Label Propagation on Self-Supervised Representation Space

Qi Lei

Princeton University $\Longrightarrow$ NYU

Joint work with
Tianle Cai, Ruiqi Gao, and Jason Lee

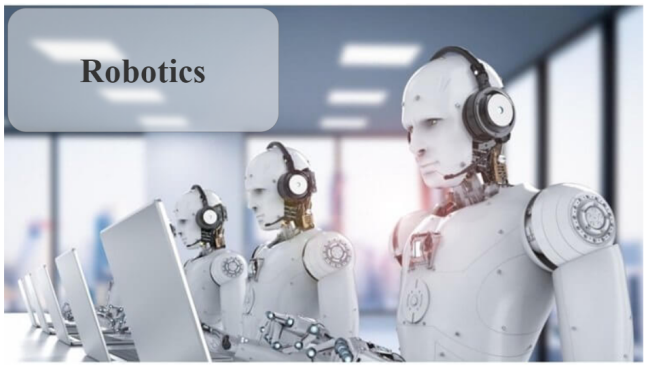
https://arxiv.org/abs/2102.11203

A.I. is Everywhere

A.I. is Everywhere

A.I. is Everywhere

A.I. is Everywhere



Downsides of the Empirical Paradigm

What Went Wrong?

- Non-robust
  - to adversarial attacks
  - to distributional shift
- Unprincipled

General target: Principled ways to handle distributional shift.

What Went Wrong?

- Non-robust
  - to adversarial attacks
  - to distributional shift
- Unprincipled

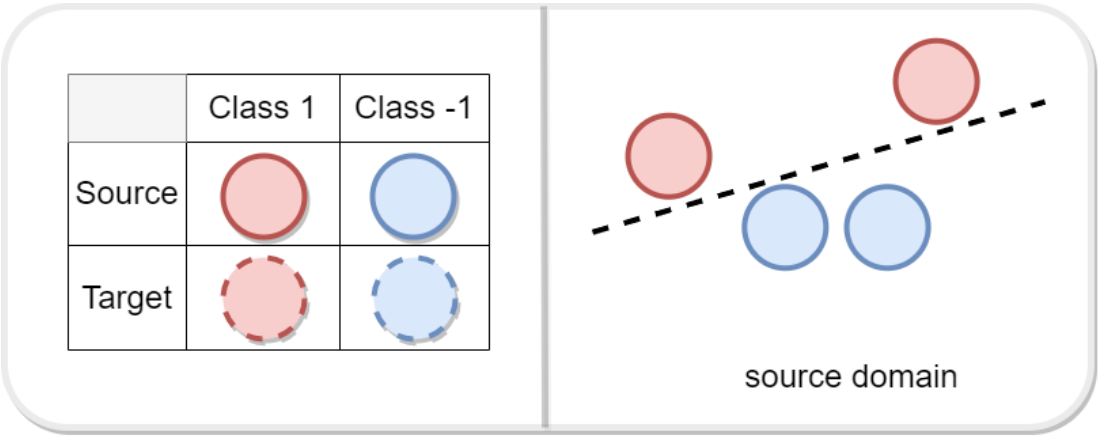
General target: Principled ways to handle distributional shift.

Outline

Types of Distribution Shifts

Distribution shift: $\mathbb{P}_S(X, Y) \neq \mathbb{P}_T(X, Y)$.

- Covariate Shift: $\mathbb{P}_S(X) \neq \mathbb{P}_T(X)$ $(\mathbb{P}_S(Y|X) = \mathbb{P}_T(Y|X))$

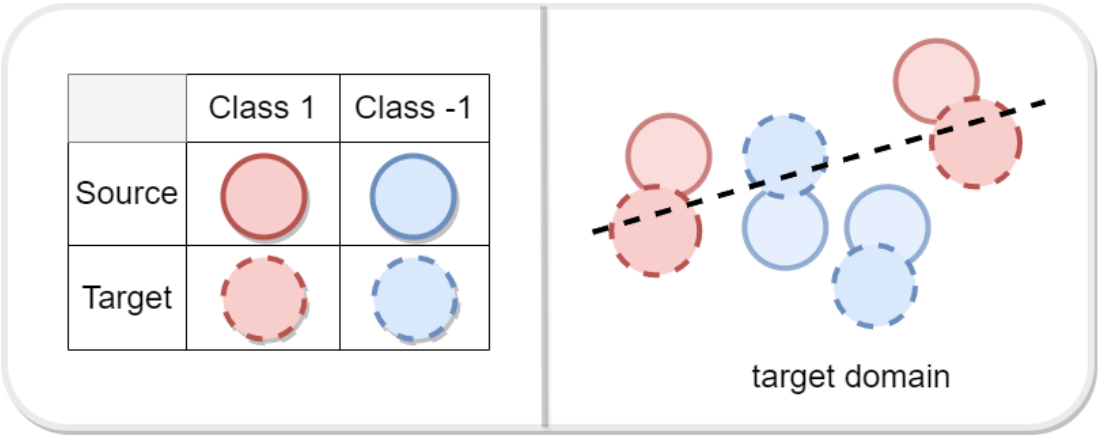


S : source domain, T : target domain

Types of Distribution Shifts

Distribution shift: $\mathbb{P}_S(X, Y) \neq \mathbb{P}_T(X, Y)$.

- Covariate Shift: $\mathbb{P}_S(X) \neq \mathbb{P}_T(X)$ $(\mathbb{P}_S(Y|X) = \mathbb{P}_T(Y|X))$



S : source domain, T : target domain

Types of Distribution Shifts

Distribution shift: $\mathbb{P}_S(X, Y) \neq \mathbb{P}_T(X, Y)$.

- Covariate Shift: $\mathbb{P}_S(X) \neq \mathbb{P}_T(X)$ $(\mathbb{P}_S(Y|X) = \mathbb{P}_T(Y|X))$
- Model Shift: $\mathbb{P}_S(Y|X) \neq \mathbb{P}_T(Y|X)$

S : source domain, T : target domain

Types of Distribution Shifts

Distribution shift: $\mathbb{P}_S(X, Y) \neq \mathbb{P}_T(X, Y)$.

