distinguish between the two equivalent generative models and thus has to be entangled to at least one of them.

As a concrete example:



$$\begin{bmatrix}\hat{z}_1\\ \hat{z}_2\end{bmatrix} = \begin{bmatrix}cos(45), -sin(45)\\ sin(45), cos(45)\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix} \qquad p(z_1, z_2) = N(0,I) \quad \longrightarrow \quad p(\hat{z}_1, \hat{z}_2) = N(0,I)$$

In causality and ICA literature:

After observing x, we can construct infinitely many generative models which have the same marginal distribution of x. Any one of these models could be the true causal generative model for the data, and the right model cannot be identified given only the distribution of x

In practice:

- The theorem shows that unsupervised disentanglement learning is fundamentally impossible for arbitrary generative models, this does not necessarily mean it is an impossible endeavour in practice.
- After all, real world generative models may have a certain structure that could be exploited through suitably chosen inductive biases.
- Theorem clearly shows that inductive biases are required both for the models (so that we find a specific set of solutions) and for the data sets (such that these solutions match the true generative model)

Inductive biases: the strength of regularization strength, the choice of neural architecture, different random seed.

Results 1:

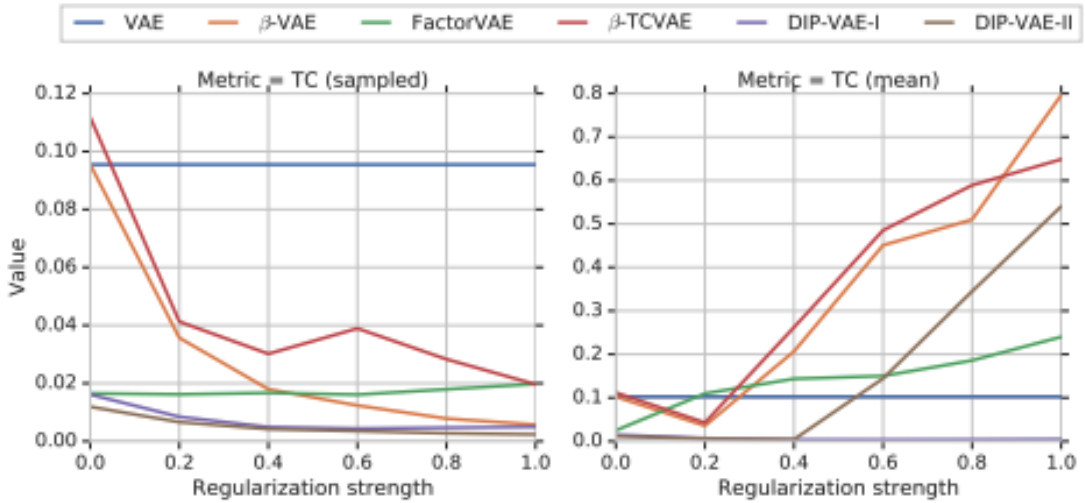


Figure 1. Total correlation based on a fitted Gaussian of the sampled (left) and the mean representation (right) plotted against regularization strength for Color-dSprites and approaches (except AnnealedVAE). The total correlation of the sampled representation decreases while the total correlation of the mean representation increases as the regularization strength is increased.

Implication:

The considered methods are effective at enforcing an aggregated posterior whose individual dimensions are not correlated but that this does not seem to imply that the dimensions of the mean representation (usually used for representation) are uncorrelated.

Results 2: how disentanglement is affected by the model choice, the hyperparameter selection and randomness?

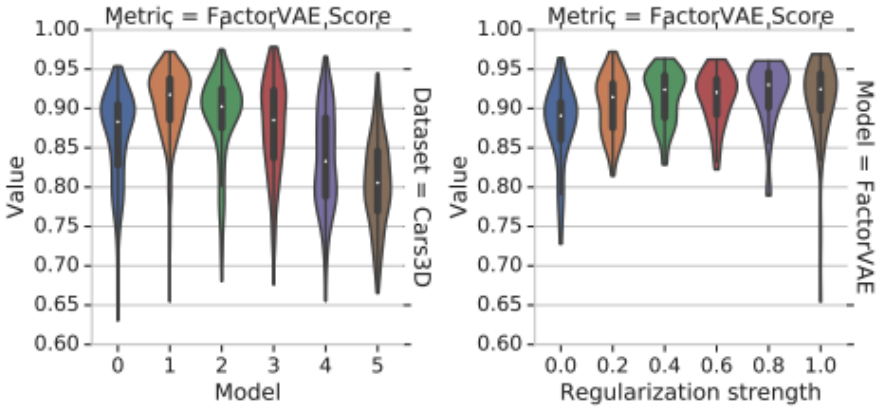


Figure 3. (left) FactorVAE score for each method on Cars3D. Models are abbreviated (0=β-VAE, 1=FactorVAE, 2=β-TCVAE, 3=DIP-VAE-I, 4=DIP-VAE-II, 5=AnnealedVAE). The variance is due to different hyperparameters and random seeds. The scores are heavily overlapping. (right) Distribution of FactorVAE scores for FactorVAE model for different regularization strengths on Cars3D. In this case, the variance is only due to the different random seeds. We observe that randomness (in the form of different random seeds) has a substantial impact on the attained result and that a good run with a bad hyperparameter can beat a bad run with a good hyperparameter.

