Leveraging Ensemble Diversity for Robust Self-Training

Ambroise Odonnat, **Vasilii Feofanov**, Ievgen Redko

Huawei Paris Noah's Ark Lab
École des Ponts ParisTech

CAp RFIAP 2024, July 2
Published in AISTATS 2024

Introduction

In some applications data acquisition is cheaper than labeling,

Introduction

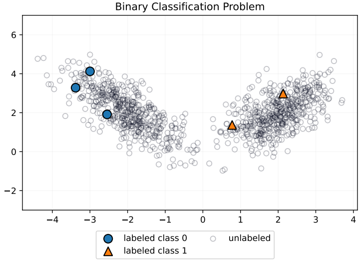
And supervised learning is inefficient.

Introduction

Semi-supervised learning: learn with both few labeled and many unlabeled training examples.

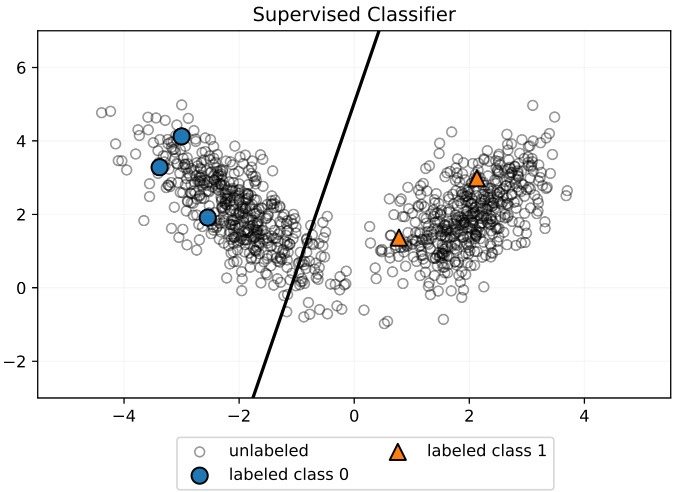
Family of SSL Methods:

- **Pseudo-labeling**,
- Graph-based algorithms,
- Cluster-then-label,
- Unsupervised feature learning.



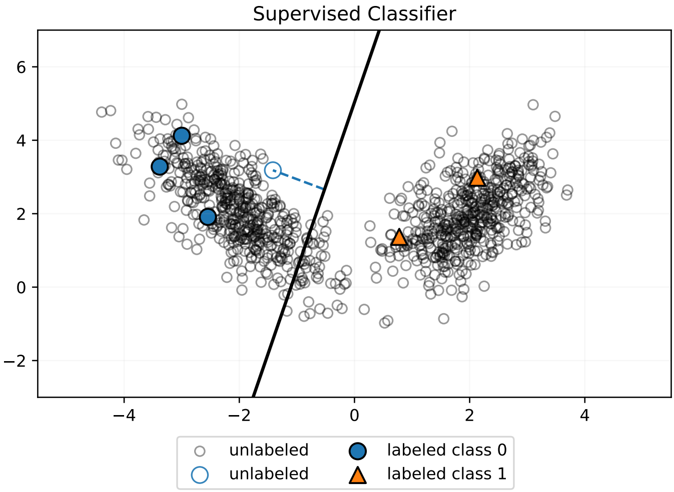
Self-training

Start from a supervised classifier trained on the labeled set.

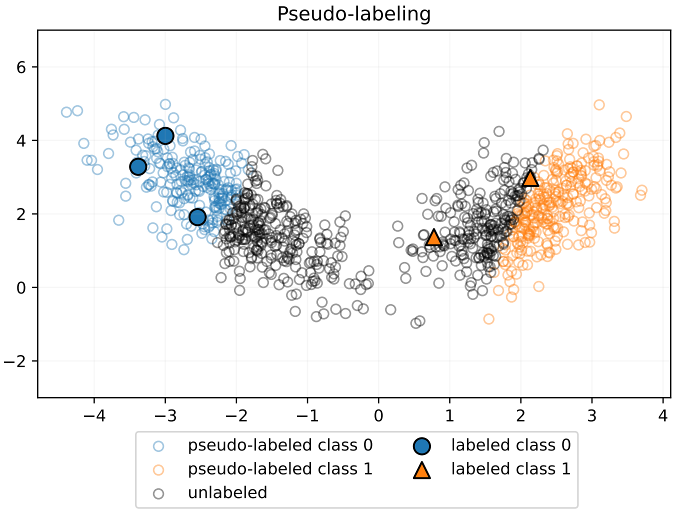


Self-training

Predict labels and confidence scores for unlabeled data.