- Covariate Shift: $\mathbb{P}_S(X) \neq \mathbb{P}_T(X)$ $(\mathbb{P}_S(Y|X) = \mathbb{P}_T(Y|X))$
- Model Shift: $\mathbb{P}_S(Y|X) \neq \mathbb{P}_T(Y|X)$
- Label Shift: $\mathbb{P}_S(Y) \neq \mathbb{P}_T(Y)$

S : source domain, T : target domain

Covariate shift

Source Target

Entity30 - Passerine

Source Target

Entity30 - Tableware

Our target in this talk: Principled way to handle covariate shift with additional unlabeled samples

BREEDS dataset: (Santurkar et al., 2021)

Existing Method and Their Success Scenarios

- Importance sampling:
  - $\arg\min_f \sum_i \frac{p_T(x_i)}{p_S(x_i)} \text{loss}(f(x_i), y_i)$ [1]
  - Caveat:
    1) Requires $\text{supp}(P_T) \subset \text{supp}(P_S)$.
    2) Hard to estimate the density ratio.
    3) Reduce the bias but might inflate the variance.

[1] (Shimodaira 2000, Lin et al., 2002; Zadrozny, 2004)

Existing Method and Their Success Scenarios

- Importance sampling:
  - $\arg\min_f \sum_i \frac{p_T(x_i)}{p_S(x_i)} \text{loss}(f(x_i), y_i)$
- Distributional matching:
  - Learn invariant representation (e.g. domain-adversarial algorithms) [1]
  - Caveat: Forcing representation to match may not preserve the right information for $Y|X$. [2]

[1] (Glorot et al., 2011; Ajakan et al., 2014; Long et al., 2015; Ganin et al., 2016)

[2] Failure analysis given in (Zhao et al., 2019a; Wu et al., 2019; Li et al., 2020)

Existing Method and Their Success Scenarios

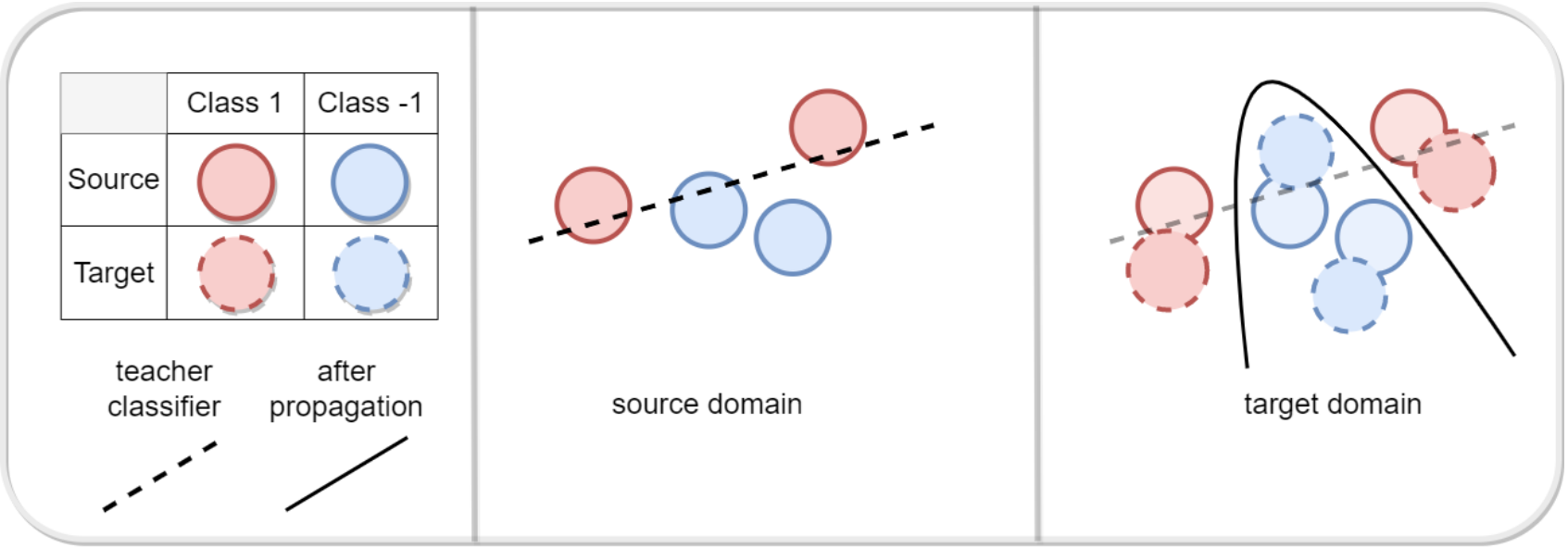
- Importance sampling:
  - $\arg\min_f \sum_i \frac{p_T(x_i)}{p_S(x_i)} \text{loss}(f(x_i), y_i)$
- Distributional matching:
  - Learn invariant representation (e.g. domain-adversarial algorithms)
- Label propagation: [1]

  How, why and when does label propagation work?