Implication:

The disentanglement scores of unsupervised models are heavily influenced by randomness (in the form of the random seed) and the choice of the hyperparameter (in the form of the regularization strength).

Results 3: Are these disentangled representations useful for downstream tasks in terms of the sample complexity of learning?

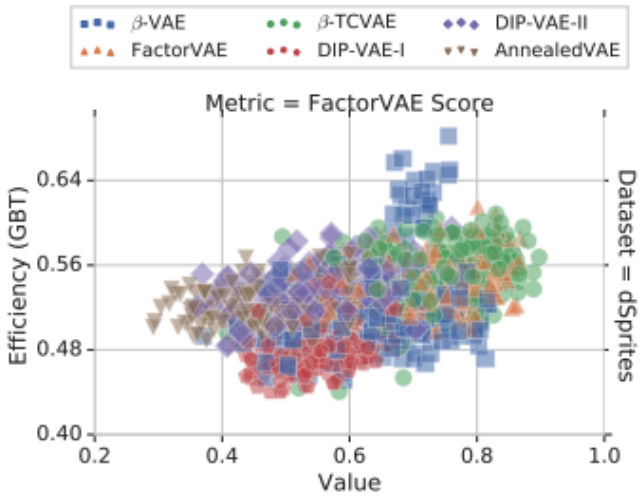


Figure 6. Statistical efficiency of the FactorVAE Score for learning a GBT downstream task on dSprites.

Implication:
There is no clear evidence that disentangled representations will be useful for downstream tasks, but there are many more potential notions of usefulness such as interpretability and fairness that we have not considered in our experimental evaluation.

Suggestions:

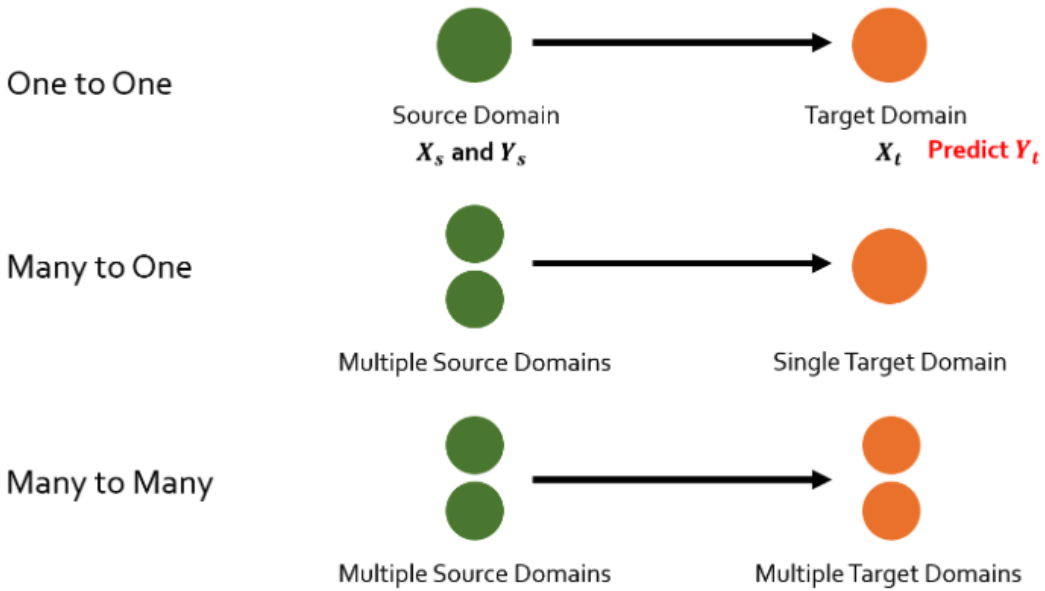
- The role of inductive biases and implicit and explicit supervision should be made explicit: unsupervised model selection persists as a key question.
- The concrete practical benefits of enforcing a specific notion of disentanglement of the learned representations should be demonstrated.
- Experiments should be conducted in a reproducible experimental setup on data sets of varying degrees of difficulty.

