MACQUARIE University
SYDNEY·AUSTRALIA

Deep Non-IID Learning

IJCAI 2023 Tutorial

Zhilin Zhao and Longbing Cao

August 20, 2023

Non-IID learning: https://datasciences.org/publications/non-iid-learning
Shallow and Deep Non-IID Learning on Complex Data

Contents

Independent Identically Distributed (IID)

- Data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ is composed of N samples that are independently drawn from the same joint distribution $p(\mathbf{x}, y)$, i.e.,

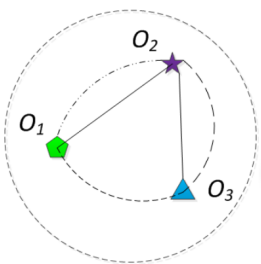
$$(\mathbf{x}_i, y_i) \sim p(\mathbf{x}, y).$$

- A learning algorithm is built to learn

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}, y)}{p(\mathbf{x})}.$$

Classic Statistical Assumption - IIDness & IID Learning

IID learning:
Dominates classic analytics,
AI/KDD/ML/CVPR/Statistics research & development

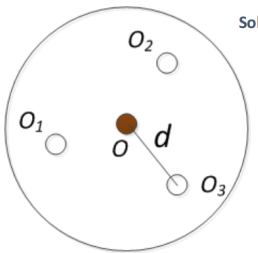


Problem

Assumption

IIDness:

Independence +
Identical Distribution
O_1, O_2, O_3 are IID



Solution

O_1, O_2, O_3 are IID:
An independent O:
$d_3 = ||O_3 - O||$

Discriminative Learning with IID Assumption

- Learn a posteriori distribution $p(y|\mathbf{x})$
- Model:
  - e.g., Classification and Clustering models
- Assumption:
  - $\mathbf{x}_i \perp \mathbf{x}_j$
  - $p(y_i|\mathbf{x}_i)$

Generative Learning with IID Assumption

- Learn the joint probability $p(\mathbf{x}, y)$
  - Learning $p(\mathbf{x}|y)$ with $p(y)$
  - Bayes' theorem: $p(y|\mathbf{x}) = p(\mathbf{x}|y)p(y)/p(\mathbf{x})$
- Models:
  - e.g., generators
- Assumption:
  - $\mathbf{x}_i \perp \mathbf{x}_j$
  - $y_i \perp y_j$

Examples: IID Distance Measures and Functions

- Samples are IID.
- Variables are random
  - Euclidean distance: $d(\mathbf{x}_1, \mathbf{x}_2) = \|\mathbf{x}_1 - \mathbf{x}_2\|$
  - Hamming distance: $d(\mathbf{s}_1, \mathbf{s}_2) = \sum_{i=1}^{M} \delta(\mathbf{s}_1[i], \mathbf{s}_2[i])$
  - Mahalanobis distance: $d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T \mathbf{S}^{-1} (\mathbf{x}_1 - \mathbf{x}_2))}$

Questions & Thinking: What if samples are dependent and follow different distributions?

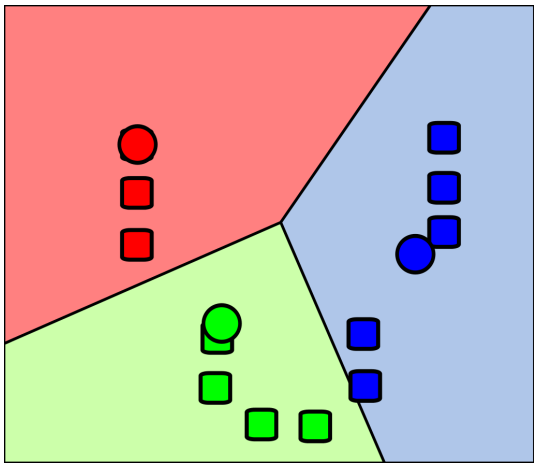
Statistics of IID Data

- Variance of samples: $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \boldsymbol{\mu})^2$
- Covariance of variables: $\text{cov}(\mathbf{x}, \mathbf{y}) = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \boldsymbol{\mu_x})(\mathbf{y}_i - \boldsymbol{\mu_y})$
- Cross entropy: $\mathcal{H}(p, q) = -\int_{\mathcal{X}} p(x)\log q(x)dx$
- KL-divergence / Relative entropy: $\mathcal{D}(p||q) = \mathcal{H}(p, q) - \mathcal{H}(p)$

Questions & Thinking: What if samples and distributions are dependent?

Example: IID K-means

- **K-means:**
  - Target: $\arg\min_{\mathbf{S}} \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in \mathbf{S}_k} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|$
  - $\mathbf{x}_i$ is a individual sample.
  - $\mathbf{S}_k$ is a individual cluster.
- **What Makes K-means IID?**
  - Sample IIDness: all samples are independent.
  - Cluster IIDness: all clusters are independent.
  - Global to local: Global partition $\rightarrow$ local distribution



Example: IID Decision Tree

- **Objective functions:**
  - $(\mathbf{x}, y) = (x_1, x_2, x_3, \cdots, x_k, y)$
  -

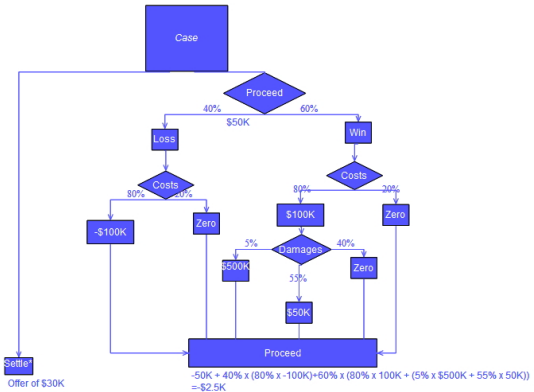
$$\underbrace{E_A(IG(T,a))}_{\text{Expected Information Gain}} = \underbrace{I(T;A)}_{\text{Mutual Information between } T \text{ and } A} = \underbrace{H(T)}_{\text{Entropy (parent)}} - \underbrace{H(T|A)}_{\text{Weighted Sum of Entropy (Children)}}$$

$$= -\sum_{i=1}^{K} p_i \log_2 p_i - \sum_{a} p(a) \sum_{i=1}^{K} -\Pr(i|a) \log_2 \Pr(i|a).$$

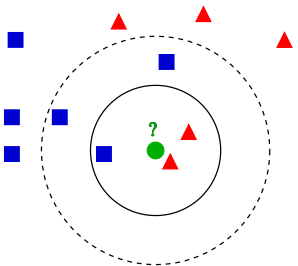
- Note: (1) T: The data set, (2) A: An attribute, (3) a: A value of A, (4) $\mathbf{x}$: a sample, (5) y: a label, (6) K: The number of classes, (7) p_i : the probability of class i, and (8) p_a: the probability of value a.

Example: IID Decision Tree

- **Questions & Thinking:**
  - What if objects x_k and x_j are dependent?
  - What if values a_1 and a_2 are dependent?
  - What if classes i_1 and i_2 have different distributions?



Example: IID KNN

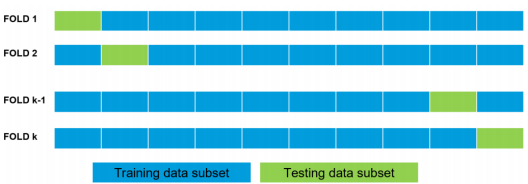


Questions & Thinking:

- What if samples are dependent?
- What if neighbors are dependent?
- What if samples are drawn from different distributions?

Example: IID K-fold Cross Validation & Sampling, Batching

Randomly sample k-folds



Questions & Thinking:

- What if the samples in the data are non-IID?
- What if the samples in the training set are non-IID?
- What if the samples in the training and the test sets are non-IID? i.e., OOD problem

Potential Risk of IID Learning

- **Results delivered by IID learning on non-IID data could be:**
  - incomplete
  - partial characterization
  - biased
  - misleading
- **Many 'benchmarks' may be unfair and wrong.**
- **Questions & Thinking:**
  - Why does learning bias exist?
  - Beyond fitting issues, what other issues may have caused learning bias?

Contents

Independent Identically Distributed (IID)

- Data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ is composed of N samples that are independently drawn from the same joint distribution $p(\mathbf{x}, y)$, i.e.,

$$(\mathbf{x}_i, y_i) \sim p(\mathbf{x}, y).$$

- A learning algorithm is built to learn

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}, y)}{p(\mathbf{x})}.$$

- Question:
  - Learning $p(y|\mathbf{x})$ in terms of $p(y_i|\mathbf{x}_i)$ on each sample $\mathbf{x}_i$.
  - What if $(\mathbf{x}_i, y_i)$ and $(\mathbf{x}_j, y_j)$ are coupled (non-independent)?
  - What if $(\mathbf{x}_i, y_i) \sim p_i(\mathbf{x}, y)$ and $(\mathbf{x}_j, y_j) \sim p_j(\mathbf{x}, y)$ are coupled (non-identically distributed)?

Independent Identically Distributed (IID)

- $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$ is d-dimensional vector.
- What if features x_i and x_j $(i, j \in [D])$ are not independent?
- What if features x_i and x_j $(i, j \in [D])$ are not identically distributed? i.e., $p_i(x)$ and $p_j(x)$ are different?
- What if label classes y_i and y_j $(i, j \in [K])$ are dependent?
- What if label classes y_i and y_j $(i, j \in [K])$ follow different distributions $p_i(y)$ and $p_j(y)$?

Beyond Statistical IID: Non-IIDness

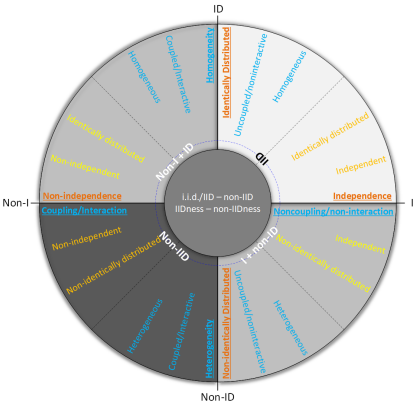
- **Statistical IID:** Independence + Identical Distribution
- **Non-IID case:** variables and their data hold non-independence and non-identical distribution.
- **Non-independence** expands to diverse interactions, couplings, and entanglement (interaction for short).
- **Non-identical distribution** expands to comprehensive heterogeneities.
- **Heterogeneities and interactions** go beyond statistical IID and form the general non-IIDness [1] [2] [3].

[1] L. Cao. Beyond i.i.d.: Non-IID thinking, informatics, and learning. IEEE Intell. Syst., 37(4), pp. 5–17, 2022.

[2] L. Cao. Non-IIDness Learning in Behavioral and Social Data, The Computer Journal, 57(9), pp. 1358-1370, 2014.

[3] L. Cao. Coupling Learning of Complex Interactions, IP&M, 51(2), pp. 167-186, 2015.

Beyond IID: IID to Non-IID Space



Two perspectives:

- Statistical independence and distribution
- Beyond statistics interactions and heterogeneities

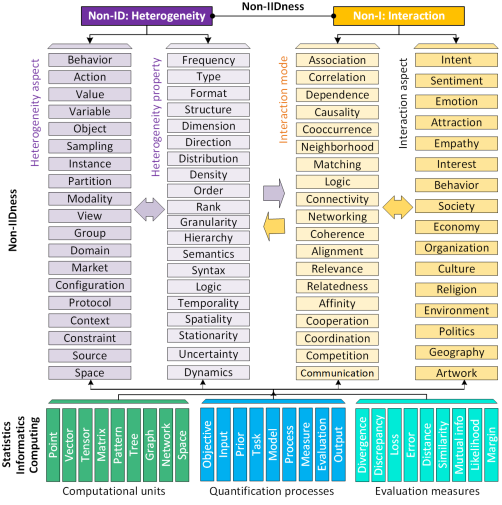
Four quadrants:

- IID
- Non-I
- I + Non-ID
- Non-IID

Concept of Non-IIDness

- **Heterogeneities and interactions** form the general non-IIDnesses.
- **Heterogeneities**
  - Aspects: behavior, action, value, variable, object, partition, modality, view, source, etc.
  - Properties: frequency, type, format, structure, dimension, direction, distribution, etc.
- **Interactions**
  - Within and between values, attributes, objects, sources, aspects, $\cdots$
  - Structures, distributions, relations, $\cdots$
  - Methods, models, $\cdots$
  - Results, targets, impact, $\cdots$

Aspects of Non-IIDness



The terminology and conceptual map of non-IIDness beyond statistical IID:

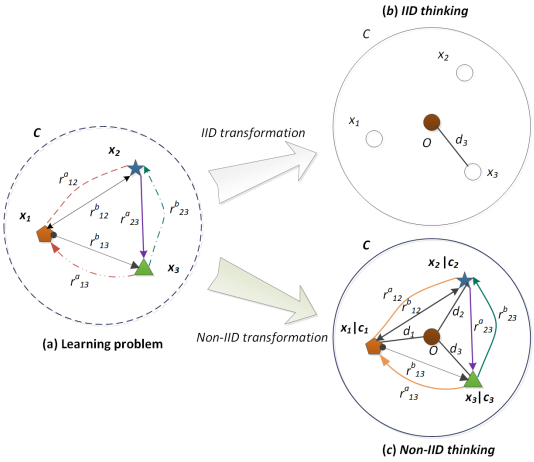
- Non-ID - heterogeneities
- Non-I - interactions

Interactions vs. General Relations [4]

- **Types:** numerical, categorical, textual, mixed structure, syntactic, semantic, organizational, social, cultural, economic, uncertain, unknown/latent relation, etc.
- Interaction goes beyond existing relations including dependence, correlation, association, and causality.
- Mathematically, association, causality, correlation, and dependence are specific, descriptive, explicit, etc.
- **Interactions:** explicit + implicit, qualitative + quantitative, descriptive + deep, specific + comprehensive, local + global, etc.