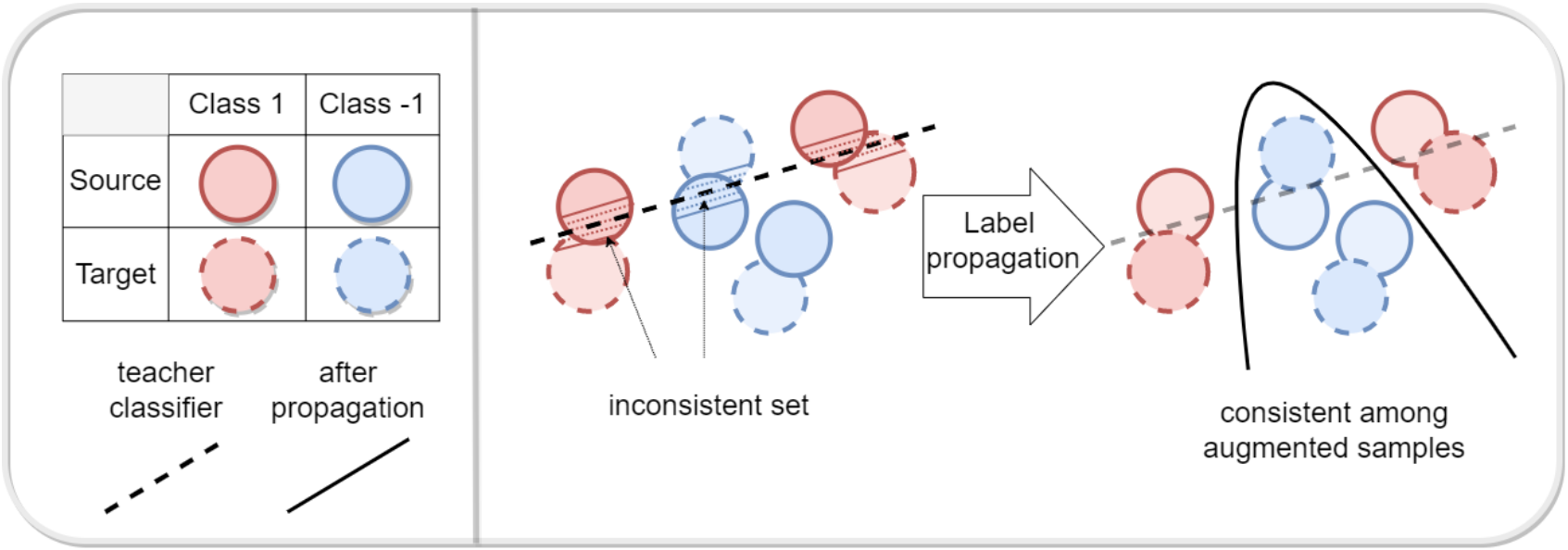
[1] (Miyato et al., 2018; Qiao et al., 2018; Xie et al., 2020)

Outline

Our New Framework: Subpopulation Shift



Our New Framework: Subpopulation Shift

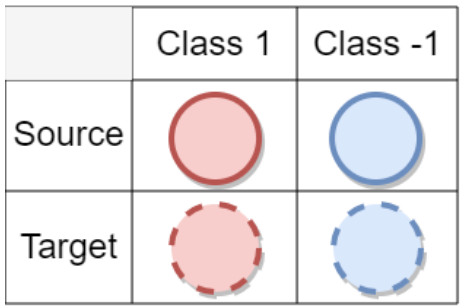


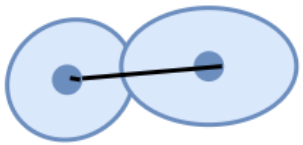
Main idea:

1) minimize the difference to the teacher classifier in the source domain, while

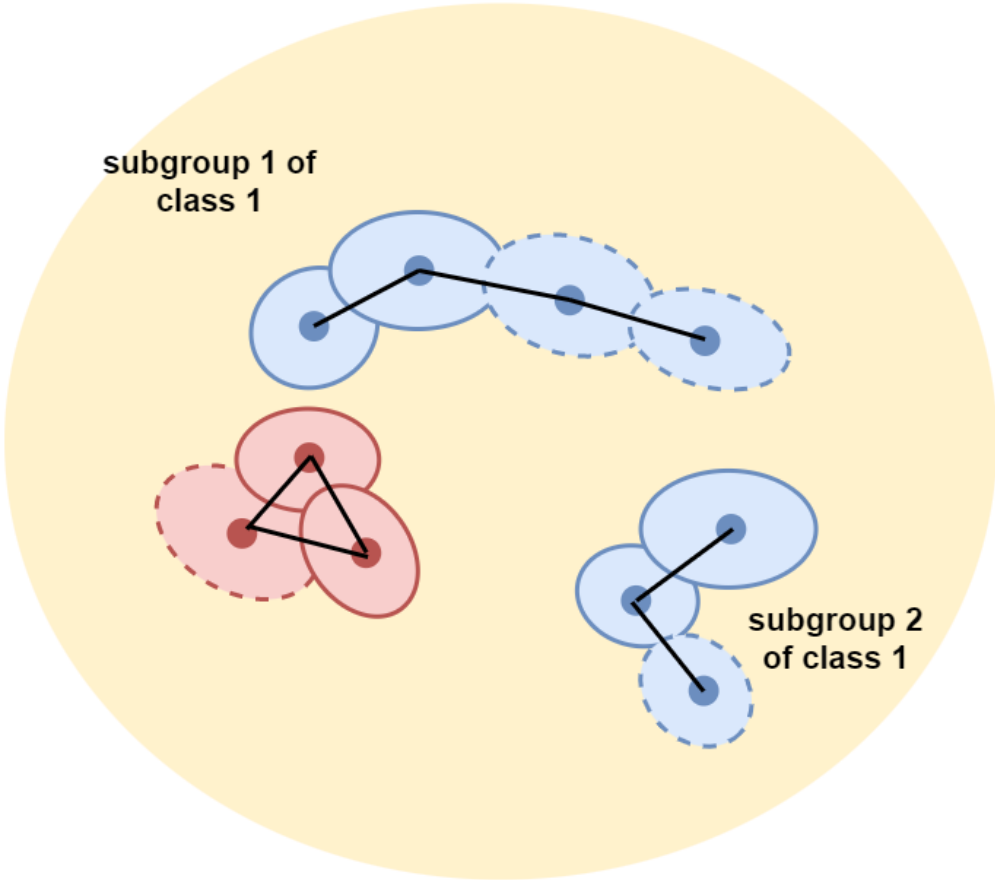
2) ensuring the samples with similar semantic meanings to predict similarly (among all unlabeled samples)