Continuously indexed domain adaptation

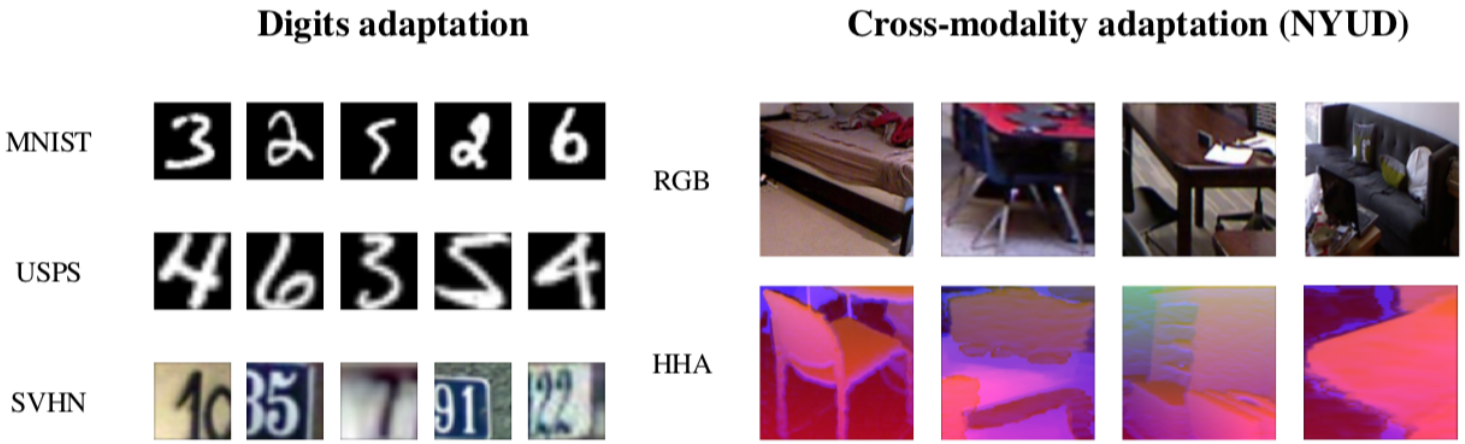
Hao Wang, Hao He, Dina Katabi

MIT CSAIL

Motivation:
Existing domain adaptation focus on transferring knowledge between domains with categorical indices: such as $A \rightarrow B$, or if multi-source domains are available, then the domain adaptation is between $A_1, A_2, \cdots, A_n \rightarrow B$



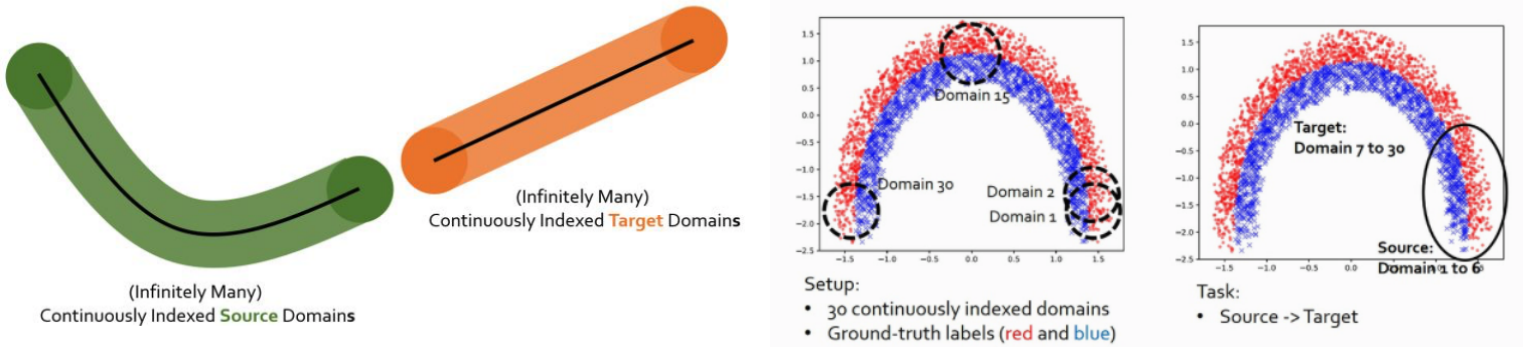
For example, the most frequently used two datasets for domain adaptation are MNIST and SVHN, either adpating from MNIST to SVHN or adapting from SVHN to MNIST.



There are many tasks involve continuous indexed domains.

Motivating Example 1:

In medical applications, one needs to do transfer learning across patients of different ages.

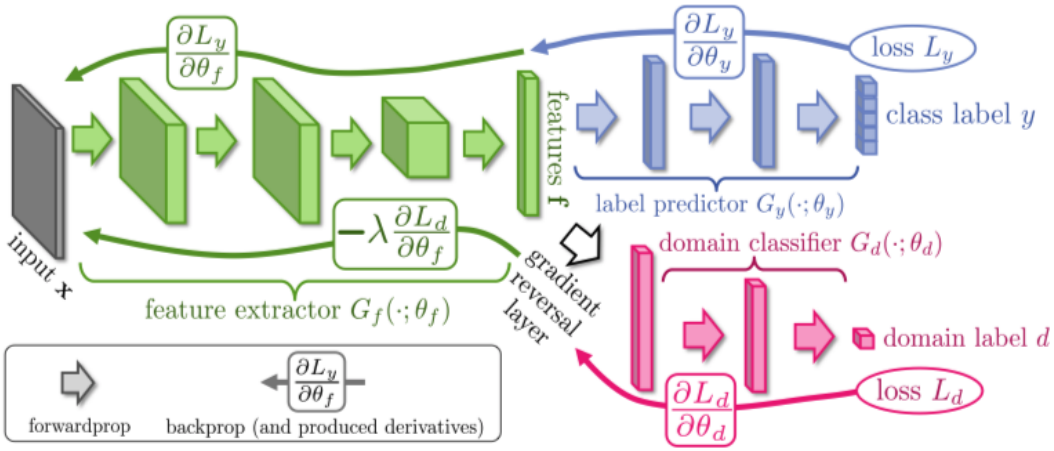


Motivating Example 2:

Underwater robots have to operate at different water depths and viscosity, and one expects that adaptation across datasets from different depths or viscosity (e.g., lake vs. sea) should take into account the relationship between the robot operation and the physical properties of the liquid in which it operates.