[4] C. Wang, F. Giannotti, and L. Cao. Learning Complex Couplings and Interactions. IEEE Intell. Syst. 36(1), pp. 3-5, 2021.

IID Thinking vs. Non-IID Thinking [5]



- **IID thinking** transforms a complex system to be IID.
- **Non-IID thinking** transforms the problem to be non-IID, where non-IIDnesses are characterized and incorporated into the problem-solving system.

[5] https://datasciences.org/non-iid-learning/

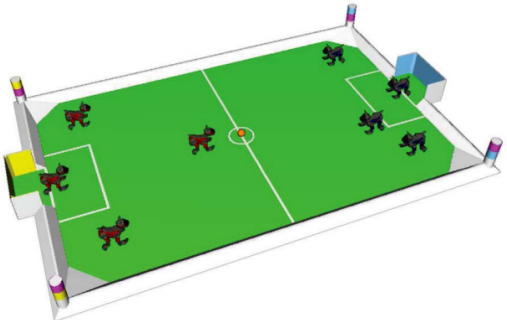
Quantifying Heterogeneity and Interaction

- **Quantifying and incorporating** heterogeneity and interaction into non-IID frameworks.
- **Quantifying Heterogeneities:**
  - quantifying heterogeneous objects (e.g., formats and distributions)
  - quantifying heterogeneity properties (e.g., features, granularity)
  - formulating heterogeneity aspects in terms of their quantified properties (e.g., types and dynamics of features)
- **Quantifying Interactions:**
  - mathematical relation learning/modeling (e.g., dependence)
  - deep interaction modeling and learning (by deep latent relations)
  - coupling learning (e.g., coupled object similarity learning [6] and unsupervised heterogeneous coupling learning)

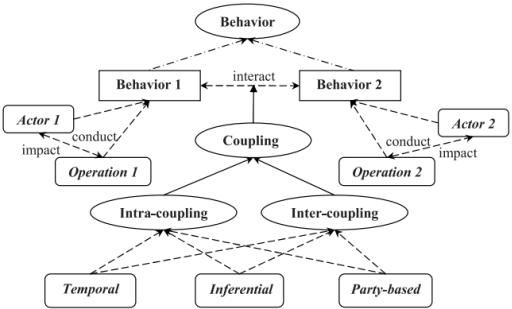
[6] C. Wang, X. Dong, F. Zhou, L. Cao, C. Chi, Coupled Attribute Similarity Learning on Categorical Data, IEEE Transactions on Neural Networks and Learning Systems, 26(4): 781-797, 2015.

Example: Group Behavior Interactions [7]

Behavior interactions in a group are often associated with varying coupling relationships, for instance, conjunction or disjunction.



Robocup soccer competition.

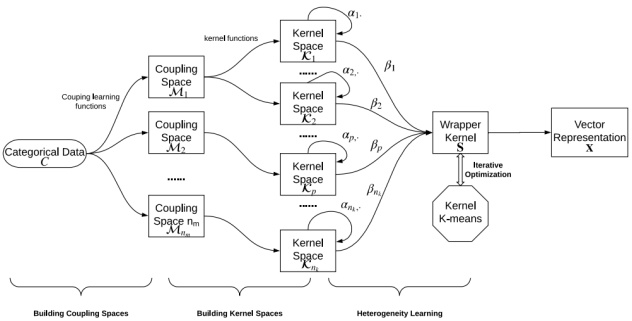


Relationships between coupled behaviors.

[7] C. Wang, L. Cao, C. Chi. Formalization and Verification of Group Behavior Interactions. IEEE T. Systems, Man, and Cybernetics: Systems 45(8): 1109-1124 (2015)

Example: Coupled Representation Learning (UNTIE) [8]

- **Target:** unsupervised representation learning for categorical data
- **Idea:** UNTIE first transforms the coupling spaces to multiple kernel spaces. Then, UNTIE learns the heterogeneities within and between couplings in these kernel spaces by solving a kernel k-means objective.



[8] C. Zhu, L. Cao, and J. Yin, Unsupervised Heterogeneous Coupling Learning for Categorical Representation. IEEE Trans. Pattern Anal. Mach. Intell. 44(1): 533-549, 2022.

Example: Coupled Representation Learning (UNTIE)

- Mapping categorical data to intra-attribute coupling space:

$$\mathcal{M}_{Ia}^{(i)} = \left\{ m_{Ia}^{(i)} \left(v_i^{(j)} \right) | v_i^{(j)} \in V_j \right\}$$

- Mapping categorical data to inter-attribute coupling space:

$$\mathcal{M}_{Ie}^{(i)} = \left\{ m_{Ie}^{(i)} \left(v_i^{(j)} \right) | v_i^{(j)} \in V^{(j)} \right\}$$

- Mapping coupling spaces to multiple kernel spaces:

$$\mathbf{K}_p = \begin{bmatrix} k_p(\mathbf{m}_1, \mathbf{m}_1) & k_p(\mathbf{m}_1, \mathbf{m}_2) & \cdots & k_p(\mathbf{m}_1, \mathbf{m}_{n_v^*}) \\ k_p(\mathbf{m}_2, \mathbf{m}_1) & k_p(\mathbf{m}_2, \mathbf{m}_2) & \cdots & k_p(\mathbf{m}_2, \mathbf{m}_{n_v^*}) \\ \vdots & \vdots & \ddots & \vdots \\ k_p(\mathbf{m}_{n_v^*}, \mathbf{m}_1) & k_p(\mathbf{m}_{n_v^*}, \mathbf{m}_2) & \cdots & k_p(\mathbf{m}_2, \mathbf{m}_{n_v^*}) \end{bmatrix}$$

Example: Coupled Representation Learning (UNTIE)

- Mapping heterogeneous kernel space to a final representation:

$$S_{ij} = \sum_{p=1}^{n_k} \mathbf{K}_{p,\mathbf{i}\cdot}^{\top} \boldsymbol{\omega}_p \mathbf{K}_{p,\mathbf{i}\cdot}$$

- kernel k-means-based representation learning:

$$\begin{aligned} \min_{\mathbf{H},\boldsymbol{\omega}} \ & \mathsf{Tr}\left(\mathbf{S}(\mathbf{I}_{n_0} - \mathbf{H}\mathbf{H}^{\top})\right) \\ \text{s.t.} \ & \mathbf{H} \in \mathcal{R}^{n_0 \times n_0}, \\ & \mathbf{H}\mathbf{H}^{\top} = \mathbf{I}_{n_c}. \end{aligned}$$

Experiments

Dataset	UNTIE	Couplings	CDE	COS	Ahmad	DILCA	Rough	Hamming	BiGAN_WD	VAE_WD	Δ
Zoo	**76.12**	74.85	75.04	72.10	71.34	71.34	62.79	73.27	56.93	24.41	1.44%
DNAPromoter	**95.28**	92.45	61.61	49.24	49.92	85.85	63.20	52.68	51.99	50.87	10.98%
Hayesroth	**54.17**	**54.17**	52.85	38.98	33.76	32.87	38.92	33.06	44.91	37.14	2.50%
Hepatitis	70.40	**73.64**	69.82	46.29	66.72	65.13	59.21	59.21	61.08	51.24	0.00%
Audiology	34.99	34.48	32.18	27.71	**35.38**	31.77	22.36	29.05	20.00	19.97	0.00%
Housevotes	**90.51**	88.36	89.65	88.36	88.36	88.79	87.04	86.64	83.64	53.84	0.96%
Spect	55.04	55.04	52.55	36.26	34.93	34.76	**57.63**	35.94	34.71	48.38	0.00%
Mofn3710	56.65	44.69	56.65	50.18	50.22	48.68	50.62	50.98	**60.34**	49.00	0.00%
Soybeanlarge	**69.29**	64.88	62.19	60.10	56.84	59.42	46.41	55.31	48.38	14.83	11.42%
Primarytumor	24.62	24.87	23.43	19.81	23.65	21.76	22.38	**26.19**	22.17	14.68	0.00%
Dermatology	**97.51**	72.78	73.10	74.58	72.87	72.61	57.99	66.60	38.54	23.82	30.75%
ThreeOf9	34.86	34.86	54.63	35.32	35.32	35.32	**65.19**	54.22	50.03	54.64	0.00%
Wisconsin	93.91	95.58	**96.20**	94.28	95.12	95.49	94.44	89.98	74.26	81.45	0.00%
Crx	**85.49**	52.65	52.65	36.99	52.65	79.29	63.47	79.29	51.81	51.69	7.82%
Breastcancer	93.27	94.75	95.20	93.56	94.89	**95.25**	94.37	93.27	65.94	79.15	0.00%
Mammographic	82.77	**82.89**	81.66	80.06	81.66	82.65	80.67	81.50	60.48	70.59	0.00%
Tictactoe	54.80	**62.61**	54.80	51.88	50.87	52.97	50.19	53.59	54.38	50.24	0.00%
Flare	37.08	31.20	32.44	35.79	34.20	35.59	38.85	**39.22**	31.98	22.30	0.00%
Titanic	33.72	29.77	33.72	29.77	33.72	33.72	**36.27**	33.72	31.58	28.61	0.00%
DNAnominal	**89.79**	67.70	51.14	41.91	46.68	59.18	43.28	41.44	35.18	32.21	51.72%
Splice	79.73	42.29	**87.12**	31.31	47.34	45.87	42.79	42.48	26.60	32.55	0.00%
Krvskp	51.09	51.09	51.03	46.72	**55.17**	**55.17**	53.73	53.86	42.94	50.36	0.00%
Led24	**69.50**	45.82	48.03	53.91	51.83	61.08	32.65	28.82	18.38	13.12	13.79%
Mushroom	82.69	82.76	82.83	**82.91**	82.86	82.39	78.18	82.29	71.48	60.78	0.00%
Connect4	**33.20**	31.14	31.91	27.23	32.88	33.14	30.34	31.43	30.53	29.18	0.18 %
Averaged Rank*	**2.82**	4.34	3.62	6.62	4.9	4.78	5.7	5.66	7.8	8.76	0.8

Clustering F-Score with Different Embedding Methods.

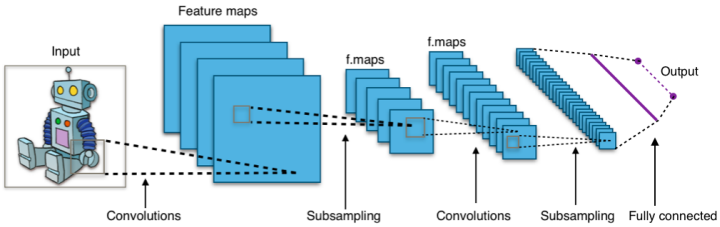
Question

Do Deep Neural Networks Capture Non-IIDnesses?

- What non-IIDnesses they **can** capture?
- What non-IIDnesses they **cannot** capture?

Example: Convolutional Neural Network (CNN) [9]

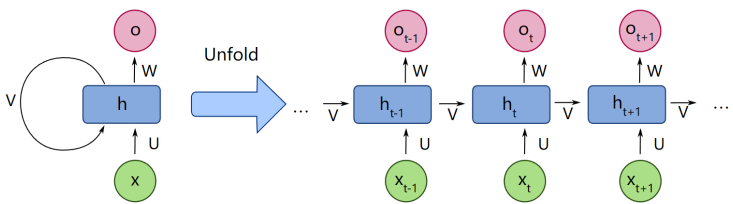
- CNN exploits **spatial locality** by enforcing a local connectivity pattern between neurons of adjacent layers.
- CNN explores the **spatial couplings** between an input feature and its neighbors.