Pseudo-label most confident data and include to the labeled set.



Retrain the model and repeat the same procedure again.

Self-training

And again...

Self-training

Until there are no data to pseudo-label.

Self-training

Self-training pushed the boundary away from the confident data

Two Fundamental Questions

1. *Confidence Estimation* $\rightarrow$ How to rank unlabeled data?

Two Fundamental Questions

1. *Confidence Estimation* $\rightarrow$ How to rank unlabeled data?
2. *Pseudo-Labeling Policy* $\rightarrow$ How to selected unlabeled data for pseudo-labeling at each iteration?
   - Questions 2 has been studied a lot (Amini et al., 2022).

Two Fundamental Questions

1. *Confidence Estimation* $\rightarrow$ How to rank unlabeled data?
2. *Pseudo-Labeling Policy* $\rightarrow$ How to selected unlabeled data for pseudo-labeling at each iteration?
   - Questions 2 has been studied a lot (Amini et al., 2022).

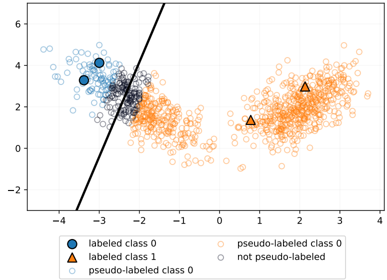
In this work, we focus on *Confidence Estimation*.

Two Fundamental Questions

1. *Confidence Estimation* → How to rank unlabeled data?
2. *Pseudo-Labeling Policy* → How to selected unlabeled data for pseudo-labeling at each iteration?
   - Questions 2 has been studied a lot (Amini et al., 2022).

In this work, we focus on *Confidence Estimation*.

Biased prediction confidence ⇒ wrong direction can be chosen.



Sample Selection Bias

- Confidence can be biased when labeled and unlabeled data are not i.i.d.

Sample Selection Bias

- Confidence can be biased when labeled and unlabeled data are not i.i.d.
- **Sample Selection Bias(SSB)**: data labeling subject to constraints

Sample Selection Bias

- Confidence can be biased when labeled and unlabeled data are not i.i.d.
- **Sample Selection Bias(SSB)**: data labeling subject to constraints
  - Creation of group study in clinical trials;
  - People with poor mobility less likely to be in street surveys;
  - Labeling can be constrained for privacy reasons.

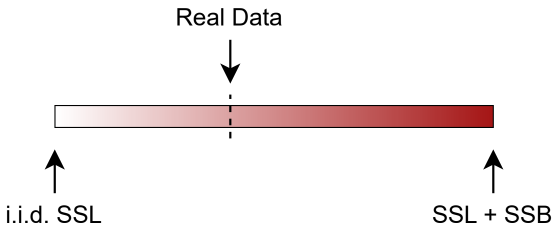
Sample Selection Bias

- Confidence can be biased when labeled and unlabeled data are not i.i.d.
- **Sample Selection Bias(SSB)**: data labeling subject to constraints
  - Creation of group study in clinical trials;
  - People with poor mobility less likely to be in street surveys;
  - Labeling can be constrained for privacy reasons.
- Studied (Zadrozny, 2004) but not in the case of SSL.

SSL under Sample Selection Bias

We consider SSL + SSB:

1. Few labeled examples (SSL)
2. Biased labeling procedure (SSB)



Goal → obtain a method good on **both** i.i.d. SSL and SSL + SSB.

