Supplementary Materials:
Continuously Indexed Domain Adaptation

1 Proof

Lemma 1.1 (Uniqueness of Constant Expectation). *$\mathbf{z}$ and u are random variables. If $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u]$ is constant w.r.t $\mathbf{z}$, then $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u] = \mathbb{E}_{u\sim p(u)}[u], \forall \mathbf{z}$.*

Proof. Let $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u] = \mu, \forall\, \mathbf{z}$. We then have $\mathbb{E}_{u\sim p(u)}[u] = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\mathbb{E}_{u\sim p(u|\mathbf{z})}[u] = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\mu = \mu$. □

Lemma 1.2 (Uniqueness of Constant Expectation and Variance). *$\mathbf{z}$ and u are random variables. If $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u]$ and $\mathbb{V}_{u\sim p(u|\mathbf{z})}[u]$ are constants w.r.t $\mathbf{z}$, then $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u] = \mathbb{E}_{u\sim p(u)}[u]$ and $\mathbb{V}_{u\sim p(u|\mathbf{z})}[u] = \mathbb{V}_{u\sim p(u)}[u]$ for any $\mathbf{z}$.*

Proof. Let $\mathbb{E}_{u\sim p(u|\mathbf{z})}[u] = \mu$ and $\mathbb{V}_{u\sim p(u|\mathbf{z})}[u] = \sigma^2$ for any $\mathbf{z}$. By the previous lemma, we have $\mathbb{E}_{u\sim p(u)}[u] = \mu$. For the variance, we have:

$$\begin{aligned}\mathbb{V}_{u\sim p(u)}[u] &= \mathbb{E}_{u\sim p(u)}[(u-\mathbb{E}[u])^2] = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\mathbb{E}_{u\sim p(u|\mathbf{z})}[(u-\mathbb{E}[u|\mathbf{z}])^2]\\ &= \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\mathbb{V}_{u\sim p(u|\mathbf{z})}[u] = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\sigma^2 = \sigma^2,\end{aligned}$$

concluding the proof. □

Lemma 1.3 (Optimal Discriminator for PCIDA). *With E fixed, the optimal D is*

$$\begin{aligned}D^*_{\mu,E}(\mathbf{z}) &= \mathbb{E}_{u\sim p(u|\mathbf{z})}[u],\\ D^*_{\sigma^2,E}(\mathbf{z}) &= \mathbb{V}_{u\sim p(u|\mathbf{z})}[u],\end{aligned}$$

where $\mathbf{z} = E(\mathbf{x}, u)$.

Proof. The optimal D:

$$D^*_E(\mathbf{x}, u) = \underset{D}{\operatorname{argmin}}\, \mathbb{E}_{(\mathbf{z},u)\sim p(\mathbf{z},u)}[L_d(D(\mathbf{z}), u)],$$

where the objective function expands to

$$\begin{aligned}&\mathbb{E}_{(\mathbf{z},u)\sim p(\mathbf{z},u)}[L_d((D_\mu(\mathbf{z}), D_{\sigma^2}(\mathbf{z})), u)]\\ =&\mathbb{E}_{(\mathbf{z},u)\sim p(\mathbf{z},u)}[\frac{(D_\mu(\mathbf{z})-u)^2}{2D_{\sigma^2}(\mathbf{z})} + \frac{1}{2}\log D_{\sigma^2}(\mathbf{z})]\\ =&\mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\mathbb{E}_{u\sim p(u|\mathbf{z})}[\frac{(D_\mu(\mathbf{z})-u)^2}{2D_{\sigma^2}(\mathbf{z})} + \frac{1}{2}\log D_{\sigma^2}(\mathbf{z})].\end{aligned}$$

Notice that

$$\begin{aligned}&\mathbb{E}_{u\sim p(u|\mathbf{z})}[\frac{(D_\mu(\mathbf{z})-u)^2}{2D_{\sigma^2}(\mathbf{z})} + \frac{1}{2}\log D_{\sigma^2}(\mathbf{z})]\\ =&\frac{\mathbb{E}[u^2|\mathbf{z}]}{2D_{\sigma^2}(\mathbf{z})} - \frac{D_\mu(\mathbf{z})}{D_{\sigma^2}(\mathbf{z})}\mathbb{E}[u|\mathbf{z}] + \frac{D_\mu(\mathbf{z})^2}{2D_{\sigma^2}(\mathbf{z})} + \frac{1}{2}\log D_{\sigma^2}(\mathbf{z}).\end{aligned}$$

Taking the derivative w.r.t. $D(\mathbf{z})$ and setting it to 0, we get the optimal $D^*_{\mu,E}(\mathbf{z}) = \mathbb{E}[u|\mathbf{z}]$ and $D^*_{\sigma^2,E}(\mathbf{z}) = \mathbb{V}[u|\mathbf{z}]$, completing the proof. □

Table 1: 11 domain indices in the *SHHS* dataset.

u_1	Age
u_2	Resting heart rate
u_3	Gender
u_4	Physical functioning
u_5	Role limitation due to physical health
u_6	General health
u_7	Role limitation due to emotional problems
u_8	Energy/fatigue
u_9	Emotional well being
u_{10}	Social functioning
u_{11}	Pain Level

Table 2: Network structure for the encoder.