Subpopulation on the Representation Space



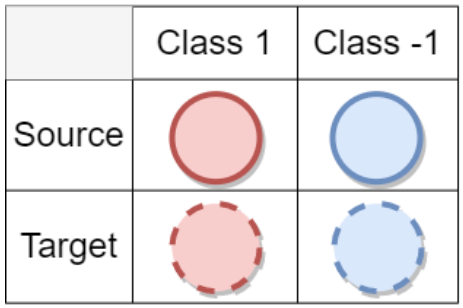


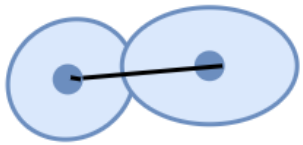
Components connected through data augmentation



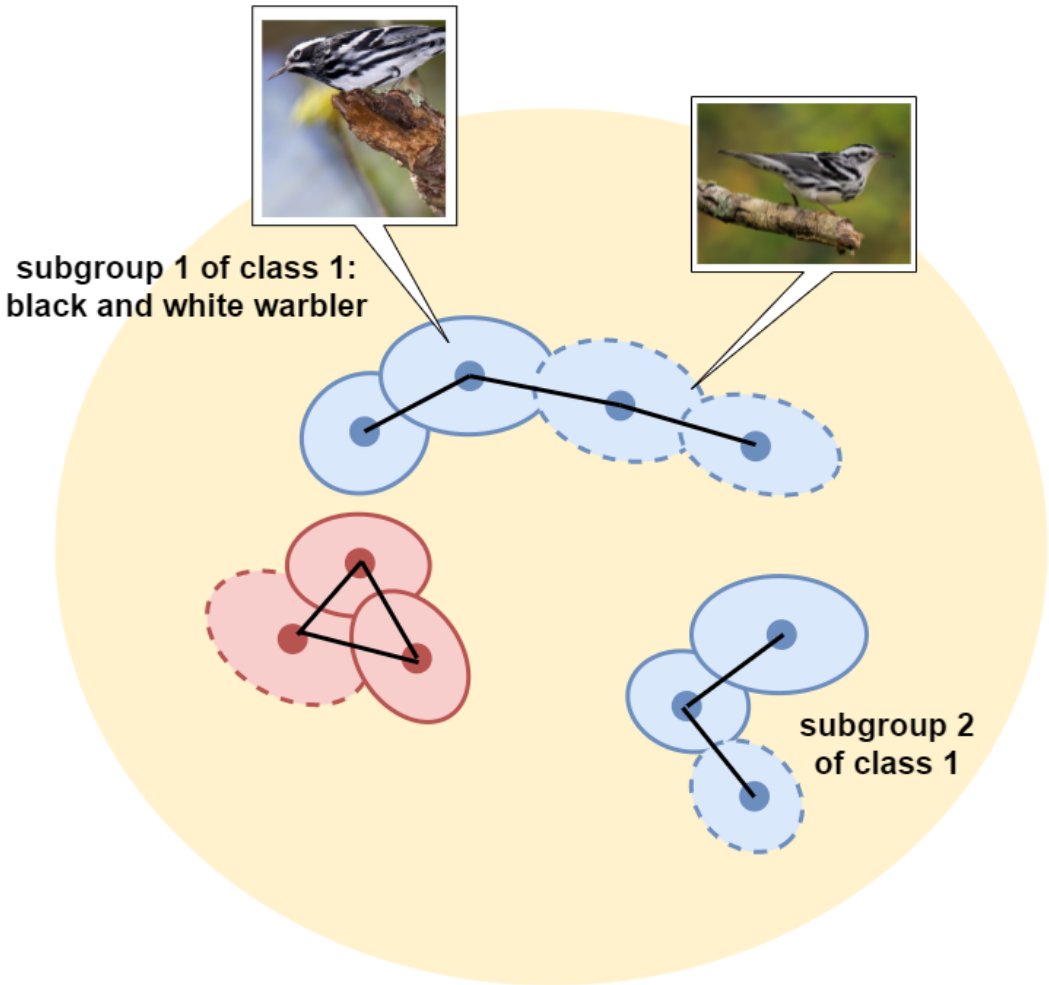
Our other work (`https://arxiv.org/abs/2008.01064`) proves how self-supervised learning learns good representation.

Subpopulation on the Representation Space





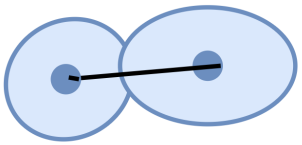
Components connected through data augmentation