[9] H. Lee, R. B. Grosse, R. Ranganath, A. Y. Ng, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. ICML 2009: 609-616

Example: Recurrent Neural Network (RNN) [10]

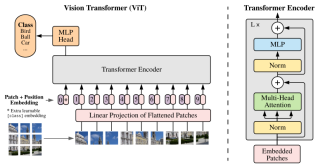
- RNN uses internal state (memory) to process **arbitrary sequences** of inputs.
- RNN explores the **temporal couplings** between an input feature and its context.



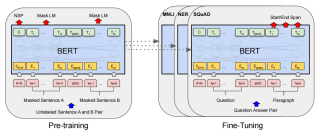
[10] A. Graves, J. Schmidhuber, Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks, NIPS, 545-552, 2008.

Example: Transformer [11]

- Transformer relies on the **attention mechanism**.
- Transformer explores the **couplings** between features.



Transformer for Vision



Transformer for Language

[11] A. Dosovitskiy et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. ICLR 2021.

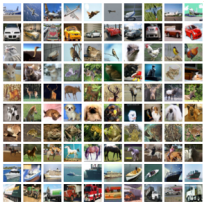
Issues of Deep Learning

- Deep neural networks cannot capture:
  - **Distribution Discrepancy → Distributional Vulnerability**
  - **Feature Causation / Hierarchicalization → Excessive Reliance**
  - **Data Heterogeneity → Biased Representation**

Consequence: Distributional Vulnerability [12]

- Reasons: **Distribution Discrepancy**
  - Networks merely focus on learning to predict labels for training samples, i.e., **in-distribution**.
  - Networks cannot access the samples drawn from distributions different from that of the training samples, i.e., **out-of-distribution**.
  - Networks ignore the **distribution discrepancy** between in- and out-of-distribution samples.
- Results:
  - Networks could provide **unexpected high-confidence** predictions for out-of-distribution samples!
  - **Out-of-distribution detection**

Training in-distribution samples



Test out-distribution samples

[12] Z. Zhao, L. Cao, and K.-Y. Lin, Revealing the distributional vulnerability of discriminators by implicit generators, IEEE Trans. Pattern Anal. Mach. Intell., 45(7): 8888-8901, 2023.

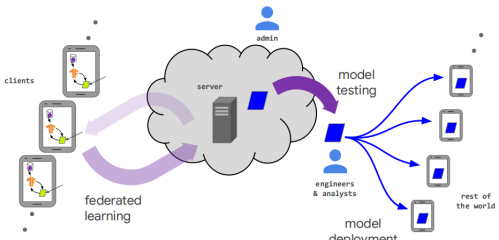
Consequence: Excessive Reliance [13]

- Reasons: **Feature Causation / Hierarchicalization**
  - Networks merely focus the **spurious features** that are unrelated to the core concept, i.e., green pastures and deserts.
  - Networks discard the **invariant features** that are related to the core concept, i.e., cows and camels.
  - Networks ignore the **couplings** between spurious and invariant features, i.e., cows in green pastures and camels in deserts.
- Results:
  - Networks cannot **generalize** to samples with covariare shift!
  - **Out-of-distribution generalization and domain adaptation**

[13] M. Arjovsky, L. Bottou, I. Gulrajani, D. Lopez-Paz, Invariant Risk Minimization, CoRR, 2019.

Consequence: Biased Representation [14]

- Reasons: **Data Heterogeneity**
  - Training samples are heterogeneous.
  - Networks will be misled by samples from different distributions.
- Results:
  - **Networks converge poorly!**
  - **Federated Learning**



[14] Kairouz et al., Advances and Open Problems in Federated Learning. Found. Trends Mach. Learn. 14(1-2): 1-210, 2021.

Concept: Deep Non-IID Learning

Deep non-IID learning refers to the deep learning of non-IIDnesses in data, behaviors, and systems.

Deep non-IID learning aims to

- address non-IID challenges (such as distributional vulnerability caused by out-of-distribution) existing in deep learning theories and systems;
- identify, represent, analyze, discover, and manage data non-IIDnesses by new deep learning theories and approaches;
- develop non-IID deep learning theories and systems that enable non-IID learning by deep neural networks and following deep learning principles.

Approaches: Deep Non-IID Learning

- **Coupled representation learning:** couplings within inputs, between inputs and hidden features, and between inputs and outputs
- **Deep variational learning:** statistical learning + deep learning, e.g., Bayesian deep learning
- **Information theoretic deep learning:** information theory + deep learning
- **Non-IID deep neural learning:** novel deep neural networks addressing non-IIDnesses

Contents

Deep Non-IID Learning Tasks and Applications

In deep learning frameworks:

- Coupled Representation Learning
- Distribution Discrepancy Estimation
- Out-of-distribution Detection
- Out-of-distribution Generalization
- Domain Adaptation
- Federated Learning

Deep Representation Learning [15]

- **Tasks:** learning representations of the data to extract useful information by deep neural networks.
- **Issues:**
  - Ignore the distribution discrepancy between training and test samples, i.e., OOD generalization and detection issues.
  - Ignore the complex couplings between features and values.
  - Ignore the heterogeneous and hierarchical couplings of samples.

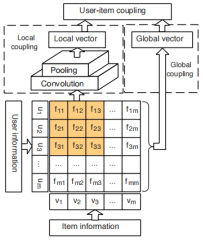
[15] Y. Bengio, A. C. Courville, and P. Vincent, Representation Learning: A Review and New Perspectives. IEEE Trans. Pattern Anal. Mach. Intell. 35(8): 1798-1828, 2013.

Coupled Representation Learning

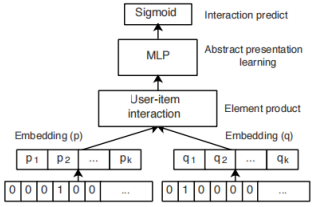
- **Coupled Representation Learning:** Integrating coupling learning with deep representation learning
- **Challenges:**
  - Learning input/attribute couplings and interactions
  - Learning hidden feature couplings
  - Learning observable and hidden feature couplings
  - Learning hierarchical couplings
  - Learning contextual interactions

Coupled Collaborative Filtering (CoupledCF) [16]

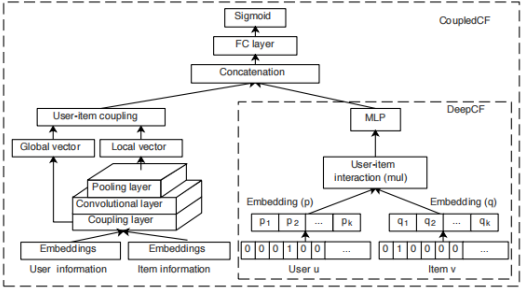
- **Key:** Explore the explicit and implicit couplings within/between users and items.



CNN-based network learns explicit user-item couplings.



DeepCF learns implicit user-item couplings.

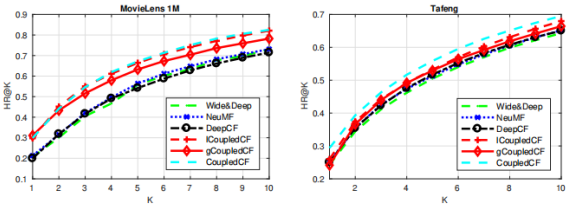


CoupledCF jointly learns explicit and implicit user-item couplings.

[16] Q. Zhang, L. Cao, C. Zhu, Z. Li and J. Sun. CoupledCF: Learning Explicit and Implicit User-item Couplings in Recommendation for Deep Collaborative Filtering, IJCAI2018

Coupled Collaborative Filtering (CoupledCF)

	MovieLens1M		Tafeng	
	HR@10	NDCG@10	HR@10	NDCG@10
NeuMF	0.731	0.448	0.6519	0.4329
Wide&Deep	0.73	0.447	0.642	0.4233
deepCF	0.7147	0.4312	0.6506	0.4322
lCoupledCF	0.8212	0.5408	0.6798	**0.47**
gCoupledCF	0.7826	0.5252	0.6643	0.4205
CoupledCF	**0.8252**	**0.544**	**0.6953**	0.4623



Metric-based Auto-Instructor (MAI) [17]

- **Key:** Explore the heterogeneous couplings between categorical and numerical features.
- **Plain features:** concatenation of one-hot representation of categorical data and numerical data.
- **Coupled features:** product kernel of numerical variable and categorical value:

$$p(a_i^x, v_j) = \frac{1}{N} \sum_{k=1}^{N} \left\{ L_\lambda(v_j^k, v_j) W(\frac{a_i^k - a_i^x}{h_i}) \right\}$$

[17] S. Jian, L. Hu, L. Cao, and K. Lu. Metric-based Auto-Instructor for Learning Mixed Data Representation. AAAI2018

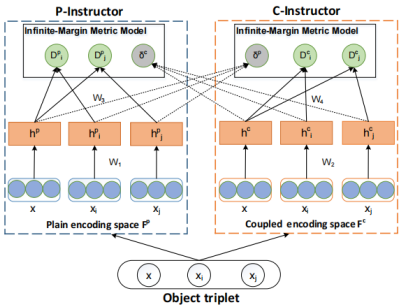
Metric-based Auto-Instructor (MAI)

- Distance metric:

$$D^p(\mathbf{h}^p, \mathbf{h}_i^p) = (\mathbf{h}^p - \mathbf{h}_i^p)\mathbf{W}_3(\mathbf{h}^p - \mathbf{h}_i^p)^\top$$
$$D^c(\mathbf{h}^c, \mathbf{h}_i^c) = (\mathbf{h}^c - \mathbf{h}_i^c)\mathbf{W}_4(\mathbf{h}^c - \mathbf{h}_i^c)^\top$$

- P-Instructor and C-Instructor over triplets:

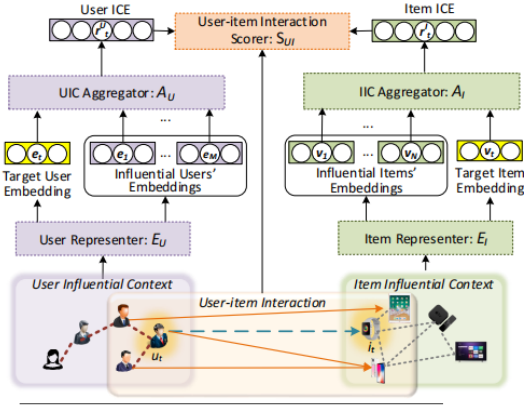
$$L_{\Theta^p} = -\sum_{\langle x, x_i, x_j \rangle} \log P_{\Theta^p}\left(D_i^p > D_j^p | \delta_{\mathbf{h}^c}^c\right)$$
$$L_{\Theta^c} = -\sum_{\langle x, x_i, x_j \rangle} \log P_{\Theta^c}\left(D_i^c > D_j^c | \delta_{\mathbf{h}^p}^p\right)$$



Datasets	Plain encoding	Coupled encoding	CoupledMC	Autoencoder	MAI-F	MAI-D
Echo	0.1789±0.1033	0.1749±0.0444	0.1237±0.1147	0.2493±0.0207	**0.3246±0.0000**	**0.3304±0.0000**
Hepatitis	0.1453±0.0703	0.1761±0.0292	0.1532±0.0342	0.1689±0.0163	**0.1848±0.0000**	**0.1905±0.0000**
MPG	0.1490±0.0106	0.1477±0.0184	0.1373±0.0347	0.1536±0.0086	**0.1831±0.0232**	**0.1770±0.0000**
Heart	**0.3130±0.0688**	0.1439±0.0642	0.1037±0.1215	**0.3302±0.0042**	0.2632±0.0000	0.2774±0.0000
ACA	0.3204±0.1518	0.3433±0.1726	0.3182±0.0627	0.3477±0.0844	**0.4258±0.0000**	**0.4258±0.0000**
CRX	0.2322±0.1191	0.0836±0.1109	0.2714±0.1361	0.1445±0.1477	**0.4267±0.0000**	**0.4267±0.0000**
CMC	0.0293±0.0052	0.0269±0.0013	**0.0333±0.0070**	0.0292±0.0037	**0.0327±0.0077**	0.0303±0.0081
Income	0.1139±0.0361	**0.1414**±0.0291	0.1258±0.0658	0.1314±0.0000	**0.1325±0.0000**	**0.1325±0.0000**
Average	0.1853±0.0707	0.1547±0.0588	0.1583±0.0722	0.1944±0.0353	**0.2467±0.0064**	**0.2488±0.0010**

Heterogeneous relations-Embedded Recommender System (HERS) [18]

- **Key:** Explore three heterogeneous relations: user-user, item-item, and user-item.