Kernel	Stride	Channel In	Channel Middle	Channel Out	Type	Number
13	2	1	-	64	Conv	1
9	1	64	64	64	ResBlock	1
9	2	64	64	128	ResBlock	1
9	1	128	128	128	ResBlock	1
7	1	128	128	256	ResBlock	1
9	5	256	256	256	ResBlock	1
5	1	256	256	512	ResBlock	1
5	1	512	512	512	ResBlock	1
5	1	512	384	384	ResBlock	1
9	5	384	384	384	ResBlock	1
3	1	384	384	384	ResBlock	1
5	1	384	384	384	ResBlock	1

2 Experiments

In this section we provide more details for our experiments. The code is available at https://github.com/hehaodele/CIDA.

2.1 Experiment on the Healthcare Datasets

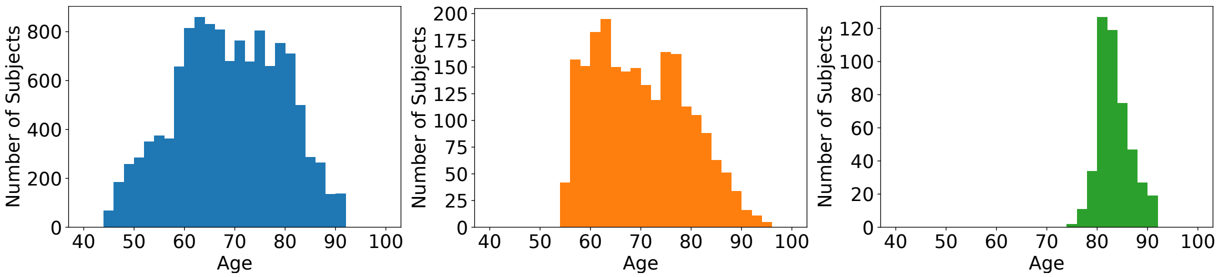
Dataset details. The three real-world medical datasets [7, 8, 2] with detailed information are publicly available[1]. They can all be freely accessed upon request and submission of relevant IRB documents. In Fig. 1 we plot the histograms subjects' age in the three medical datasets. All the three datasets contains many health retaled variables of the subjects. In Table 1, we list all the variables we considered as the domain indices.

Implementations. We use the same neural network architecture in all methods for fair comparison. Table 2 shows the neural network architecture for the encoders taking breathing signals $\mathbf{x}$ as input. 'Number' in the tables indicates the number of corresponding blocks stacked in the network. The predictor includes 3 fully connected layers, each with batch normalization and ReLU. Similarly, the discriminator includes 5 fully connected layers. For the baseline models, we explore different λ_d (the hyperparameter for the discriminator term) in the range $\{0.2, 0.5, 1.0, 2.0, 5.0\}$ and find that $\lambda_d = 2.0$ produce stable and the best results in the toy datasets. We follow recommendations from the original papers for other hyperparameters. We set $\lambda_d = 2.0$ for all methods including CIDA/PCIDA. We train all models using the Adam optimizer [5] with a learning rate of 10^{-4}. We run all experiments on a server with four NVIDIA Titan Xp GPUs.

2.2 Experiment on the Rotating MNIST

Dataset details. In our *Rotating MNIST*, there are 60,000 images in each domain index interval spanning 45 degrees from $[0°, 45°)$ to $[315°, 360°)$. They are generated by rotating each of the 60,000 image in the MNIST training set by a

[1] https://sleepdata.org/



(a) SHHS (b) MESA (c) SOF

Figure 1: Age histograms for three medical datasets.

angle randomly sampled the corresponding domain index interval. Therefore, this dataset contains images with rotation angles evenly spread in the range $[0, 360°)$. We note that this is different from the *Rotating MNIST* dataset in [1], where the images are Rotating by 8 fixed angles. Another difference is that in our Rotating MNIST, the amount of data in target domains is 7 times as many as that in source domain while in [1], the target domain has the same amount of data as the source domain.