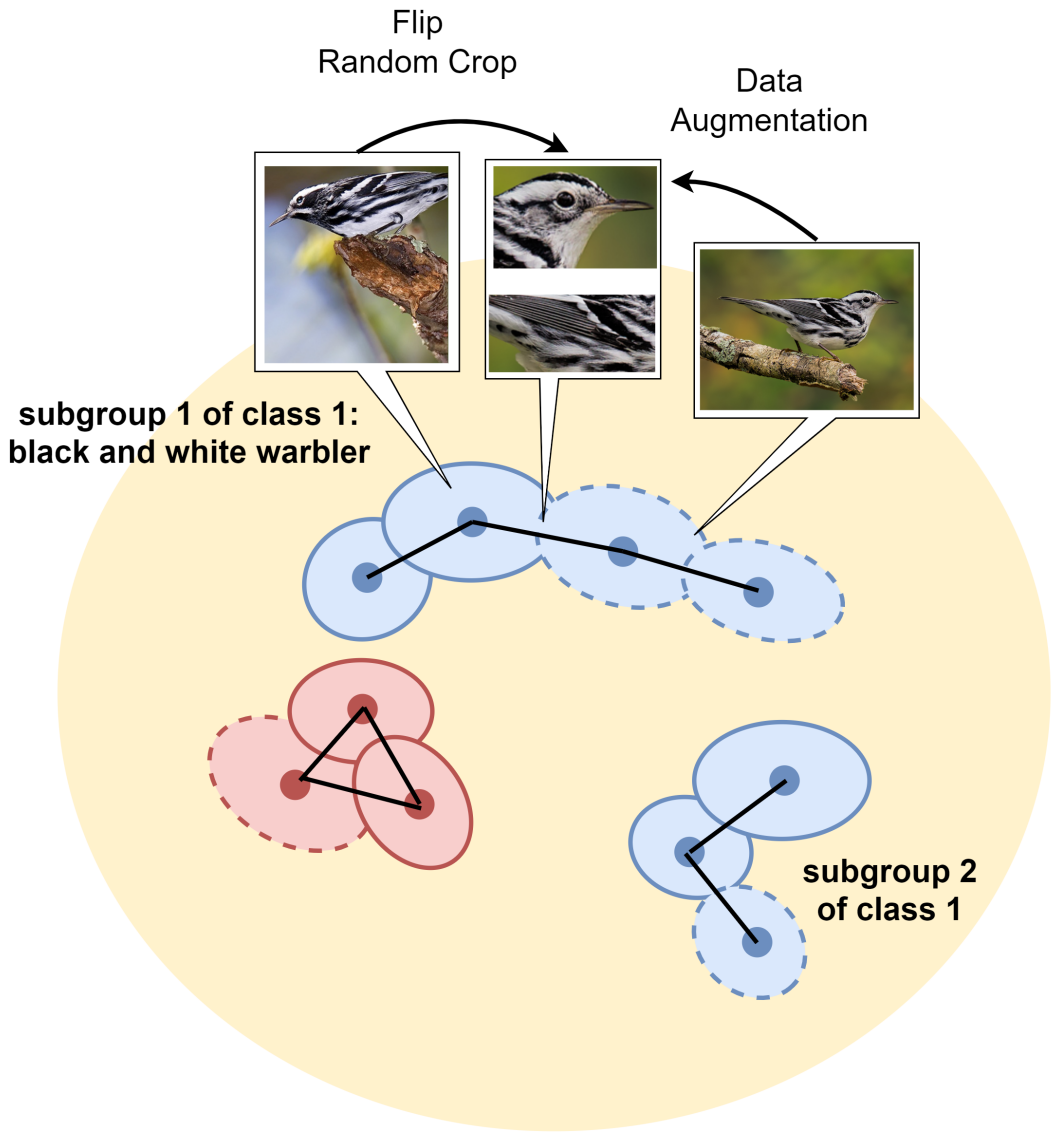
Our other work (`https://arxiv.org/abs/2008.01064`) proves how self-supervised learning learns good representation.

Subpopulation on the Representation Space

	Class 1	Class -1
Source		
Target		

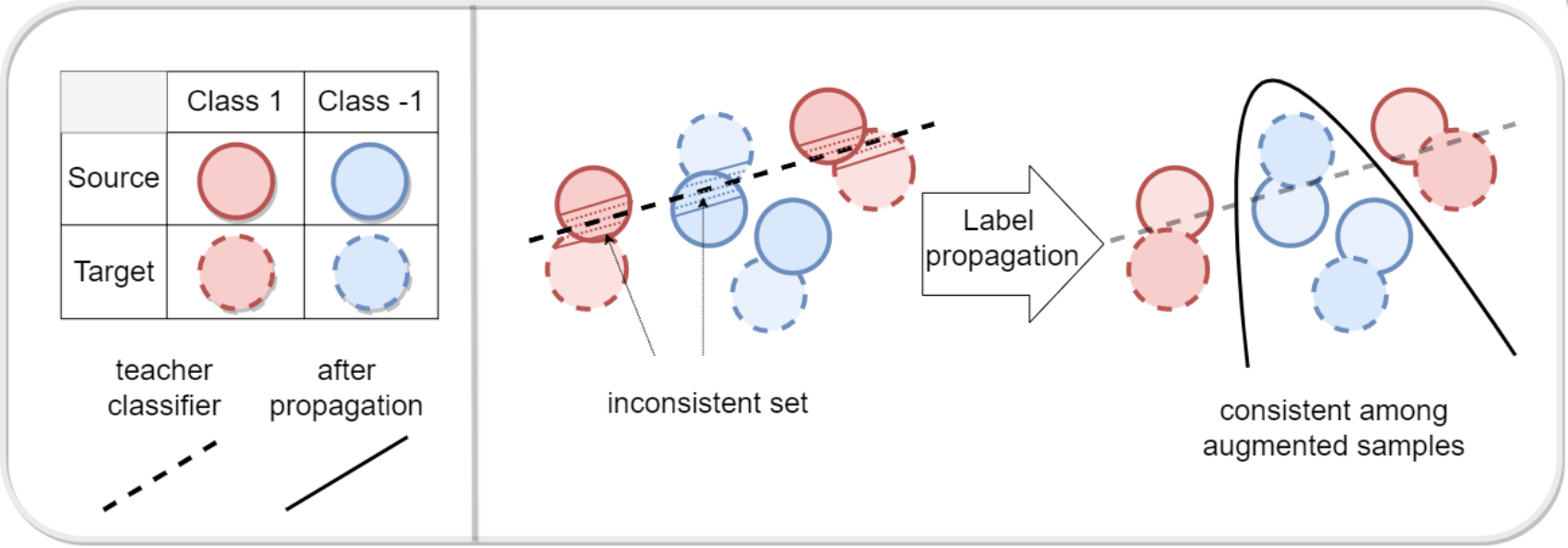


Components connected through data augmentation



Our other work (`https://arxiv.org/abs/2008.01064`) proves how self-supervised learning learns good representation.

Algorithmic Framework



- $B(x)$ maps x to the set of all its data augmentations

$$R_B(g) := P_{\frac{1}{2}(S+T)}[\exists x' \in B(x), \text{s.t. } g(x) \neq g(x')].$$

- Consistency regularizer $R_B(g)$ measures the amount of inconsistent set of g, i.e. points whose predictions are inconsistent with their augmented samples $B(\cdot)$ on g.
- teacher classifier g_{tc} + consistency regularizer $R_B(g) \Rightarrow$ Label propagation!