- User Representer E_U: it maps target user u_t and its influential users in UIC to the corresponding user embeddings, i.e., $E_U(\mathcal{U}_{u_t}) \mapsto \mathcal{E}_{u_t}$ where $\mathcal{E}_{u_t} = \{\mathbf{e}_t, \mathbf{e}_1, \cdots \mathbf{e}_M\}$.
- Item Representer E_I: it maps target item i_t and its influential items in IIC to the corresponding item embeddings, i.e., $E_I(\mathcal{I}_{i_t}) \mapsto \mathcal{E}_{i_t}$ where $\mathcal{E}_{i_t} = \{\mathbf{v}_t, \mathbf{v}_1, \cdots \mathbf{v}_N\}$.
- UIC Aggregator A_U: it learns a representation $\mathbf{r}_t^U$ for the influential context $\mathcal{C}_{u_t}$, namely influential context embedding (ICE). Formally, we have $A_U(\mathcal{C}_{u_t}, \mathcal{E}_{u_t}) \mapsto \mathbf{r}_t^U$.
- IIC Aggregator A_I: it learns i_t's ICE by aggregating the influential context $\mathcal{C}_{i_t}$, that is, $A_I(\mathcal{C}_{i_t}, \mathcal{E}_{i_t}) \mapsto \mathbf{r}_t^I$.
- User-item Interaction Scorer S_{UI}: it learns to score the interaction strength between the target user-item pair $\langle u_t, i_t \rangle$ in terms of the user ICE $\mathbf{r}_t^U$ and the item ICE $\mathbf{r}_t^I$, namely $S_{UI}(\mathbf{r}_t^U, \mathbf{r}_t^I, y_{u_t,i_t}) \mapsto s_{\langle \mathcal{C}_u, \mathcal{C}_i \rangle}$ (cf. Eq. 1).

[18] L. Hu, S. Jian, L. Cao, Z. Gu, Q. Chen, A. Amirbekyan. HERS: Modeling Influential Contexts with Heterogeneous Relations for Sparse and Cold-start Recommendation, AAAI2019

Heterogeneous relations-Embedded Recommender System (HERS)

	Delicious				Lastfm			
	MAP@5	MAP@20	nDCG@5	nDCG@20	MAP@5	MAP@20	nDCG@5	nDCG@20
BPR-MF	0.4157	0.3225	0.4318	0.3744	0.5154	0.4586	0.6252	0.6334
SoRec	0.4174	0.3390	0.4476	0.3965	0.5350	0.4775	0.6412	0.6457
Social MF	0.4181	0.3409	0.4520	0.4017	0.5489	0.4907	0.6544	0.6575
SoReg	0.4239	0.3444	0.4577	0.4056	0.5495	0.4878	0.6548	0.6541
CMF	0.4375	0.3507	0.4739	0.4158	0.5530	0.4928	0.6549	0.6749
FM	0.4246	0.3363	0.4522	0.3896	0.5366	0.4837	0.6453	0.6723
NFM	0.4565	0.3754	0.4924	0.4347	0.5462	0.4885	0.6516	0.6702
ICAU-HERS	**0.5477**	**0.4200**	**0.6064**	**0.5273**	**0.5865**	**0.5302**	**0.6913**	**0.7021**

Item recommendation for test users of Delicious and Lastfm

Distribution Discrepancy Estimation

- **Task:** Evaluate the discrepancy between two probability distributions given their corresponding samples.
- **Assumption:** Samples from each distribution are independent.
- **Challenge:** Estimate the distributional discrepancy and non-IIDness between two datasets.

Maximum Mean Discrepancy with A Deep Kernel (MMD-D) [19]

- **Key:** Explore the distributional discrepancy between two datasets by deep kernels.
- **Insight:** Kernels constructed by deep neural nets can adapt to variations in distribution smoothness and shape over space.
- **Model:**

$$\sqrt{\mathbb{E}\left[k_w(x,x') + k_w(y,y') - 2k_w(x,y)\right]},$$
$$k_w(x,y) = [(1-\epsilon)k_1(\phi_w(x),\phi_w(y)) + \epsilon]k_2(x,y),$$
$$x, x \sim p, y, y' \sim q.$$

[19] F. Liu, W. Xu, J. Lu, G. Zhang, A. Gretton, and D. J. Sutherland, Learning deep kernels for non-parametric two-sample tests, ICML, pp. 6316–6326, 2020.

Experiments

N	ME	SCF	C2ST-S	C2ST-L	MMD-O	MMD-D
200	0.414±0.050	0.107±0.018	0.193±0.037	0.234±0.031	0.188±0.010	**0.555±0.044**
400	0.921±0.032	0.152±0.021	0.646±0.039	0.706±0.047	0.363±0.017	**0.996±0.004**
600	**1.000±0.000**	0.294±0.008	**1.000±0.000**	0.977±0.012	0.619±0.021	**1.000±0.000**
800	**1.000±0.000**	0.317±0.017	**1.000±0.000**	**1.000±0.000**	0.797±0.015	**1.000±0.000**
1 000	**1.000±0.000**	0.346±0.019	**1.000±0.000**	**1.000±0.000**	0.894±0.016	**1.000±0.000**
Avg.	0.867	0.243	0.768	0.783	0.572	**0.910**

Average test power over the MNIST dataset.

H-divergence [20]

- **Key:** Explore the distributional discrepancy between two datasets by empirical risk minimization.
- **Insight:** Two distributions are different if the optimal decision loss is higher on their mixture than on each individual distribution.
- **Model:**

$$\phi(\epsilon_u(h_u^*) - \epsilon_p(h_p^*), \epsilon_u(h_u^*) - \epsilon_q(h_q^*)),$$
$$h_u^* \in \arg\min_{h\in\mathcal{H}} \epsilon_u(h), h_q^* \in \arg\min_{h\in\mathcal{H}} \epsilon_q(h), h_p^* \in \arg\min_{h\in\mathcal{H}} \epsilon_p(h).$$

[20] S. Zhao, A. Sinha, Y. He, A. Perreault, J. Song, and S. Ermon, Comparing distributions by measuring differences that affect decision making, ICLR, pp. 1–20, 2022.

Experiments

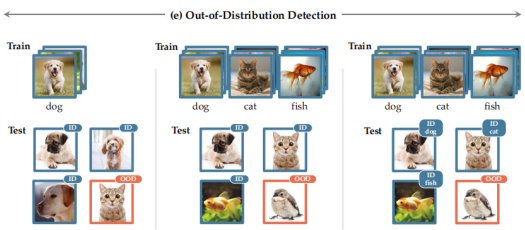
N	ME	SCF	C2ST-S	C2ST-L	MMD-O	MMD-D	H-Div
200	0.414±0.050	0.107±0.018	0.193±0.037	0.234±0.031	0.188±0.010	0.555±0.044	**1.000±0.000**
400	0.921±0.032	0.152±0.021	0.646±0.039	0.706±0.047	0.363±0.017	0.996±0.004	**1.000±0.000**
600	1.000±0.000	0.294±0.008	1.000±0.000	0.977±0.012	0.619±0.021	1.000±0.000	**1.000±0.000**
800	1.000±0.000	0.317±0.017	1.000±0.000	1.000±0.000	0.797±0.015	1.000±0.000	**1.000±0.000**
1000	1.000±0.000	0.346±0.019	1.000±0.000	1.000±0.000	0.894±0.016	1.000±0.000	**1.000±0.000**
Avg.	0.867	0.243	0.768	0.783	0.572	0.910	**1.000**

Average test power over the MNIST dataset.

Out-of-distribution Detection [21]

Focusing on distributional discrepancy between in- and out-of-samples:

- **In-distribution (ID) samples:** Test samples drawn from the same unknown distribution of training samples.
- **Out-of-distribution (OOD) samples:** Test samples drawn from distributions differing from the unknown distribution.
- **Over-confidence Problem:** A network learned from ID samples could assign high-confidence predictions for OOD samples.



[21] J. Yang, K. Zhou, Y. Li, and Z. Liu, Generalized out-of-distribution detection: A survey, CoRR, pp. 1–20, 2021

Out-of-distribution Detection

- **Task:** identify whether a test sample is drawn from an ID or OOD.
- **Assumption:**
  - ID samples are applied to train a network.
  - ID and OOD samples are drawn from different distributions.
  - OOD samples are with semantic shift w.r.t. ID samples.
- **Challenge**
  - explore the distributional discrepancy between ID and OOD samples.
  - explore the non-IIDnesses between ID and OOD samples.

Maximum over Softmax Probabilities (MSP) [22]

- **Key:** Distinguish ID and OOD samples according to OOD scores, and ID and OOD samples are expected to own high and low scores, respectively.
- **Metric:** AUROC can be interpreted as the probability that an ID sample has a greater score than an OOD sample.

- **Insight:** Correctly classified examples tend to have greater maximum softmax probabilities than erroneously classified and out-of-distribution examples.
- **Model:** $\mathcal{S}(\mathbf{x}) = \max_y q_\theta(y|\mathbf{x})$

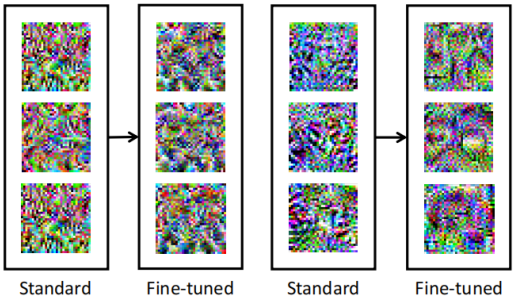
[22] D. Hendrycks and K. Gimpel, A baseline for detecting misclassified and out-of-distribution examples in neural networks, ICLR, 2017, pp. 1–12.

Experiments

In-Distribution / Out-of-Distribution	AUROC /Base Softmax	AUROC /Base AbMod	AUPR In/Base Softmax	AUPR In/Base AbMod	AUPR Out/Base Softmax	AUPR Out/Base AbMod
MNIST/Omniglot	95/50	100/50	95/52	100/52	95/48	100/48
MNIST/notMNIST	87/50	100/50	88/50	100/50	90/50	100/50
MNIST/CIFAR-10bw	98/50	100/50	98/50	100/50	98/50	100/50
MNIST/Gaussian	88/50	100/50	88/50	100/50	90/50	100/50
MNIST/Uniform	99/50	100/50	99/50	100/50	99/50	100/50
Average	93	100	94	100	94	100

Fine-tuning Discriminators by Implicit Generators (FIG) [23]

- **Key:** Explore the non-IIDness between ID and OOD samples by generating specific OOD samples for a given pretrained network.
- **Insight:** An OOD sample with high-confidence prediction has low entropy.
- **Method:**
  - Derive an implicit generator for a pretrained network without training.
  - Drawn OOD samples from the implicit generator.
  - Fine-tune a pretrained network with its specific OOD samples.



(a) CIFAR10 (Dog) (b) SVHN (Digit 6)

[23] Z. Zhao, L. Cao, and K.-Y. Lin, Revealing the distributional vulnerability of discriminators by implicit generators, IEEE Trans. Pattern Anal. Mach. Intell., 45(7): 8888-8901, 2023.

Insight

- **Prediction:** A pretrained network learned from ID samples

$$q_\theta(y|\mathbf{x}) = \frac{\exp f_\theta(\mathbf{x}, y)}{\sum_{y' \in [C]} \exp f_\theta(\mathbf{x}, y')}.$$

  It will provide unexpected high-confidence predictions for OOD samples.