A direct method is treating the age of the source and target domain as domain labels, but this is unlikely to yield the optimal result, since it doesn't consider the distance between different domains.
In other word, if the distance of domain indices are close $d(u_1, u_2)$, the joint distributions $P(y_{u_1}, x_{u_1}), P(y_{u_2}, x_{u_2})$ are also similar.

How do existing methods work?



The role of discriminator: binary classifier determines whether the data comes from source domain 1, or from target domain 0.

Setup

Task: unsupervised domain adaption, which means we only have labeled data in the source domain and unlabeled data in the target domain.

- We have a set of continuous domain indices: $\mathcal{U} = \mathcal{U}_s \cup \mathcal{U}_t$, also $\mathcal{U}$ is a metric space.
- In source domains whose indices are $u_i^s \in \mathcal{U}_s$, we have labeled data $\{(x_i^s, y_i^s, u_i^s)\}_{i=1}^n$
- In target domains whose indices are $u_i^t \in \mathcal{U}_t$, we have unlabeled data $\{(x_i^t, u_i^t)\}_{i=1}^m$, the goal is to predict $\{(y_i^t)\}_{i=1}^m$ for data in the target domains.

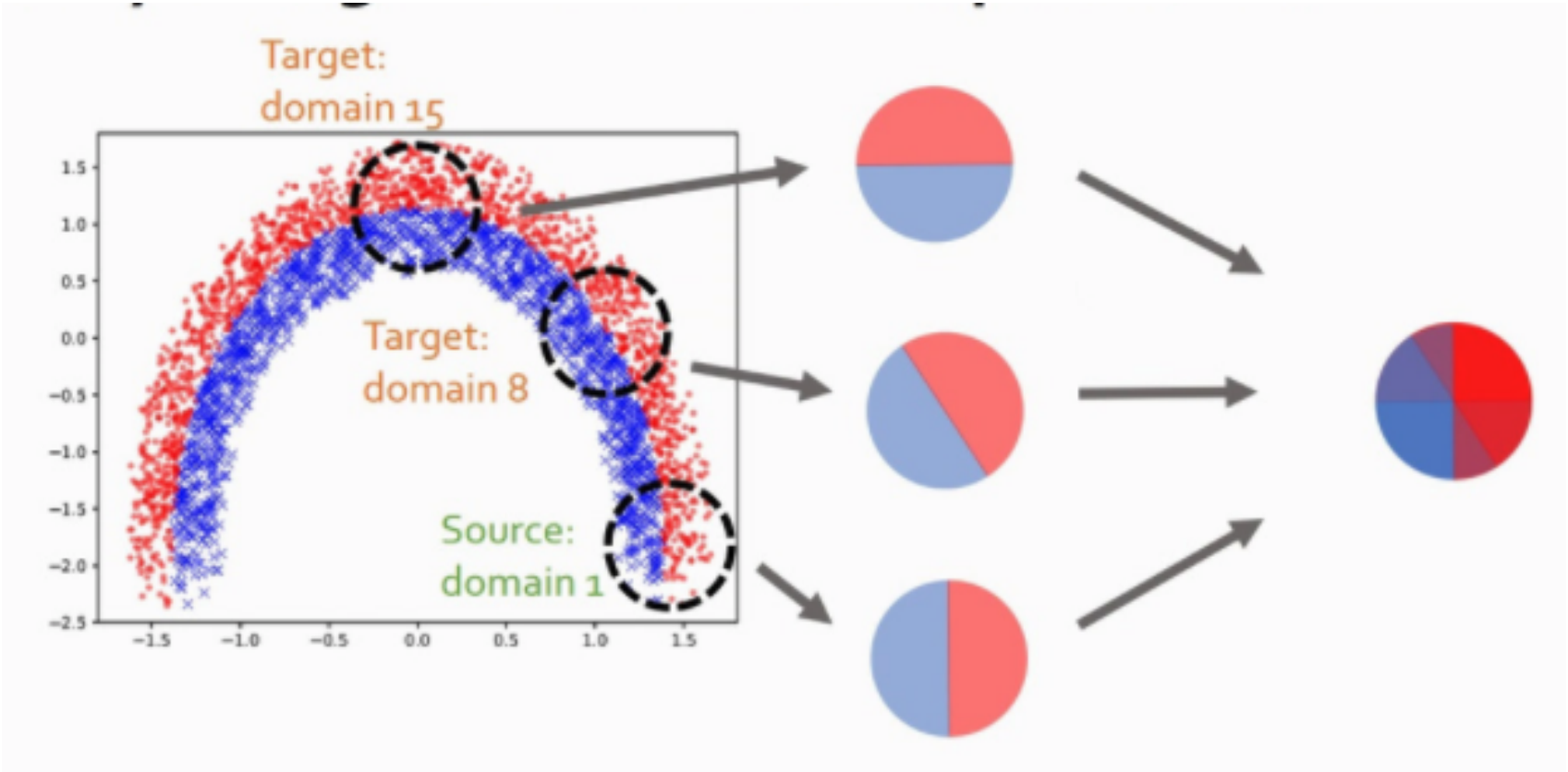
Proposed method

Main idea: Learn an encoder E and predictor F such that distribution of encoding $z = E(x, u)$ from all domains $\mathcal{U}$ are aligned.
Formally, domain invariant encodings require that

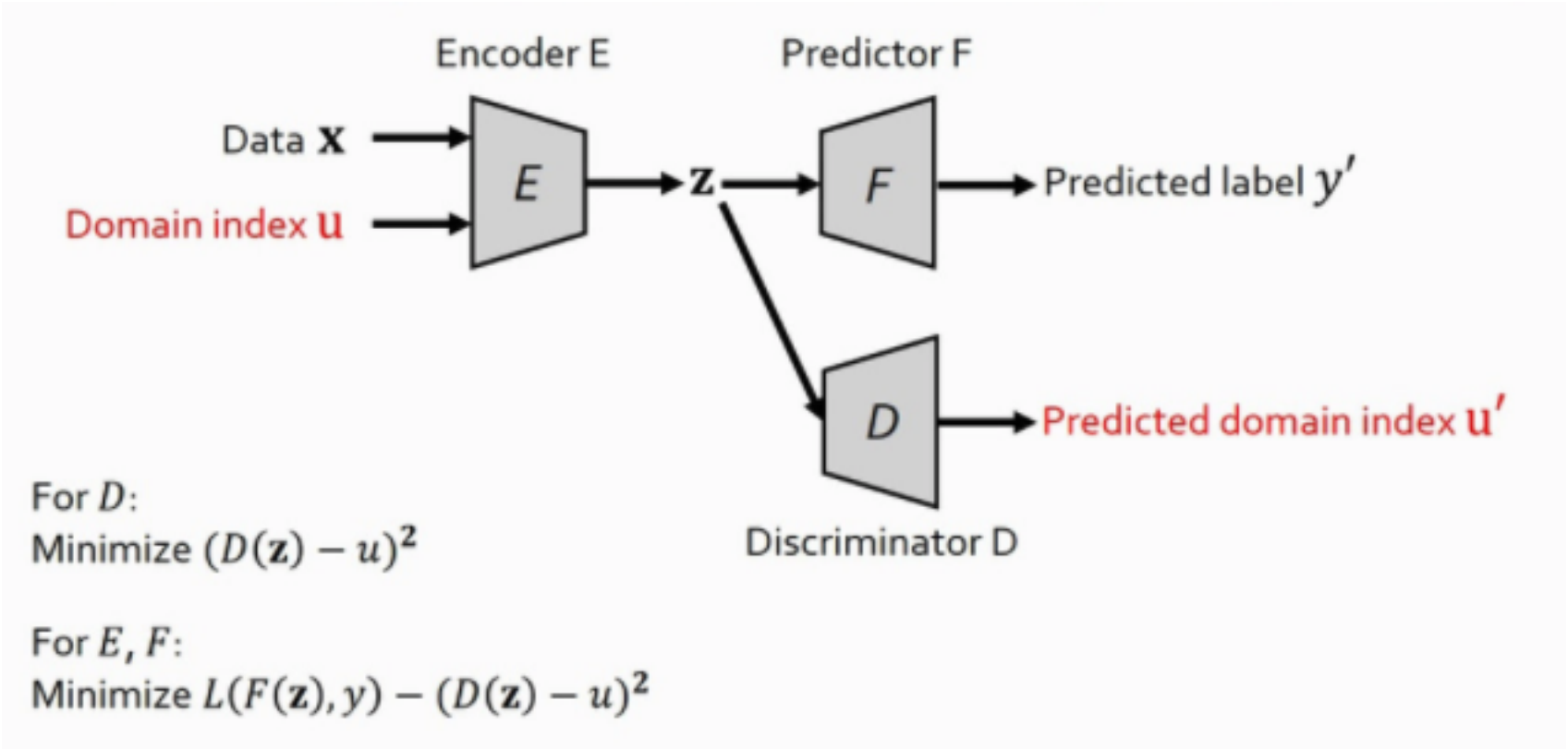
$$p(z|u_1) = p(z|u_2) \text{ or } p(u_1|z) = p(u_2|z), \forall u_1, u_2 \in \mathcal{U},$$

this is achieved with the help of a discriminator D.

In continuously indexed domains, there is an underlying assumption, similar domain indices implies similar encoding.