Algorithm

Main idea:
1) minimize the difference to the teacher classifier in the source domain, while
2) ensuring augmented data to predict similarly (among all unlabeled samples)

- Define consistency regularization:

$$R_B(g) := P_{\frac{1}{2}(S+T)}[\exists x' \in B(x), \text{s.t. } g(x) \neq g(x')].$$

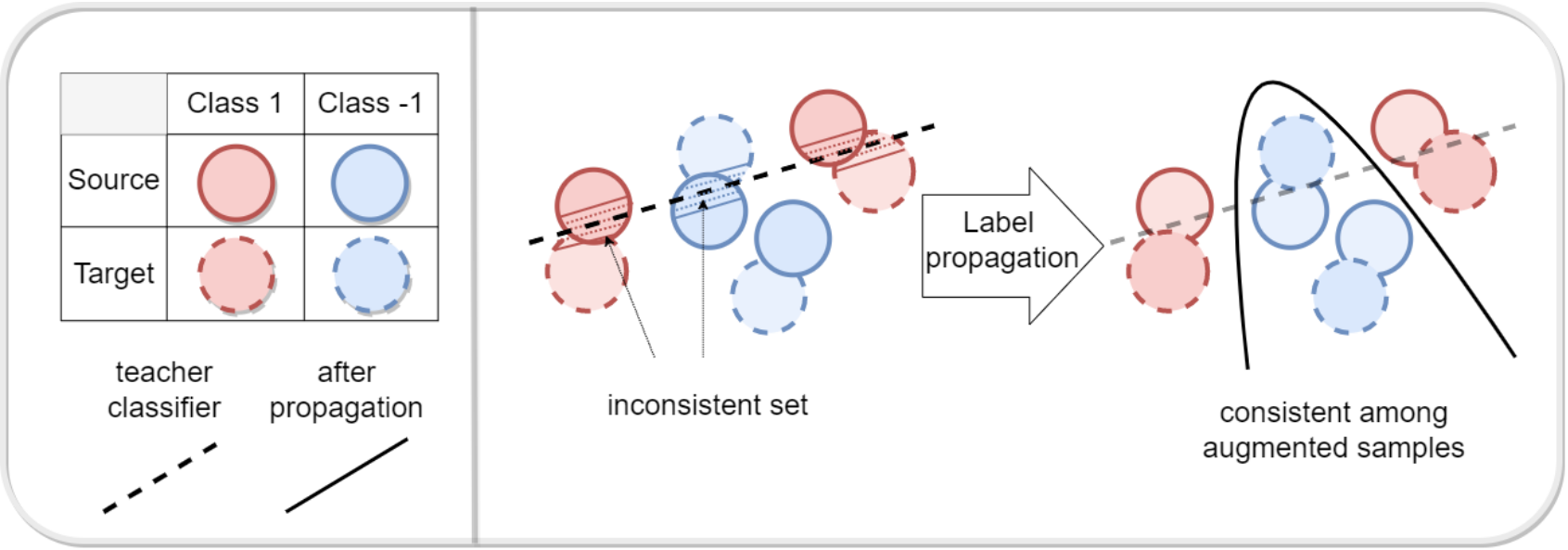
Our algorithm:

$$g \leftarrow \arg\min P_{x \sim p_S}[g(x) \neq g_{\text{tc}}(x)], \text{ s.t. } R_B(g) \leq \mu.$$

Outline

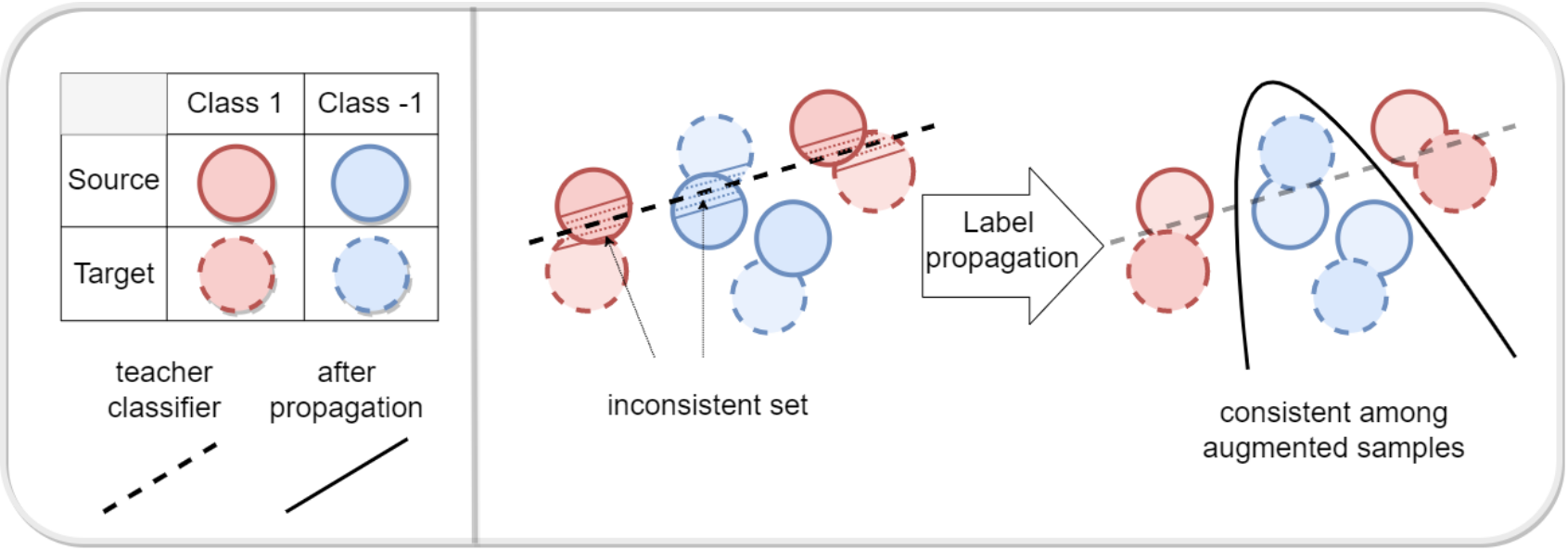
Warm-Up: Ideal Case

Ideal case: realizable robust classifier ($\exists g^*, R_B(g^*) = 0.$)



Warm-Up: Ideal Case

Ideal case: realizable robust classifier ($\exists g^*, R_B(g^*) = 0.$)



Our algorithm learns a perfect classifier (if the teacher classifier is at least 51% correct on each component).