- **Shannon Entropy:** An OOD sample with high-confidence prediction has low entropy

$$H_{\theta,\mathbf{x}}(C) = -\sum_{y \in [C]} q_\theta(y|\mathbf{x}) \log q_\theta(y|\mathbf{x}).$$

- **Generator:** An implicit generator is proportional to the negative entropy

$$q_\theta(\mathbf{x}) \propto \frac{\exp\left(-E_\theta(\mathbf{x})\right)}{\int \exp\left(-E_\theta(\mathbf{x}')\right) \mathrm{d}\mathbf{x}'},$$

$$E_\theta(\mathbf{x}) \triangleq \sum_{y \in [C]} f_\theta(\mathbf{x}, y) \left(1 - \exp f_\theta(\mathbf{x}, y)\right).$$

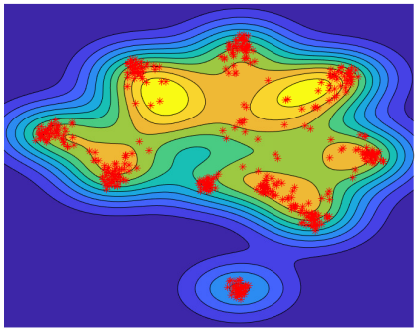
Experiments

In-dist	Out-of-dist	GS / MIXUP / AD / JCL / DCC* / FIG			
		ResNet18	VGG19	ShuffleNetV2	DenseNet100
SVHN	LSUN(r)	99.2 / 95.4 / 94.7 / 98.7 / 94.5 / **99.4**	98.5 / 96.3 / **99.7** / 98.3 / 91.0 / 99.3	98.0 / 95.9 / 96.4 / 91.6 / 99.1 / **99.7**	95.4 / 95.9 / 90.0 / 91.4 / **98.4** / **98.4**
	LSUN(c)	98.4 / 92.7 / 95.8 / **99.7** / 97.6 / 97.1	98.9 / 94.3 / 96.4 / **98.2** / 97.0 / 98.0	96.7 / 92.7 / 95.2 / 94.3 / **98.8** / 97.7	92.9 / 95.2 / 90.8 / 91.8 / **99.1** / 95.9
	TinyImageNet(r)	99.0 / 95.2 / 95.1 / 98.9 / 94.9 / **99.4**	98.6 / 96.8 / **99.7** / 98.3 / 93.4 / 99.1	98.0 / 96.2 / 96.8 / 92.9 / **99.9** / 99.6	95.5 / 95.7 / 91.2 / 90.3 / 98.0 / **98.8**
	TinyImageNet(c)	98.7 / 94.8 / 96.6 / **99.7** / 97.4 / 98.2	99.2 / 95.7 / **99.5** / 98.4 / 95.7 / 98.8	97.3 / 96.0 / 96.1 / 95.4 / 98.4 / **99.0**	93.5 / 95.9 / 91.3 / 91.9 / **99.5** / 97.5
	Caltech256(r)	95.9 / 90.9 / 93.3 / 90.4 / 92.5 / **97.2**	95.4 / 92.9 / 93.8 / 95.2 / 91.5 / **95.5**	95.2 / 92.7 / 94.2 / 91.2 / **97.8** / 97.3	91.9 / 93.5 / 88.8 / 89.6 / **95.5** / 95.3
	Caltech256(c)	97.7 / 91.7 / 94.1 / 97.3 / 94.5 / **98.8**	96.9 / 94.1 / 92.4 / 97.7 / 88.7 / **98.3**	96.5 / 90.5 / 92.5 / 92.6 / 98.5 / **98.7**	94.8 / 94.7 / 89.5 / 90.8 / 97.0 / **99.2**
	COCO(r)	97.5 / 92.9 / 94.6 / 94.6 / 95.1 / **98.4**	96.6 / 94.9 / **99.5** / 96.3 / 91.6 / 97.2	96.7 / 94.6 / 96.3 / 91.5 / 97.4 / **98.8**	94.2 / 95.3 / 88.4 / 90.0 / 96.1 / **96.7**
	COCO(c)	97.9 / 91.1 / 94.2 / 97.6 / 93.6 / **99.1**	97.2 / 93.7 / 97.2 / 98.1 / 88.4 / **98.4**	96.6 / 90.5 / 95.2 / 92.6 / 98.7 / **99.2**	95.0 / 93.9 / 91.0 / 90.9 / 97.2 / **99.4**
	Ave.	98.0 / 93.1 / 94.8 / 97.1 / 95.0 / **98.5**	97.7 / 94.8 / 97.3 / 97.6 / 92.2 / **98.1**	96.9 / 93.7 / 95.3 / 92.8 / 98.6 / **98.8**	94.2 / 95.0 / 90.1 / 90.8 / **97.6** / **97.6**
CIFAR10	LSUN(r)	92.8 / 92.8 / 91.9 / 90.8 / 98.7 / **99.0**	89.4 / 95.3 / 80.4 / 90.8 / 96.4 / **97.4**	83.0 / 83.5 / 81.4 / 88.8 / 98.6 / **99.8**	92.2 / 87.8 / 90.6 / 94.7 / **99.4** / 99.1
	LSUN(c)	95.0 / 95.7 / 94.1 / 90.8 / 98.2 / **98.9**	92.3 / 95.7 / 86.4 / 90.1 / **97.3** / 96.7	89.0 / 86.7 / 82.5 / 91.9 / **98.0** / 93.0	93.1 / 96.1 / 91.5 / 97.3 / **98.3** / 98.0
	TinyImageNet(r)	91.9 / 89.8 / 89.1 / 92.7 / 95.4 / **99.0**	86.8 / 93.9 / 78.7 / 84.3 / 92.4 / **96.4**	82.0 / 82.6 / 77.2 / 84.7 / **97.3** / 96.2	91.5 / 87.6 / 85.9 / 93.6 / **99.1** / 97.3
	TinyImageNet(c)	93.2 / 93.4 / 93.0 / 92.7 / **96.2** / 95.7	89.7 / 94.3 / 84.4 / 92.7 / 91.3 / **94.9**	87.4 / 85.9 / 85.8 / 88.2 / **96.5** / 92.2	92.3 / 93.4 / 89.3 / 96.2 / **98.7** / 96.5
	Caltech256(r)	86.9 / 80.0 / 85.9 / **92.9** / 85.0 / 88.0	82.5 / **86.1** / 76.1 / 84.3 / 80.4 / 83.4	79.3 / 78.9 / 76.3 / 81.2 / **84.6** / 83.0	86.7 / 79.5 / 85.1 / **90.1** / 87.6 / 87.8
	Caltech256(c)	93.0 / 90.3 / 91.5 / 84.3 / 91.7 / **94.7**	88.5 / **92.7** / 79.4 / 89.5 / 87.6 / 90.7	82.5 / 80.4 / 78.1 / 79.1 / 87.1 / **91.9**	91.0 / 89.9 / 90.8 / **95.2** / 91.3 / 94.4
	COCO(r)	87.9 / 83.9 / 87.2 / **91.7** / 85.9 / 90.5	85.0 / **88.2** / 79.4 / 85.2 / 81.0 / 86.5	80.5 / 79.9 / 80.8 / 82.3 / 85.1 / **88.3**	87.6 / 83.8 / 85.8 / 88.5 / 88.8 / **89.6**
	COCO(c)	92.7 / 87.5 / 91.6 / 85.2 / 89.9 / **94.5**	88.4 / 93.8 / 79.2 / 90.8 / 87.3 / **91.7**	84.1 / 81.6 / 78.7 / 79.6 / 87.9 / **92.4**	91.0 / 89.5 / 90.7 / 93.9 / 90.6 / **96.2**
	Ave.	91.7 / 89.2 / 90.5 / 90.1 / 92.6 / **95.0**	87.8 / 92.5 / 80.5 / 88.5 / 89.2 / **92.2**	83.5 / 82.4 / 80.1 / 84.5 / 91.9 / **92.1**	90.7 / 88.4 / 88.7 / 93.7 / 94.2 / **94.9**
CIFAR100	LSUN(r)	83.6 / 78.0 / 82.7 / 87.6 / 93.4 / **93.8**	79.2 / 75.4 / 71.5 / 80.7 / **87.3** / 82.5	71.9 / 55.9 / 68.8 / 65.7 / 80.4 / **82.3**	81.9 / 75.2 / 82.6 / 86.1 / **98.7** / 98.6
	LSUN(c)	85.4 / 77.6 / 81.5 / 80.5 / **88.3** / 85.0	83.7 / 80.9 / 78.3 / 81.9 / 85.6 / **85.9**	75.1 / 71.2 / 76.7 / 77.3 / **87.7** / 82.9	81.6 / 81.9 / 81.4 / 88.4 / **95.3** / 94.6
	TinyImageNet(r)	82.9 / 74.4 / 81.5 / 87.2 / 92.8 / **97.1**	76.6 / 75.6 / 70.9 / 80.5 / **81.6** / 80.0	72.5 / 61.1 / 64.4 / 63.5 / 78.4 / **84.7**	82.5 / 74.1 / 82.3 / 83.5 / 98.6 / **98.7**
	TinyImageNet(c)	87.1 / 83.7 / 83.8 / 83.3 / **91.4** / 89.6	83.4 / 81.8 / 77.3 / 79.9 / 83.9 / **87.2**	78.9 / 78.8 / 78.5 / 75.9 / **88.5** / 86.4	84.1 / 84.9 / 84.0 / 87.8 / **97.6** / 96.6
	Caltech256(r)	75.3 / 75.2 / 76.2 / 79.7 / **83.3** / 82.4	71.5 / 71.2 / 69.1 / **87.8** / 77.3 / 76.9	67.2 / 68.9 / 68.5 / 67.8 / **74.6** / 72.5	74.4 / 72.3 / 75.8 / 81.3 / 82.9 / **83.4**
	Caltech256(c)	82.1 / 83.7 / 81.6 / 83.6 / 87.9 / **89.2**	79.7 / 79.9 / 74.7 / 80.6 / 76.7 / **84.5**	71.7 / 70.0 / 71.5 / 71.2 / 76.3 / **81.5**	81.6 / 80.1 / 81.3 / 85.7 / 86.9 / **92.9**
	COCO(r)	77.4 / 78.8 / 78.8 / 80.2 / 83.2 / **84.1**	75.6 / 77.7 / 72.9 / 76.8 / **79.7** / 79.2	70.8 / 69.7 / 71.5 / 69.2 / **77.8** / 75.6	77.0 / 79.8 / 78.5 / 80.6 / 84.5 / **85.4**
	COCO(c)	83.2 / 80.5 / 82.2 / 82.5 / 89.0 / **93.4**	81.7 / 79.3 / 75.8 / 81.7 / 78.4 / **85.6**	71.8 / 70.5 / 71.1 / 71.6 / 78.3 / **85.6**	82.0 / 79.6 / 82.3 / 84.5 / 88.1 / **94.7**
	Ave.	82.1 / 79.0 / 81.0 / 83.1 / 88.7 / **89.3**	78.9 / 77.7 / 73.8 / 81.2 / 81.3 / **82.7**	72.5 / 68.3 / 71.4 / 70.3 / 80.2 / **81.4**	80.6 / 78.5 / 81.0 / 84.7 / 91.6 / **93.1**

OOD detection performance of FIG in terms of AUROC

Learning from Cross-class Vicinity Distribution (LCVD) [24]

- **Key:** Explore the non-IIDness between ID and OOD samples by considering the vicinity distributions of ID samples.
- **Insight:** An OOD input generated by mixing multiple in-distribution inputs does not belong to the same classes as its constituents.
- **Method:**
  - Construct the OOD samples of an ID sample by combining it with different classes of ID samples.
  - Maximize the cross-entropy loss on OOD samples to encourage low confidence.



[24] Z. Zhao, L. Cao, and K.-Y. Lin, Out-of-distribution Detection by Cross-class Vicinity Distribution of In-distribution Data, IEEE Trans. Neural Networks Learn. Syst., 2023.