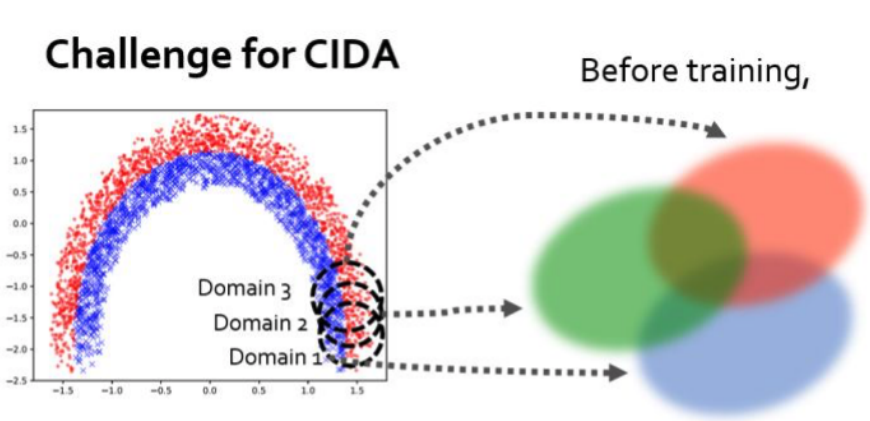
CIDA

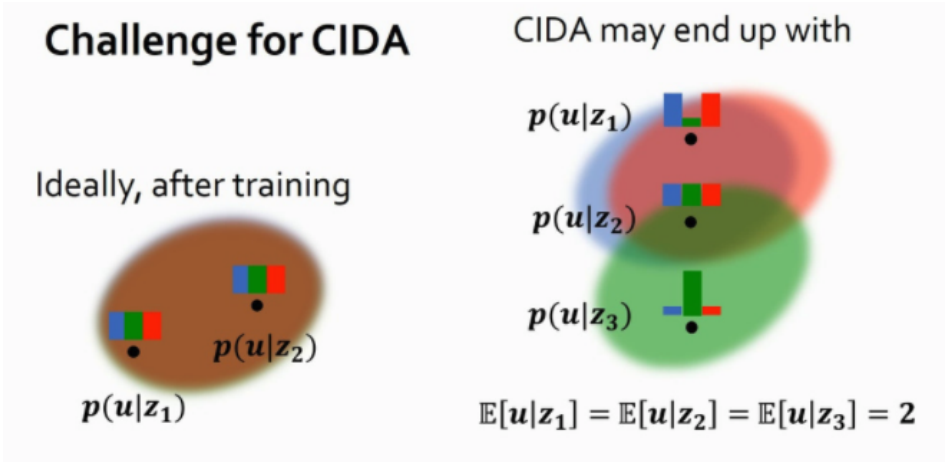


In vanilla CIDA, the discriminator D is designed to regress the domain index.

Probabilisitc CIDA

Challenges for vanilla CIDA:





It is only able to match the expectation which is the first order moments. In order to match both mean and variance of the distributions $p(u|z)$, they propose a variant called Probabilistic CIDA.

Probabilistic CIDA

In probabilistic CIDA, the discriminator predicts the distribution of $p(u|z)$ instead of providing point estimation, specifically, it outputs both the mean and covariance of $p(u|z)$ as $D_\mu(z)$ and $D_{\sigma^2}(z)$

the loss function of the D:

$$L_d(D(z), u) = \frac{(D_\mu(z) - u)}{2D_{\sigma^2}(z)} + \frac{1}{2}\log D_{\sigma^2}(z)$$

Theoretical analysis

Informal statement:

- CIDA converges, if and only if, the expectation of the domain index $\mathbb{E}[u|z]$ is identical for any embedding z.
- PCIDA converges, if and only if, the expectation and the variance of the domain index $\mathbb{E}[u|z]$ and $\mathbb{V}[u|z]$ is identical for any embedding z.