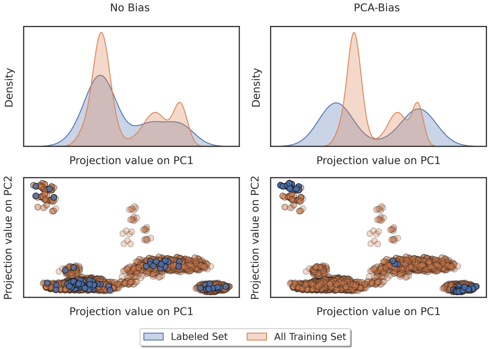
Implementation of SSL + SSB

- We select labeled data in biased manner by modeling $s_i \in \{0, 1\}$ with $\mathbb{P}(s_i | \mathbf{x}_i, y_i = k)$.

Implementation of SSL + SSB

- We select labeled data in biased manner by modeling $s_i \in \{0,1\}$ with $\mathbb{P}(s_i|\mathbf{x}_i, y_i = k)$.
- `PCA-Bias` algorithm:
  1. Apply `PCA` on training data from class k;
  2. Compute $\mathsf{PC}_1(\mathbf{x}_i)$;
  3. $\mathbb{P}(s_i = 1|\mathbf{x}_i, y_i = k) \propto \exp(r \cdot |\mathsf{PC}_1(\mathbf{x}_i)|), \quad r > 0.$

- We select labeled data in biased manner by modeling $s_i \in \{0, 1\}$ with $\mathbb{P}(s_i|\mathbf{x}_i, y_i = k)$.
- `PCA-Bias` algorithm:
  1. Apply `PCA` on training data from class k;
  2. Compute $\mathrm{PC}_1(\mathbf{x}_i)$;
  3. $\mathbb{P}(s_i = 1|\mathbf{x}_i, y_i = k) \propto \exp(r \cdot |\mathrm{PC}_1(\mathbf{x}_i)|), \quad r > 0.$

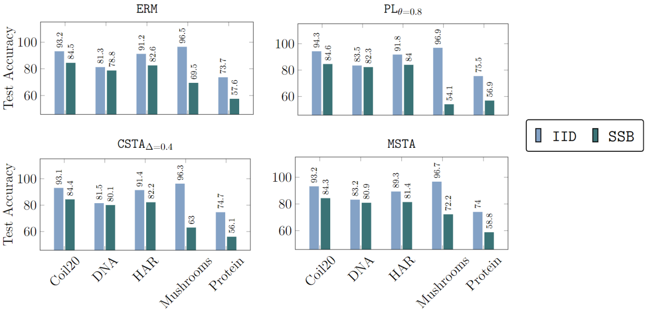


Failure of Self-training under SSL+SSB

- Base Classifier:
  - `ERM`: (MLP) learned on the labeled set;
- Self-training Policies:
  - $\texttt{PL}_{\theta=0.95}$: fixed threshold θ (Lee et al., 2013);
  - $\texttt{CSTA}_{\Delta=0.4}$: $\Delta\%$ most confident (Cascante-Bonilla et al., 2021);
  - `MSTA`: trade-off between the estimated error and amount of pseudo-labeling (Feofanov et al., 2019).

Failure of Self-training under SSL+SSB

- Base Classifier:
  - ERM: (MLP) learned on the labeled set;
- Self-training Policies:
  - $\mathrm{PL}_{\theta=0.95}$: fixed threshold θ (Lee et al., 2013);
  - $\mathrm{CSTA}_{\Delta=0.4}$: $\Delta\%$ most confident (Cascante-Bonilla et al., 2021);
  - MSTA: trade-off between the estimated error and amount of pseudo-labeling (Feofanov et al., 2019).

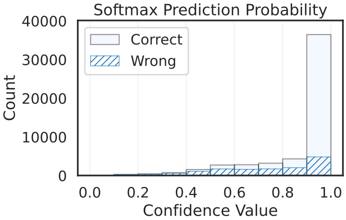


Overconfidence of Softmax

- Confidence estimation = ranking from easy to hard.

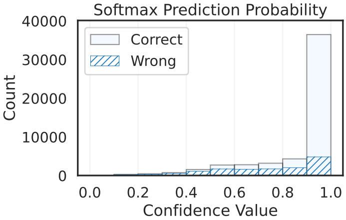
Overconfidence of Softmax

- Confidence estimation = ranking from easy to hard.
- Softmax-based confidence is unreliable:
  - Overconfident;
  - Biased towards the labeled set.