Algorithm Framework

- Derive the generic expected risk:

$$\mathcal{R}(\theta) = -\int \log Q_\theta(y|\mathbf{x}) \mathrm{d}P_I(\mathbf{x}, y) + \int \log(1 - Q_\theta(y|\mathbf{x})) \mathrm{d}P_O(\mathbf{x}, y).$$

- Construct vicinity distribution:

$$\widetilde{P}_I(\mathbf{x}, y) = \frac{1}{N_I} \sum_{i=1}^{N_I} \delta(\mathbf{x} = \mathbf{x}_i^I, y = y_i^I) \text{ for ID samples,}$$

$$\widetilde{P}_O(\mathbf{x}, y) = \frac{1}{N_I} \sum_{i=1}^{N_I} \mathbb{E}_{\mathbf{x}_1^I} \ldots \mathbb{E}_{\mathbf{x}_{M-1}^I} \left[\delta\left(\mathbf{x} = \mathbf{x}^O, y = y^O\right)\right] \text{ for OOD samples.}$$

- Estimate the generic empirical risk:

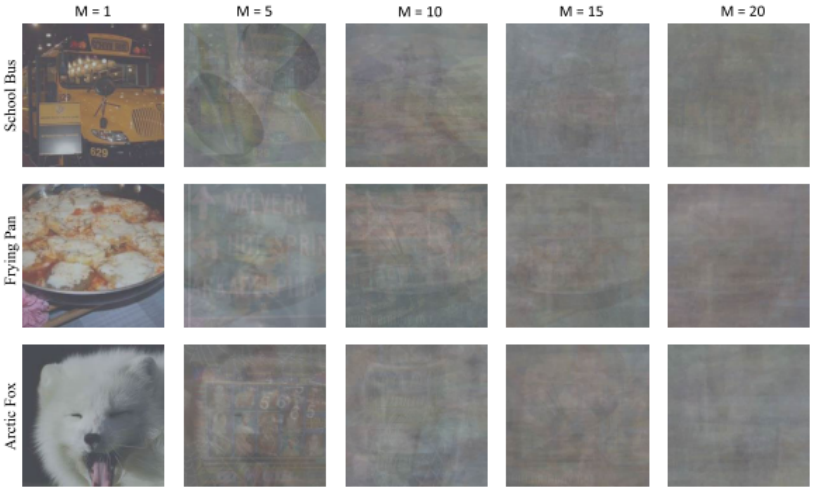
$$\widetilde{\mathcal{R}}(\theta) = -\sum_{i=1}^{N_I} \log Q_\theta(y_i^I|\mathbf{x}_i^I) - \sum_{j=1}^{N_O} \log\left(1 - Q_\theta(y_j^O|\mathbf{x}_j^O)\right).$$

Experiments

In-dist	Method	AUROC ↑	AUPRIN ↑	AUPROUT ↑	FPR ↓	Detection ↓
CIFAR10	OE	78.1	77.1	76.4	73.1	26.3
	POEM	79.5	73.7	78.2	67.2	26.7
	MIXUP	76.5	74.3	77.3	65.3	27.2
	JCL	78.1	79.8	77.6	74.3	25.7
	JEM	74.3	68.2	74.8	72.2	27.9
	CSI	79.2	75.7	78.4	67.8	24.5
	SSD	80.2	76.7	79.4	66.8	24.5
	LCVD	**82.4**	**80.3**	**80.1**	**65.1**	**23.6**
SVHN	OE	92.1	90.0	92.4	38.7	16.1
	POEM	94.5	94.2	92.8	32.4	12.0
	MIXUP	92.9	90.3	93.4	36.3	11.6
	JCL	93.1	91.7	92.6	32.6	15.9
	JEM	93.7	91.4	91.2	36.8	16.5
	CSI	93.9	91.0	92.0	39.3	11.7
	SSD	94.4	92.1	93.0	38.3	11.6
	LCVD	**95.8**	**94.7**	**93.7**	**30.6**	**10.5**
Mini-Imagenet	OE	73.0	76.0	68.9	85.8	31.9
	POEM	74.6	77.3	69.7	84.6	30.1
	MIXUP	74.7	77.6	69.6	85.4	29.3
	JCL	73.1	75.5	67.0	87.4	32.1
	JEM	73.7	68.2	74.8	72.2	27.8
	CSI	76.2	74.8	67.7	72.7	32.2
	SSD	76.8	74.8	67.8	71.2	32.4
	LCVD	**78.6**	**79.6**	**76.7**	**70.7**	**20.4**

OOD detection performance of LCVD in terms of AUROC

Experiments



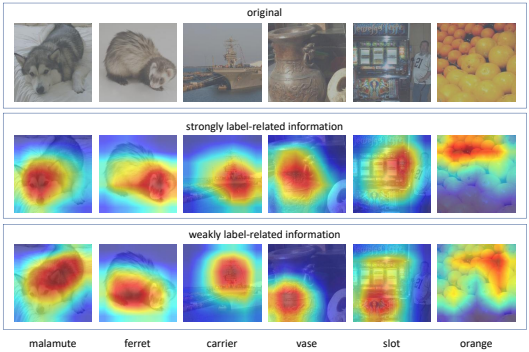
OOD samples drawn from the cross-class vicinity distribution of the training ID samples in Mini-Imagenet

Dual Representation Learning (DRL) [25]

- **Key:** Explore the non-IIDness within ID samples by exploring strongly and weakly label-related information.
- **Insight:**
  - A single network cannot capture all the label-related information.

$$\max \mathcal{I}(\mathcal{D}; \mathcal{Y}) - \beta_{\mathcal{D}} \mathcal{I}(\mathcal{X}; \mathcal{D}).$$

  - Considering more label information makes networks harder to provide high-confidence predictions for OOD samples.



[25] Z. Zhao and L. Cao, Dual Representation Learning for Out-of-distribution Detection, Transactions on Machine Learning Research, 2023.

Algorithm Framework

- Strongly label-related representation is obtained from a pretrained network:

$$\mathbf{d} = g_\phi(\mathbf{x}).$$

- Weakly label-related representation is obtained by integrating multiple representations different from the strongly label-related representation:

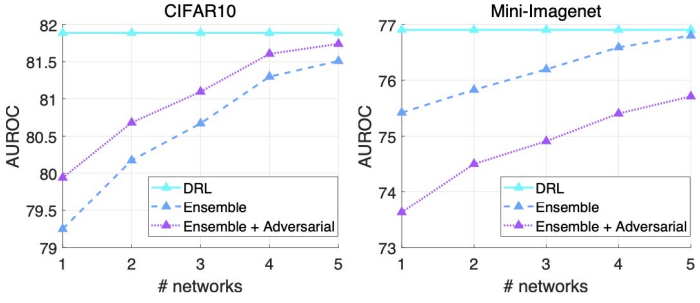
$$\mathbf{c} = \sum_{i=1}^{\infty} w_i \mathbf{z}_i = f_\theta(\mathbf{x}, \mathbf{d}).$$

- Coupling the two representations to calculate an OOD score:

$$\mathcal{S}(\mathbf{x}) = \max_{y \in [1,K]} \left(h(\mathbf{c}, y) + h(\mathbf{d}, y) \right) / 2.$$

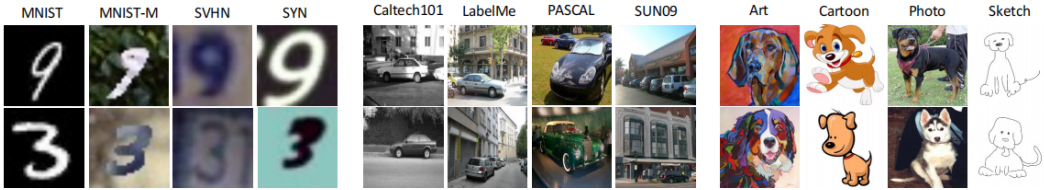
Experiment

Dataset	Metric	JCL	CSI	SSL	DeConf-C	MOS	KNN+	CIDER	DRL
CIFAR10	CIFAR100 (Near)	84.3	87.6	88.2	89.2	88.4	85.6	89.1	**89.5**
	CUB200 (Near)	60.9	61.2	62.1	62.4	60.9	62.9	63.0	**63.7**
	Oxfordflowers102 (Far)	84.6	85.4	90.4	90.1	88.7	89.6	90.5	**91.2**
	DTD47 (Far)	88.6	88.6	90.5	91.2	90.3	90.3	91.6	**92.6**



Out-of-distribution Generalization [26]

- **Task:** learn a model from the source domain that can generalize to an unseen domain.
- **Assumption:**
  - ID and OOD samples are drawn from different distributions.
  - OOD samples are with covariate shift w.r.t. ID samples.
  - OOD samples are unavailable in the training phase.
- **Challenge:** explore the non-IIDness between source and unseen domains.

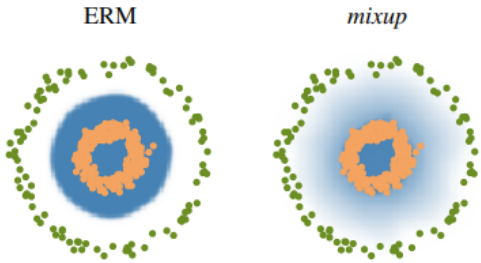


[26] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, Domain generalization: A survey, IEEE Trans. Pattern Anal. Mach. Intell., 45(4): 4396–4415, 2023.

Mixup [27]

- **Key:** Explore the distributional discrepancy between ID and OOD samples by augmenting ID samples.
- **Insight:** Convex combinations of pairs of examples and their labels can alleviate the memorization and sensitivity issues to adversarial examples.
- **Model:** Virtual feature-target vectors,

$$\widetilde{x} = \lambda x_i + (1-\lambda)x_j,$$
$$\widetilde{y} = \lambda y_i + (1-\lambda)y_j.$$



[27] H. Zhang, M. Ciss'e, Y. N. Dauphin, and D. Lopez-Paz, mixup: Beyond empirical risk minimization, ICLR, pp. 1–13, 2018.

Experiments

Model	Method	Epochs	Top-1 Error	Top-5 Error
ResNet-50	ERM (Goyal et al., 2017)	90	23.5	-
	mixup $\alpha = 0.2$	90	**23.3**	**6.6**
ResNet-101	ERM (Goyal et al., 2017)	90	22.1	-
	mixup $\alpha = 0.2$	90	**21.5**	**5.6**
ResNeXt-101 32*4d	ERM (Xie et al., 2016)	100	21.2	-
	ERM	90	21.2	5.6
	mixup $\alpha = 0.4$	90	**20.7**	**5.3**
ResNeXt-101 64*4d	ERM (Xie et al., 2016)	100	20.4	5.3
	mixup $\alpha = 0.4$	90	**19.8**	**4.9**
ResNet-50	ERM	200	23.6	7.0
	mixup $\alpha = 0.2$	200	**22.1**	**6.1**
ResNet-101	ERM	200	22.0	6.1
	mixup $\alpha = 0.2$	200	**20.8**	**5.4**
ResNeXt-101 32*4d	ERM	200	21.3	5.9
	mixup $\alpha = 0.4$	200	**20.1**	**5.0**

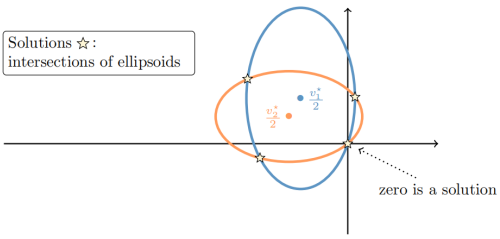
Validation errors for ERM and mixup on the development set of ImageNet-2012.

Invariant Risk Minimization (IRM) [28]

- **Key:** Explore the distributional discrepancy between ID and OOD samples by developing spurious and invariant correlations.
- **Insight:** Find a data representation such that the optimal classifier on top of that representation matches for all environments.
- **Model:**

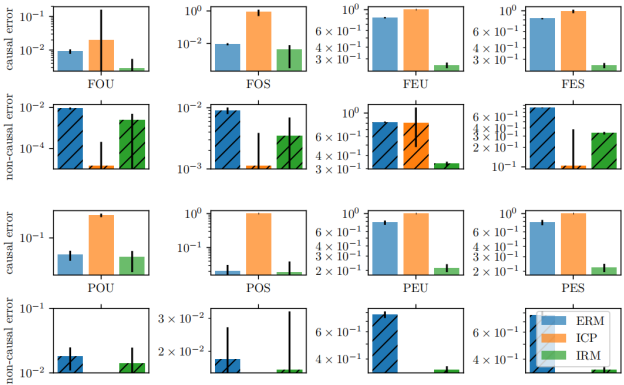
$$\min_{\Phi:\mathcal{X}\mapsto\mathcal{H}, \Phi:\mathcal{X}\mapsto\mathcal{H}} \sum_{e\in\varepsilon_{\text{tr}}} R^e(w\circ\Phi)$$

s.t. $w \in \arg_{\overline{w}:\mathcal{H}\mapsto\mathcal{Y}} \min R^e(\overline{w}\circ\Phi), \forall e \in \varepsilon_{\text{tr}}$



[28] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, Invariant risk minimization, CoRR, pp. 1–31, 2019.

Experiments

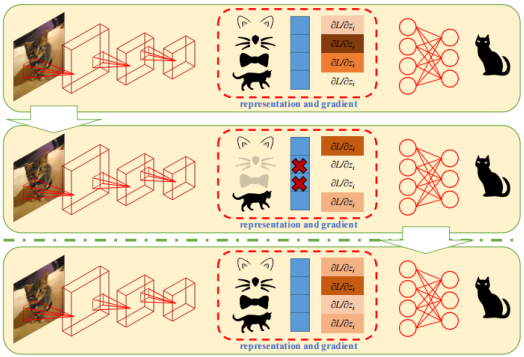


Average errors on causal (plain bars) and non-causal (striped bars) weights for our synthetic experiments.

Representation Self-Challenging (RSC) [29]

- **Key:** Explore the discrepancy between ID and OOD samples by sufficiently developing label-related information.
- **Insight:** Discarding the dominant features activated on the training data can force the network to activate remaining features that correlate with labels.
- **Model:** Masking out the bits associated with larger gradients,

$$\widetilde{\mathbf{z}} = \mathbf{z} \odot \mathbf{m}$$



[29] Z. Huang, H. Wang, E. P. Xing, and D. Huang, Self-challenging improves cross-domain generalization, ECCV, pp. 124–140, 2020.