Results

simulated dataset 1

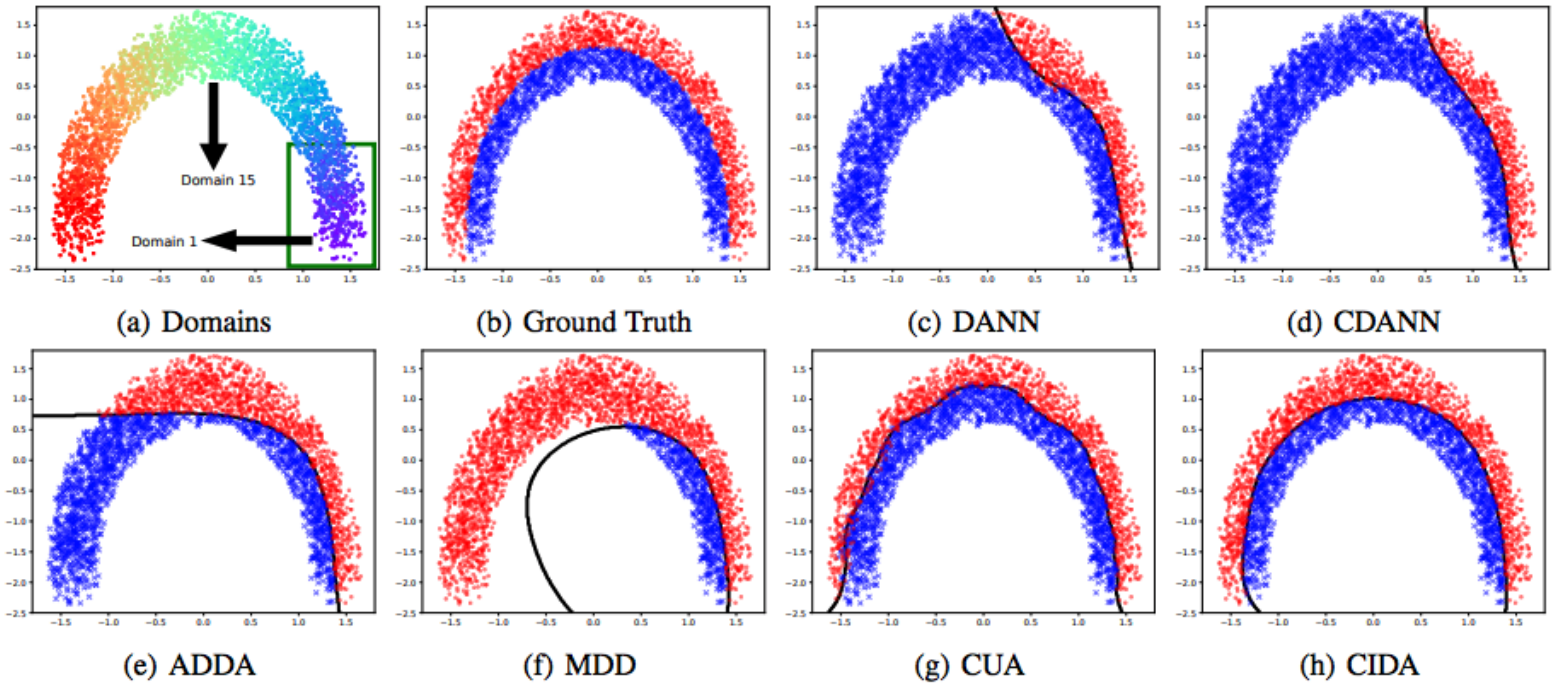


Figure 1. Results on the *Circle* dataset with 30 domains. Fig. 1(a) shows domain index by color. The first 6 domains are source domains, marked by green boxes. Red dots and blue crosses are positive and negative data samples. Black lines show the decision boundaries generated according to model predictions.