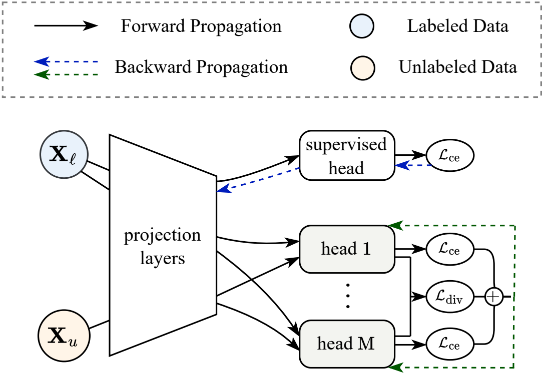


Overconfidence of Softmax

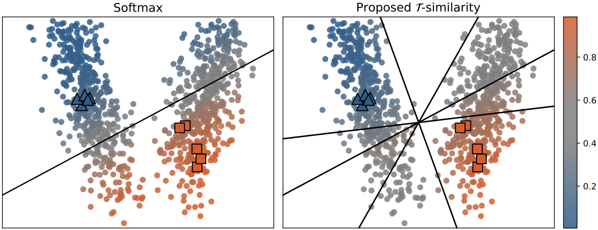
- Confidence estimation = ranking from easy to hard.
- Softmax-based confidence is unreliable:
  - Overconfident;
  - Biased towards the labeled set.



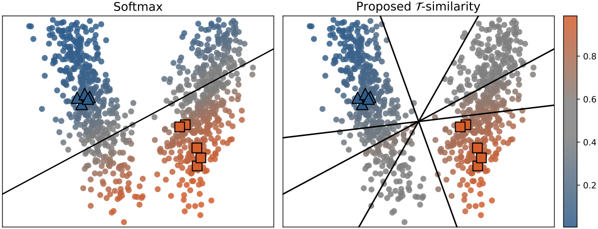
We propose new way to estimate confidence for a NN.



- Projection layers are learned through a classification head;
- Confidence estimator is ensemble of $M = 5$ linear heads that don't affect representation.



$$\min \frac{1}{M} \sum_{h \in \mathcal{T}} \underbrace{\left(\frac{1}{n_\ell} \sum_{(\mathbf{x},y) \in \mathbf{X}_\ell \times \mathbf{y}_\ell} \ell(h(\mathbf{x}), y) \right)}_{\text{supervised loss}} + \underbrace{\frac{\gamma}{n_u M(M-1)} \sum_{h \neq \tilde{h} \in \mathcal{T}} \sum_{\mathbf{x} \in \mathbf{X}_u} h(\mathbf{x})^\top \tilde{h}(\mathbf{x})}_{\text{agreement loss}}$$

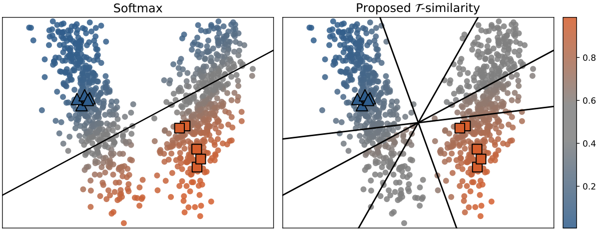


$$\min \frac{1}{M} \sum_{h \in \mathcal{T}} \underbrace{\left(\frac{1}{n_\ell} \sum_{(\mathbf{x},y) \in \mathbf{X}_\ell \times \mathbf{y}_\ell} \ell(h(\mathbf{x}), y) \right)}_{\text{supervised loss}} + \underbrace{\frac{\gamma}{n_u M(M-1)} \sum_{h \neq \tilde{h} \in \mathcal{T}} \sum_{\mathbf{x} \in \mathbf{X}_u} h(\mathbf{x})^\top \tilde{h}(\mathbf{x})}_{\text{agreement loss}}$$