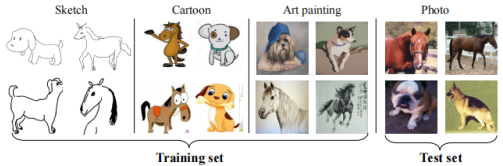
Experiments

PACS	backbone	artpaint	cartoon	sketch	photo	Avg ↑
Baseline[4]	AlexNet	66.68	69.41	60.02	89.98	71.52
Hex[31]	AlexNet	66.80	69.70	56.20	87.90	70.20
PAR[30]	AlexNet	66.30	66.30	64.10	89.60	72.08
MetaReg[1]	AlexNet	69.82	70.35	59.26	**91.07**	72.62
Epi-FCR[14]	AlexNet	64.70	72.30	65.00	86.10	72.00
JiGen[4]	AlexNet	67.63	71.71	65.18	89.00	73.38
MASF[7]	AlexNet	70.35	72.46	**67.33**	90.68	75.21
RSC(ours)	AlexNet	**71.62**	**75.11**	66.62	90.88	**76.05**
Baseline[4]	ResNet18	78.96	73.93	70.59	**96.28**	79.94
MASF[7]	ResNet18	80.29	77.17	71.69	94.99	81.03
Epi-FCR[14]	ResNet18	82.10	77.00	73.00	93.90	81.50
JiGen[4]	ResNet18	79.42	75.25	71.35	96.03	80.51
MetaReg[1]	ResNet18	**83.70**	77.20	70.30	95.50	81.70
RSC(ours)	ResNet18	83.43	**80.31**	**80.85**	95.99	**85.15**
Baseline[4]	ResNet50	86.20	78.70	70.63	97.66	83.29
MASF[7]	ResNet50	82.89	80.49	72.29	95.01	82.67
MetaReg[1]	ResNet50	87.20	79.20	70.30	97.60	83.60
RSC(ours)	ResNet50	**87.89**	**82.16**	**83.35**	**97.92**	**87.83**

DG results on PACS.

Domain Adaptation [30]

- **Task:** learn a model from the source domain that can generalize to a target domain.
- **Assumption:**
  - Samples from source and target domains are non-IID.
  - Few samples from the target domain are available in the training phase.
- **Challenge:** explore the domain discrepancy and non-IIDness between source and target domains.



[30] J. Wang, C. Lan, C. Liu, Y. Ouyang, and T. Qin, Generalizing to unseen domains: A survey on domain generalization, IJCAI, pp. 4627–4635, 2021.

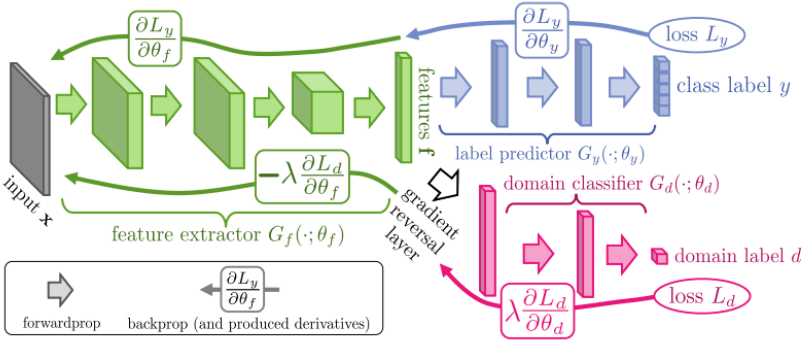
Domain-Adversarial Neural Network [31]

- **Key:** Explore the domain discrepancy between source and target domains by extracting the shared knowledge between the two domains.
- **Insight:** adversarially trains the generator and discriminator to find a representation such that the domains cannot be distinguished from each other while correctly classifying the source samples.
- **Model:**

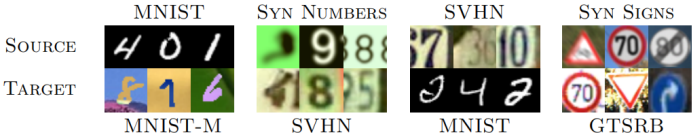
$$\widetilde{E}(\theta_f, \theta_y, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_y(G_y(G_f(\mathbf{x}_i; \theta_f); \theta_y), y_i)$$
$$- \lambda \left(\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_d(G_d(\mathcal{R}(G_f(\mathbf{x}_i); \theta_f); \theta_d), d_i) + \frac{1}{n'}\sum_{i=n+1}^{N} \mathcal{L}_d(G_d(\mathcal{R}(G_f(\mathbf{x}_i); \theta_f); \theta_d), d_i) \right).$$

[31] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res., vol. 17, pp. 1–59, 2016.

Architecture



Experiments



Examples of domain pairs.

Method	Source	MNIST	Syn Numbers	SVHN	Syn Signs
	Target	MNIST-M	SVHN	MNIST	GTSRB
Source only		.5225	.8674	.5490	.7900
SA (Fernando et al., 2013)		.5690 (4.1%)	.8644 (−5.5%)	.5932 (9.9%)	.8165 (12.7%)
DANN		**.7666** (52.9%)	**.9109** (79.7%)	**.7385** (42.6%)	**.8865** (46.4%)
Train on target		.9596	.9220	.9942	.9980

Classification accuracies for digit image classifications for different source and target domains.

Deep CORAL [32]

- **Key:** Explore the domain discrepancy between source and target domains by aligning the correlations between layer activations in networks.
- **Model:**

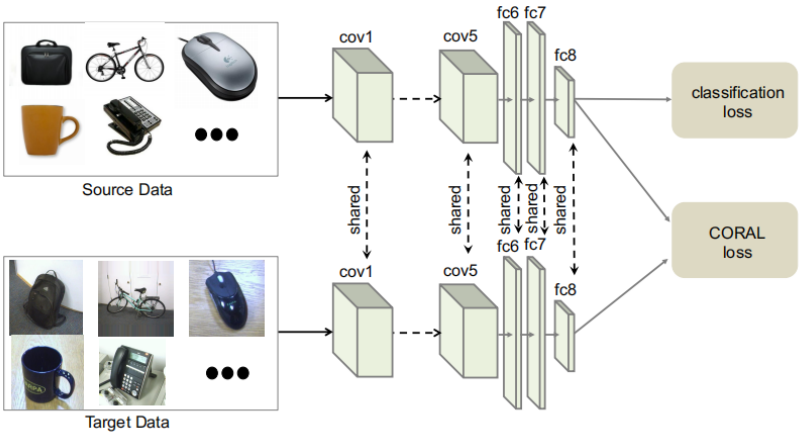
$$l_{\text{CORAL}} = \frac{1}{4d^2}\|C_S - C_T\|_F^2$$

$$C_S = \frac{1}{n_S - 1}\left(D_S^T D_S - \frac{1}{n_S}\left(\mathbf{1}^T D_S\right)^T \left(\mathbf{1}^T D_S\right)\right)$$

$$C_T = \frac{1}{n_T - 1}\left(D_T^T D_T - \frac{1}{n_T}\left(\mathbf{1}^T D_T\right)^T \left(\mathbf{1}^T D_T\right)\right)$$

[32] B. Sun and K. Saenko, Deep CORAL: correlation alignment for deep domain adaptation, ECCV Workshops, pp. 443–450, 2016.

Deep CORAL



Sample Deep CORAL architecture based on a CNN with a classifier layer.

Experiments

	A→D	A→W	D→A	D→W	W→A	W→D	AVG
GFK	52.4±0.0	54.7±0.0	43.2±0.0	92.1±0.0	41.8±0.0	96.2±0.0	63.4
SA	50.6±0.0	47.4±0.0	39.5±0.0	89.1±0.0	37.6±0.0	93.8±0.0	59.7
TCA	46.8±0.0	45.5±0.0	36.4±0.0	81.1±0.0	39.5±0.0	92.2±0.0	56.9
CORAL	65.7±0.0	64.3±0.0	48.5±0.0	**96.1**±0.0	48.2±0.0	**99.8**±0.0	70.4
CNN	63.8±0.5	61.6±0.5	51.1±0.6	95.4±0.3	49.8±0.4	99.0±0.2	70.1
DDC	64.4±0.3	61.8±0.4	52.1±0.8	95.0±0.5	**52.2**±0.4	98.5±0.4	70.6
DAN	65.8±0.4	63.8±0.4	**52.8**±0.4	94.6±0.5	51.9±0.5	98.8±0.6	71.3
D-CORAL	**66.8**±0.6	**66.4**±0.4	**52.8**±0.2	95.7±0.3	51.5±0.3	99.2±0.1	**72.1**

Object recognition accuracy for all 6 domain shifts on the standard Office dataset with deep features.

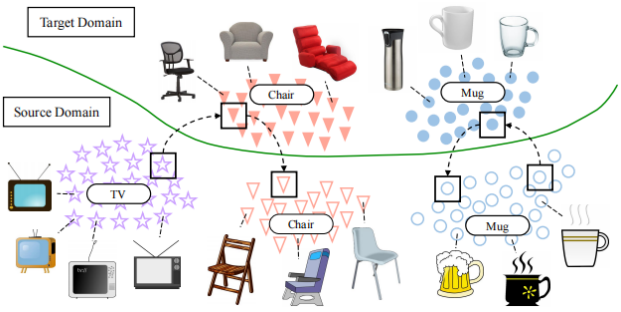
Adversarial Partial Domain Adaptation by Cycle Inconsistency [33]

- **Partial domain adaptation (PDA):**
  - Target label space is a subset of the source label space.
  - Classes absent in the target domain as outlier classes and the other classes as shared classes.
- **Challenge:** A transfer model performs even worse than a source-only model which is trained solely in the source domain.
- **Key:** Explore the non-IIDness between source and target domains by exploiting the cycle inconsistency.

[33] K.-Y. Lin, J Zhou, Y. Qiu, and W.-S. Zheng, Adversarial Partial Domain Adaptation by Cycle Inconsistency, ECCV, pp. 530-548, 2022.

Insight

- It is impossible for a source sample of outlier classes to find a target sample of the same category due to the absence of outlier classes in the target domain.
- It is possible for a source sample of shared classes.



Model

- Sample weight:

$$w_i^s = G(T_{t\to s}(T_{s\to t}(F(\mathbf{x}_i^s))))[y_i^s] + \lambda_w e_i^s G(T_{s\to t}(F(\mathbf{x}_i^s)))[y_i^s].$$

- Cross-domain feature transformation functions:

$$T_{s\to t}(\mathbf{z}^s) = \sum_{k=1}^{K} \frac{e^{\mathrm{sim}(\mathbf{z}^s, \mathbf{c}_k^t)}}{\sum_{l=1}^{K} e^{\mathrm{sim}(\mathbf{z}^s, \mathbf{c}_l^t)}} \mathbf{c}_k^t, \;\; T_{t\to s}(\mathbf{z}^t) = \sum_{k=1}^{|\mathcal{C}_s|} \frac{e^{\mathrm{sim}(\mathbf{z}^t, \mathbf{c}_k^s)}}{\sum_{l=1}^{|\mathcal{C}_s|} e^{\mathrm{sim}(\mathbf{z}^t, \mathbf{c}_l^s)}} \mathbf{c}_k^s.$$

- Prototypes:

$$\mathbf{c}_k^s \leftarrow \lambda_m \mathbf{c}_k^s + \bar{\lambda}_m \frac{\sum_{i=1}^{B} \delta(y_i^s = k)\mathbf{x}_i^s}{\sum_{i=1}^{B} \delta(y_i^s = k)}, \;\; \mathbf{c}_k^t \leftarrow \lambda_m \mathbf{c}_k^t + \bar{\lambda}_m \frac{\sum_{j=1}^{B} \delta(\hat{y}_j^t = k)\mathbf{x}_j^t}{\sum_{j=1}^{B} \delta(\hat{y}_j^t = k)}.$$