Implementations. We use the same neural network architecture in all methods for fair comparison. Mainly, we use a four-layer convolutional neural networks to encode the image and a three-layer MLP to make the prediction, while the discriminator is a four-layer MLP. In addition, we make two augmentations to provide the model with a stronger inductive bias. First, we add a Spacial Transfer Network (STN) [4] to the image encoder. Basically, the STN will take the image and the domain index as input and output a set of rotation parameters which are then applied to rotate the given image. Second, we add the dropout layers to the STN and the ConvNet backbone. As mentioned in [3], dropout can be viewed as a way of performing Bayesian inference. Here, we use this dropout switch to make image encoder either deterministic or probabilistic. For more details, please refer to our code.

2.3 Experiments on the Toy Datasets

Visualization of the decision boundary (approximately). Unlike shallow models such as logistic regression, plotting deep neural networks' exact decision boundaries is not straightforward. To generate a virtual decision boundary for visualization, we fit an SVM with the RBF kernel by neural networks' prediction and draw the decision boundary of the SVM. To be fair, when fitting the SVM, we ensure that the fitting accuracy is the same for all deep learning models. Note that since the generated boundaries are not exact, we can observe some data points on the wrong side of the boundaries.

3 Discussion

3.1 Categorical Domains versus Continuously Indexed Domains

Continuous Indices. As mentioned in the main paper, the hypothesis of 'continuous indices' is that input $\mathbf{x}$ and labels y are drawn from $p(\mathbf{x}, y|u)$ given a specific domain index $u \in \mathcal{U}$, and that $p(\mathbf{x}, y|u)$ (and $p(y|\mathbf{x}, u)$) is continuous w.r.t. u. Therefore, CIDA tries to produce correct predictions in a continuous range of target domains by effectively capturing the underlying relation (functional) between $p(y|\mathbf{x}, u)$ and u.

Distance Metrics. Such a hypothesis comes with a distance metric for domain indices, which are captured by the regression loss (e.g., euclidean distance for L_2 loss) in the discriminator. This is a key difference between CIDA and categorical domain adaptation, where any pair of domains effectively has the same distance. This is also true for categorical domain adaptation methods such as [6]. Note that [6] uses a least-square loss as a surrogate for cross-entropy to perform domain *classification* in the discriminator, therefore still treating different domains as equal. This is substantially different from CIDA where the L_2 loss and the Gaussian (or Gaussian Mixture Model) loss are use to regress the domain indices.

3.2 Matching $p(u|\mathbf{z})$ versus Matching $p(\mathbf{z}|u)$

In general, matching the entire $p(u|\mathbf{z})$ for any $\mathbf{z}$ is equivalent to matching the entire $p(\mathbf{z}|u)$ for any u. This is because $p(u|\mathbf{z}) = p(u) \iff p(\mathbf{z}|u) = p(\mathbf{z}) \iff u \perp\!\!\!\perp \mathbf{z}$. However, matching the mean and variance of $p(u|\mathbf{z})$ for any $\mathbf{z}$ is **different** from matching the mean and variance of $p(\mathbf{z}|u)$ for any u. Considering the dimension of $\mathbf{z}$ is much higher than that of u, the former is actually **stronger** alignment.

To see this, consider a simplified case where $\mathbf{z} \in \{1, 2, 3, 4\}^{100}$ and $u \in \{1, 2, 3, 4\}$. Matching the mean and variance of $p(u|\mathbf{z})$ requires matching the mean and variance of 4^{100} univariate distributions, i.e., 2×4^{100} parameters in total. On the other hand, Matching the mean and variance of $p(\mathbf{z}|u)$ only requires matching the mean and variance of 4 100-dimensional distributions, i.e., 400 parameters in total. Therefore the former implies stronger alignment.

References

[1] Andreea Bobu, Eric Tzeng, Judy Hoffman, and Trevor Darrell. Adapting to continuously shifting domains. 2018.

[2] Steven R Cummings, Dennis M Black, Michael C Nevitt, Warren S Browner, Jane A Cauley, Harry K Genant, Stephen R Mascioli, Jean C Scott, Dana G Seeley, Peter Steiger, et al. Appendicular bone density and age predict hip fracture in women. *JAMA*, 263(5):665–668, 1990.

[3] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016.

[4] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[6] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *ICCV*, pages 2813–2821, 2017.

[7] Stuart F Quan, Barbara V Howard, Conrad Iber, James P Kiley, F Javier Nieto, George T O'connor, David M Rapoport, Susan Redline, John Robbins, Jonathan M Samet, et al. The sleep heart health study: design, rationale, and methods. *Sleep*, 20(12):1077–1085, 1997.

[8] Guo-Qiang Zhang, Licong Cui, Remo Mueller, Shiqiang Tao, Matthew Kim, Michael Rueschman, Sara Mariani, Daniel Mobley, and Susan Redline. The national sleep research resource: towards a sleep data commons. *JAMA*, 25(10):1351–1358, 2018.