Beyond Perfect Case

With mis-specification: data augmentation might connect two different classes by mistake. ($R_B(g^*) = \mu > 0$.)
Main idea: majority voting among the ambiguous region.

Beyond Perfect Case

With mis-specification: data augmentation might connect two different classes by mistake. ($R_B(g^*) = \mu > 0$.)
Main idea: majority voting among the ambiguous region.

Technical assumption: $(a, c)-$expansive data augmentation:[a]

- Data augmentation enlarges any small enough set to c times larger.

[a](Wei et al. 2021)

Beyond Perfect Case

With mis-specification: data augmentation might connect two different classes by mistake. ($R_B(g^*) = \mu > 0$.)
Main idea: majority voting among the ambiguous region.

Technical assumption: $(a, c)-$expansive data augmentation:[a]

- Data augmentation enlarges any small enough set to c times larger.
- For each connected component C_i, for any set $A \subset C_i$, if
  $P_{C_i}(A) := P(x \in A | x \in C_i) \leq a \Rightarrow P_{C_i}[B(A)] \geq \min\{(c+1)P_{C_i}[A], 1\}$.

[a] (Wei et al. 2021)

Main theorem

Theorem

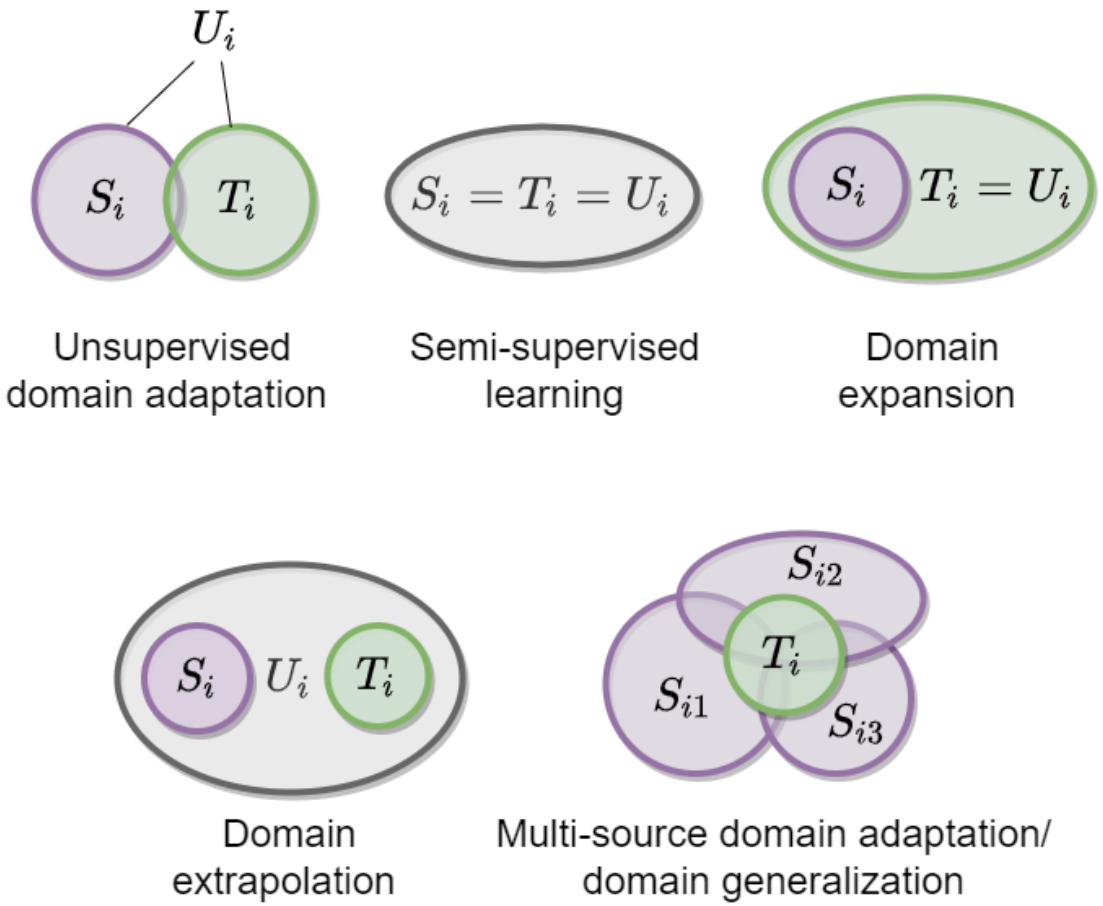
With $(\frac{1}{2}, c)$-expansive data augmentation, the target error of the classifier g returned by the algorithm is bounded by:

$$\textit{Risk} := P_{x\sim p_T}[g(x) \neq g^*(x)] \leq O(\frac{\mu}{c}),$$

where $\mu := R_B(g^*)$.