Results

simulated dataset 2

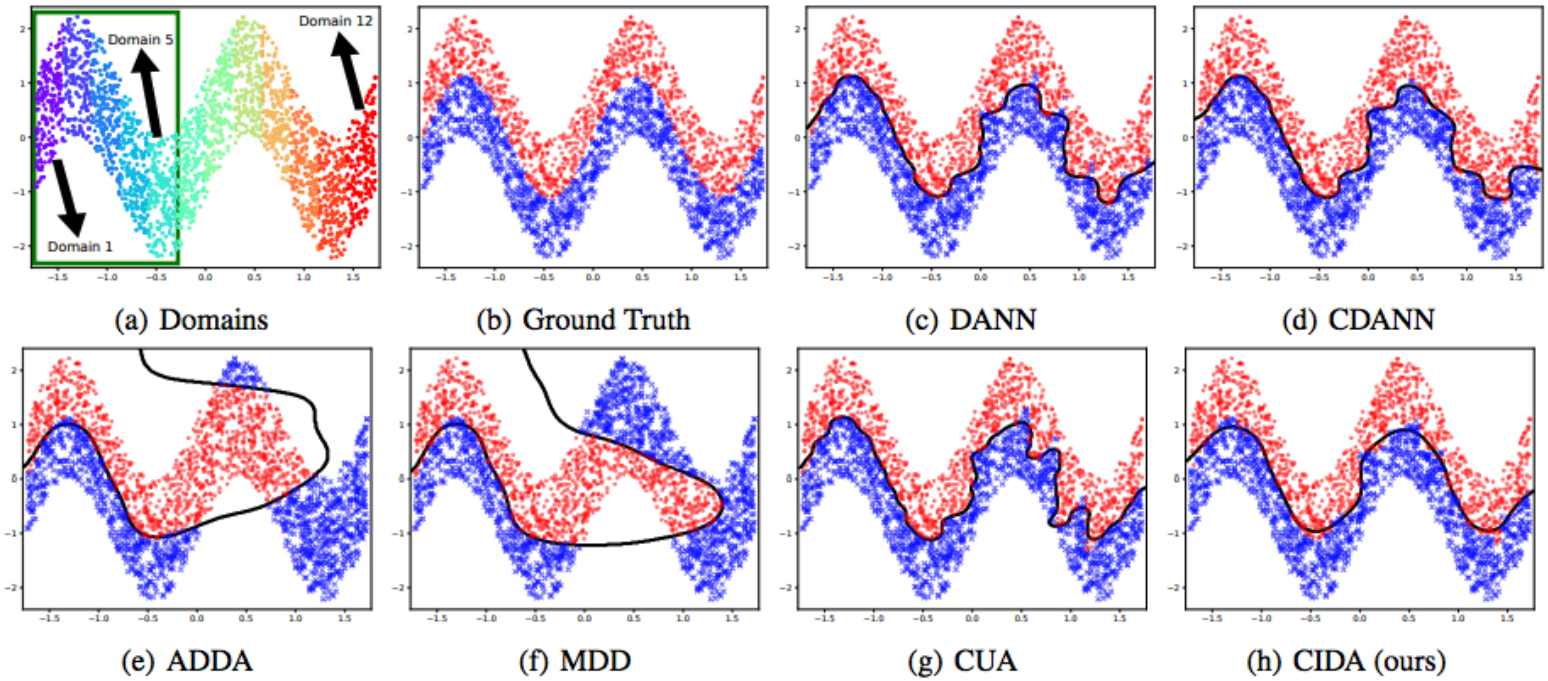


Figure 2. Results on the *Sine* dataset with 12 domains. The first 5 domains are source domains marked by green boxes. Red dots and blue crosses are positive and negative data samples. Black lines show the decision boundaries generated according to model predictions.

Rotating MNIST

The goal is to adapt from regular MNIST digits with mild rotation to significantly rotated MNIST digits.

Table 1. ***Rotating MNIST*** **accuracy (%) for various adaptation methods.** We report the accuracy at the source domain and each target domain. X° denotes the domain whose images are rotated by X° to $X + 45^\circ$. The last column shows the average accuracy across target domains. We use **bold face** to mark the best results.

Method	# Target Domains	0° (Source)	45°	90°	135°	180°	225°	270°	315°	Average
Source-Only	-	99.0	79.1	44.0	44.1	46.8	32.7	29.0	77.8	50.5
ADDA	1	97.0	72.7	39.6	42.1	43.8	30.4	26.9	77.0	47.5
DANN	1	98.5	76.0	38.1	45.5	46.6	34.7	30.7	67.0	48.4
CUA	7	91.7	**92.6**	92.2	89.5	72.5	65.6	67.8	69.3	78.5
CIDA (Ours)	∞	96.5	91.8	92.3	94.5	**93.9**	92.5	93.2	95.8	93.4
PCIDA (Ours)	∞	96.6	92.2	**92.8**	**94.9**	**93.9**	**92.7**	**93.6**	**95.9**	**93.7**

Healthcare Dataset

They use three medical datasets, Sleep Heart Health Study (SHHS), MultiEthnic Study of Atherosclerosis (MESA) and Study of Osteoporotic Fractures (SOF). Each dataset contains full-night breathing signals of subjects and the corresponding sleep stage labels (‘Awake’, ’Light Sleep’, ‘Deep Sleep’, and ‘Rapid Eye Movement (REM)’).

Intra-Dataset Adaptaion

Intra-dataset adaptation

Table 2. **Accuracy (%) for intra-dataset adaptation.** '*SHHS*@Outside → *SHHS*@(52,75]' means transferring from age range outside (52,75] (i.e., [44,52]∪(75,90]) to (52,75] within *SHHS*. 'SO' is short for 'Source-Only'. We use **bold face** mark the best results.

	Task	SO	ADDA	DANN	CDANN	MDD	CUA	CIDA	PCIDA
Domain Extrapolation	*SHHS*@[44,52] → *SHHS*@(52,90]	77.4	78.0	77.1	77.5	77.7	77.4	79.8	**80.6**
	MESA@[54,58] → *MESA*@(58,95]	80.1	80.7	79.9	80.4	80.3	80.1	**82.7**	82.5
	SOF@[75,82] → *SOF*@(82,90]	74.7	74.8	74.2	74.4	74.6	74.5	**76.7**	**76.7**
Domain Interpolation	*SHHS*@Outside → *SHHS*@(52,75]	82.4	81.7	82.5	82.3	82.5	82.4	82.2	**83.7**
	MESA@Outside → *MESA*@(58,75]	83.5	83.5	83.2	83.3	83.8	83.4	83.5	**84.7**
	SOF@Outside → *SOF*@(79,86]	71.8	71.5	71.4	70.9	71.8	71.5	71.8	**73.6**

Cross-dataset adaptation

Table 3. **Accuracy (%) for cross-dataset adaptation.** We use **bold face** to mark the best results.

Task	Source-Only	ADDA	DANN	CDANN	MDD	CUA	CIDA	PCIDA
SOF → *SHHS*	75.6	76.0	75.2	75.6	75.8	75.3	75.9	**80.1**
SOF → *MESA*	74.0	75.1	74.6	75.2	74.9	73.6	74.8	**80.0**
SHHS → *MESA*	82.8	83.0	82.6	82.1	83.0	82.1	83.2	**85.3**
MESA → *SHHS*	80.7	81.8	80.9	80.9	81.2	81.0	80.8	**83.4**
SHHS → *SOF*	78.7	79.5	79.0	79.2	79.7	79.1	**81.1**	80.9
MESA → *SOF*	75.9	76.6	77.0	76.9	76.9	76.0	**79.3**	79.0