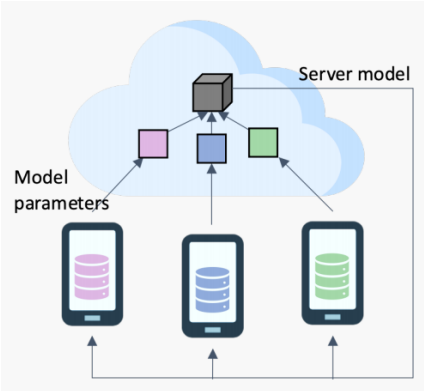
Experiments

Method	Office-31							VisDa-2017		
	A→D	A→W	D→A	D→W	W→A	W→D	Avg.	Re.→Sy.	Sy.→Re.	Avg.
ResNet-50 [22]	83.44	75.59	83.92	96.27	84.97	98.09	87.05	64.30	45.30	54.80
ADDA [59]	83.41	75.67	83.62	95.38	84.25	99.85	87.03	-	-	-
CDAN+E [43]	77.07	80.51	93.58	98.98	91.65	98.09	89.98	-	-	-
RTN [44]	66.90	75.30	85.60	97.10	85.70	98.30	84.80	72.90	50.00	61.45
†PADA [5]	89.17	88.70	94.61	99.77	95.79	100.00	94.67	69.46	62.76	66.11
†SAN [4]	94.27	93.90	94.15	99.32	88.73	99.36	94.96	69.70	49.90	59.80
†IWAN [72]	88.54	89.94	93.84	99.77	94.75	99.36	94.37	*78.18*	63.87	*71.02*
†ETN [6]	95.03	94.52	*96.21*	100.00	94.64	100.00	96.73	69.69	63.99	66.84
†MWPDA [25]	95.12	96.61	95.02	100.00	95.51	100.00	97.05	-	-	-
SSPDA [2]	90.87	91.52	90.61	92.88	94.36	98.94	93.20	-	-	-
DRCN [35]	86.00	88.05	95.60	100.00	95.80	100.00	94.30	73.20	58.20	65.70
RTNet [8]	*97.60*	96.20	92.30	100.00	95.40	100.00	96.90	-	-	-
BA3US [39]	**99.36**	*98.98*	94.82	100.00	94.99	98.73	*97.81*	-	-	-
DPDAN [26]	96.82	96.27	**96.35**	100.00	95.62	100.00	97.51	-	*65.26*	-
A2KT [29]	96.79	97.28	96.13	100.00	*96.14*	100.00	97.72	-	-	-
AdvRew [20]	91.72	97.63	95.62	100.00	95.30	100.00	96.71	-	-	-
Source-only	76.86	74.46	86.60	97.97	86.71	98.94	86.92	63.13	51.90	57.51
*DANN (baseline) [15]	59.24	56.84	70.22	82.60	86.19	90.45	74.25	50.09	44.02	47.05
†*PADA [5]	89.17	95.03	94.82	99.77	95.69	99.79	95.71	65.84	58.12	61.98
†*IWAN [72]	86.84	91.30	94.02	100.00	94.82	99.79	94.46	73.47	57.79	65.63
†*ETN [6]	84.71	87.23	94.08	98.76	94.57	98.73	93.01	67.42	60.87	64.15
Ours	96.82	**99.66**	96.14	**100.00**	**96.56**	**100.00**	**98.19**	**86.50**	**69.75**	**78.13**

Comparison with the state-of-the-art methods on Office-31 and VisDA-2017 in terms of ACC.

Non-IID Federated Learning [34] [35]

- **Task:** personalized learning on heterogeneous local data/devices without data sharing for privacy and security.
- **Assumption:**
  - A server and multiple heterogeneous and independent clients
  - iterative learning with server-client parameter messaging
- **Challenge:**
  - explore the heterogeneities between clients;
  - Some clients may be coupled or interactive.



[34] L. Cao. Non-IID Federated Learning. IEEE Intell. Syst. 37(2): 14-15, 2022

[35] A.Z. Tan, H. Yu, L. Cui, and Q. Yang, Towards personalized federated learning, IEEE Trans. Neural Networks Learn. Syst., 2022.

FedAVG [36]

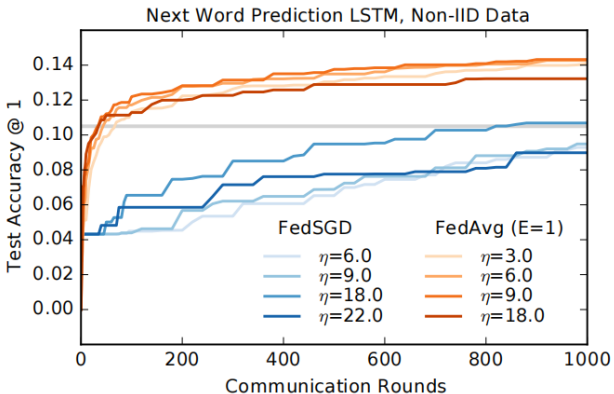
- **Key:** Explore the heterogeneity between clients by iterative model averaging.
- **Model:**

$$\min_{w \in \mathbb{R}^d} \sum_{k=1}^{K} \frac{n_k}{n} F_k(w),$$

$$F_k(w) = \frac{1}{n_k} \sum_{i \in p_k} f_i(w).$$

[36] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, Communication-Efficient Learning of Deep Networks from Decentralized Data, AISTATS, pp. 1273-1282, 2017.

Experiments



Monotonic learning curves for the large-scale language model word LSTM.

Personalized FedAvg [37]

- **Key:** Explore the heterogeneity between clients by iterative model averaging and model-agnostic meta-learning.
- **Model:**

$$\min_{w\in\mathbb{R}^d} \sum_{k=1}^{K} \frac{n_k}{n} F_k(w),$$

$$F_k(w) = \frac{1}{n_k} \sum_{i\in p_k} f_i(w - \alpha \nabla f_i(w)).$$

[37] A. Fallah, A. Mokhtari, and A. E. Ozdaglar, Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach, NeurIPS, pp. 1-12, 2020.

Experiments

Dataset	Parameters	Algorithms		
		FedAvg + update	Per-FedAvg (FO)	Per-FedAvg (HF)
MNIST	$\tau = 10, \alpha = 0.01$	$75.96\% \pm 0.02\%$	$78.00\% \pm 0.02\%$	$79.85\% \pm 0.02\%$
	$\tau = 4, \alpha = 0.01$	$60.18\ \% \pm 0.02\%$	$64.55\% \pm 0.02\%$	$70.94\% \pm 0.03\%$
CIFAR-10	$\tau = 10, \alpha = 0.001$	$40.49\% \pm 0.07\%$	$\mathbf{46.98\% \pm 0.1\%}$	$\mathbf{50.44\% \pm 0.15\%}$
	$\tau = 4, \alpha = 0.001$	$38.38\% \pm 0.07\%$	$34.04\% \pm 0.08\%$	$\mathbf{43.73\% \pm 0.11\%}$
	$\tau = 4, \alpha = 0.01$	$35.97\% \pm 0.17\%$	$25.32\% \pm 0.18\%$	$\mathbf{46.32\% \pm 0.12\%}$
	$\tau = 4, \alpha = 0.01$, diff. hetero.	$58.59\% \pm 0.11\%$	$37.71\% \pm 0.23\%$	$\mathbf{71.25\% \pm 0.05\%}$

Comparison of test accuracy of different algorithms given different parameters.

pFedMe [38]

- **Key:** Explore the heterogeneity between clients by Moreau envelopes.
- **Model:**

$$\min_{w\in\mathbb{R}^d} \sum_{k=1}^{K} \frac{n_k}{n} F_k(w),$$

$$F_k(\theta_k) = \frac{1}{n_k} \sum_{i\in p_k} f_i(\theta_k) + \frac{\lambda}{2}\|\theta_k - w\|^2.$$

[38] C. T. Dinh, N. H. Tran, and T. D. Nguyen, Personalized Federated Learning with Moreau Envelopes, NeurIPS, pp. 1-12, 2020.

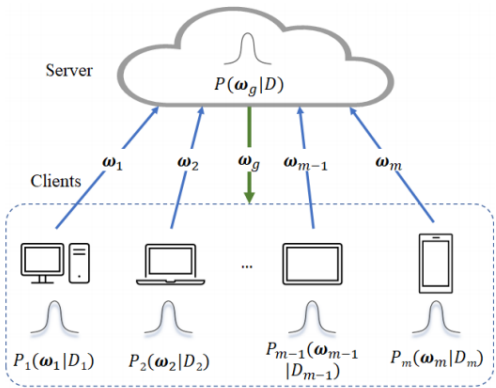
Experiments

Algorithm	Model	MNIST			Synthetic		
		λ	$\eta\,(\hat{\alpha}, \hat{\beta})$	Accuracy (%)	λ	$\eta\,(\hat{\alpha}, \hat{\beta})$	Accuracy (%)
FedAvg	MLR		0.02	93.96 ± 0.02		0.02	77.62 ± 0.11
Per-FedAvg	MLR		$0.03, 0.003$	94.37 ± 0.04		$0.02, 0.002$	81.49 ± 0.09
pFedMe-GM	MLR	15	0.01	94.18 ± 0.06	20	0.01	78.65 ± 0.25
pFedMe-PM	MLR	15	0.01	$\mathbf{95.62} \pm 0.04$	20	0.01	$\mathbf{83.20} \pm 0.06$
FedAvg	DNN		0.02	98.79 ± 0.03		0.03	83.64 ± 0.22
Per-FedAvg	DNN		$0.02, 0.001$	98.90 ± 0.02		$0.01, 0.001$	85.01 ± 0.10
pFedMe-GM	DNN	30	0.01	99.16 ± 0.03	30	0.01	84.17 ± 0.35
pFedMe-PM	DNN	30	0.01	$\mathbf{99.46} \pm 0.01$	30	0.01	$\mathbf{86.36} \pm 0.15$

Comparison using fine-tuned hyperparameters.

Bayesian Federated Learning [39]

- **Key:** Exploring non-IIDnesses in federated systems by Bayesian learning.
- **Task:** stronger model robustness and learning improved performance on small-scale data.
- **Challenge:** integrates the advantages of Bayesian learning into Federated Learning.



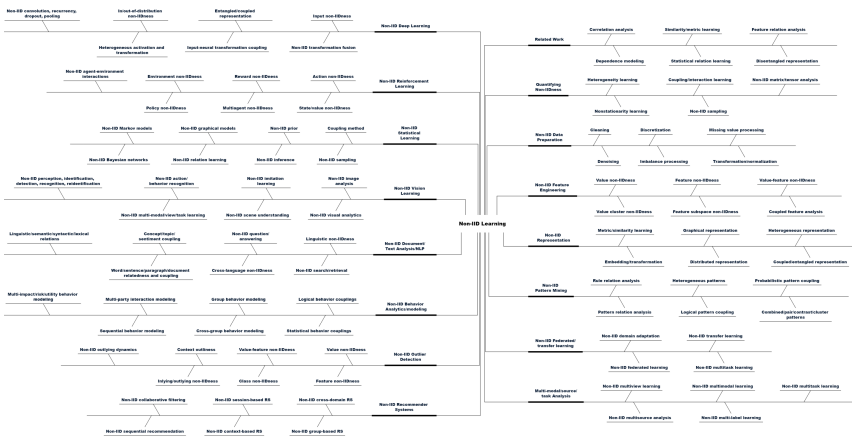
[39] L. Cao, H. Chen, X. Fan, J. Gama, Y. Ong, and V. Kumar, Bayesian Federated Learning: A Survey, IJCAI, 2023.

Contents

Non-IID Learning: A Challenging Problem

- Data Non-IIDnesses
- Data Sampling biases
- Non-IID Metrics
- Non-IID Representations
- Model Structure
- Objective Functions
- Result Interpretation
- New Perspectives

IID to Non-IID Learning Systems[40]



[40] L. Cao, P. S. Yu, Z. Zhao: Shallow and Deep Non-IID Learning on Complex Data. KDD 2022: 4774-4775

Further Research Questions: Non-IID Learning

- How do non-IIDnesses present in a system or its behaviors and data?
- How to measure and evaluate whether a dataset is non-IID?
- Do deep neural networks capture non-IIDnesses? To what extent?
- How to design an DNN to explore a specific non-IIDness from data?

Further Research Questions: Deep Learning

- **Distribution Discrepancy Estimation:** How to evaluate the couplings between two datasets and the couplings between samples?
- **Federated learning:** How to consider the non-IIDnesses within and between weakly coupled/interactive local sources, tasks, and models?
- **OOD detection:** How to measure the non-IIDnesses including/beyond distributional discrepancy between ID and OOD samples? How to measure the non-IIDnesses between OOD samples with semantic and covariate shifts?
- **Domain Adaptation:** How to measure the non-IIDnesses between source and target domains? How to decide whether the knowledge from the source domain can be transferred to the target domain according to the non-IIDnesses?

Relevant Resources

- Non-IID Learning: https://datasciences.org/non-iid-learning/
- KDD'2022 tutorial Shallow and Deep Non-IID Learning on Complex Data, KDD'2022
- IJCAI2019 tutorial Non-IID Learning of Complex Data and Behaviors
- KDD2017 tutorial on Non-IID Learning, with Tutorial Slides; and Youtube video part 1 and Youtube video part 2.

Thank you!

Comments & suggestions:
Zhilin.Zhao@mq.edu.au and Longbing.Cao@mq.edu.au
The Data Science Lab: www.datasciences.org