Remark: naturally extend to finite-sample results with standard analysis

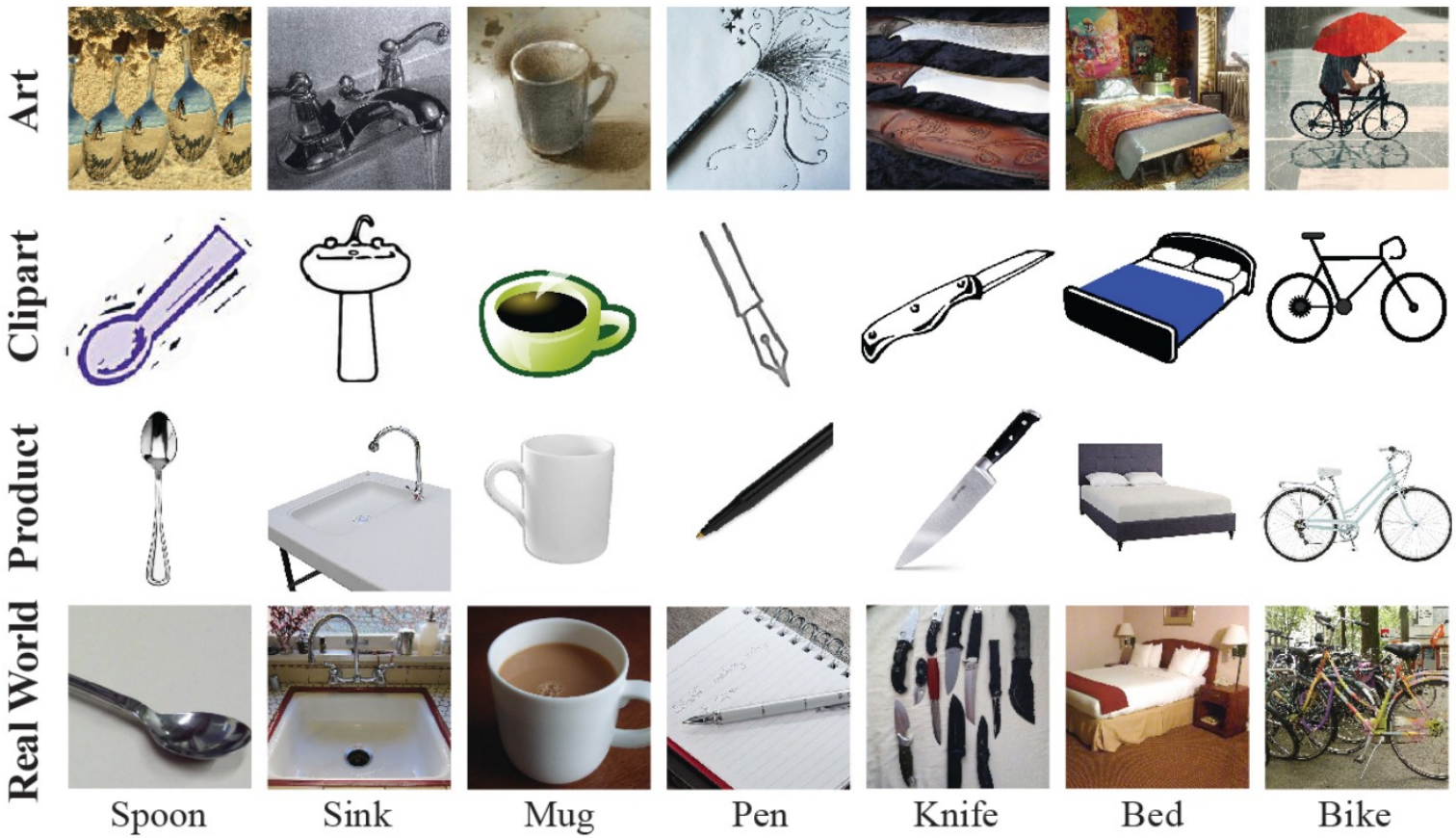
Generalized Subpopulation Shift



- The previous result holds on the general setting where the connected components of unlabeled samples "covers" S and T, on which we perform label propagation.

Outline

Datasets: Office Home



Office Home dataset: (Venkateswara et al., 2017)

Experiments

Method	A → W	D → W	W → D	A → D	D → A	W → A	Average
MDD	94.97±0.70	98.78±0.07	100±0	92.77±0.72	75.64±1.53	72.82±0.52	89.16
Ours	95.47±0.95	98.32±0.19	100±0	93.71±0.23	76.64±1.91	74.93±1.15	**89.84**

Performance of MDD[2] and our method on Office-31 dataset.

Method	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar
MDD	54.9±0.7	74.0±0.3	77.7±0.3	60.6±0.4	70.9±0.7	72.1±0.6	60.7±0.8
Ours	55.1±0.9	74.7±0.8	78.7±0.5	63.2±1.3	74.1±1.8	75.3±0.1	63.0±0.6
Method	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Average	
MDD	53.0±1.0	78.0±0.2	71.8±0.4	59.6±0.4	82.9±0.3	68.0	
Ours	53.0±0.6	80.8±0.4	73.4±0.1	59.4±0.7	84.0±0.5	**69.6**	

Performance of MDD and our method on Office-Home dataset.

[2] Margin Disparity Discrepancy (MDD) (Zhang et al. 2019)
A Theory of Label Propagation for Subpopulation Shift, ICML 2021

Datasets: Breeds

Source Target

Entity30 - Passerine

Source Target

Entity30 - Tableware

BREEDS dataset: (Santurkar et al., 2021)

Experiments: Subpopulation Shift Dataset

- ENTITY-30 task from BREEDS tasks

Method	Source Acc	Target Acc
Train on Source	91.91±0.23	56.73±0.32
DANN (Ganin et al. 2016)	92.81±0.50	61.03±4.63
MDD (Zhang et al. 2019)	92.67±0.54	63.95±0.28
Ours	90.87±0.15	72.60±0.51

Comparison of performance on ENTITY-30.

Conclusion

Consistency regularization under data augmentation not only

- corrects mis-classified samples in semi-supervised learning, but also

Conclusion

Consistency regularization under data augmentation not only

- corrects mis-classified samples in semi-supervised learning, but also
- propagates the correct labels to other domains with unlabeled samples.

Conclusion

Consistency regularization under data augmentation not only

- corrects mis-classified samples in semi-supervised learning, but also
- propagates the correct labels to other domains with unlabeled samples.
- Self-supervised learning learns the proper metric that aids the procedure.

Thank you!