We jointly train ensemble

- To fit very well the labeled data

Leveraging Ensemble Diversity



$$\min \frac{1}{M} \sum_{h \in \mathcal{T}} \underbrace{\left(\frac{1}{n_\ell} \sum_{(\mathbf{x},y) \in \mathbf{X}_\ell \times \mathbf{y}_\ell} \ell(h(\mathbf{x}), y) \right)}_{\text{supervised loss}} + \underbrace{\frac{\gamma}{n_u M(M-1)} \sum_{h \neq \tilde{h} \in \mathcal{T}} \sum_{\mathbf{x} \in \mathbf{X}_u} h(\mathbf{x})^\top \tilde{h}(\mathbf{x})}_{\text{agreement loss}}$$

We jointly train ensemble

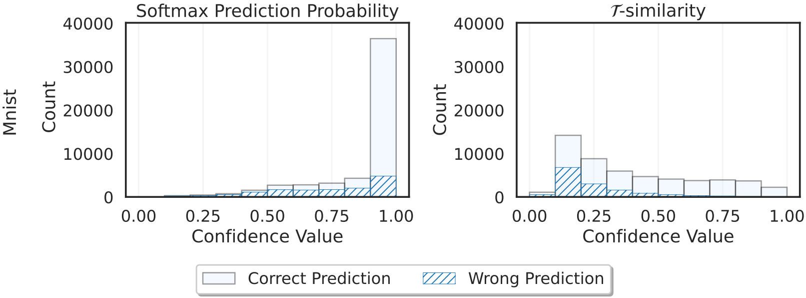
- To fit very well the labeled data
- And disagree as much possible on unlabeled data,

$\mathcal{T}$-similarity

- We define the $\mathcal{T}$-similarity as:

$$s_{\mathcal{T}}(\mathbf{x}) = \frac{1}{M(M-1)} \sum_{h \neq \tilde{h} \in \mathcal{T}} h(\mathbf{x})^\top \tilde{h}(\mathbf{x}).$$

- For any $\mathbf{x}$, we have $0 \leq s_{\mathcal{T}}(\mathbf{x}) \leq 1$.



Binary Linear Case

- Consider binary linear classification: $\mathbf{W} = \{\boldsymbol{\omega}_m \in \mathbb{R}^d | 1 \leq m \leq M\}$.

$$\underset{\mathbf{W} \in \mathbb{R}^{d \times M}}{\operatorname{argmin}} \mathcal{L}(\mathbf{W}) := \underbrace{\frac{1}{M} \sum_{m=1}^{M} \frac{1}{n_\ell} \sum_{i=1}^{n_\ell} \left(y_i - \boldsymbol{\omega}_m^\top \mathbf{x}_i\right)^2}_{\text{label fidelity term}} + \underbrace{\frac{1}{M} \sum_{m=1}^{M} \lambda_m \|\boldsymbol{\omega}_m\|^2}_{\text{regularization}}$$

$$+ \underbrace{\frac{\gamma}{M(M-1)} \sum_{m \neq k} \frac{1}{n_u} \sum_{i=n_\ell+1}^{n_\ell+n_u} w_m^\top \mathbf{x}_i w_k^\top \mathbf{x}_i}_{\text{agreement term}}$$

Binary Linear Case

- Consider binary linear classification: $\mathbf{W} = \{\boldsymbol{\omega}_m \in \mathbb{R}^d | 1 \leq m \leq M\}$.

$$\underset{\mathbf{W}\in\mathbb{R}^{d\times M}}{\operatorname{argmin}} \mathcal{L}(\mathbf{W}) \coloneqq \underbrace{\frac{1}{M}\sum_{m=1}^{M}\frac{1}{n_\ell}\sum_{i=1}^{n_\ell}\left(y_i - \boldsymbol{\omega}_m^\top \mathbf{x}_i\right)^2}_{\text{label fidelity term}} + \underbrace{\frac{1}{M}\sum_{m=1}^{M}\lambda_m \|\boldsymbol{\omega}_m\|^2}_{\text{regularization}}$$

$$+ \underbrace{\frac{\gamma}{M(M-1)}\sum_{m\neq k}\frac{1}{n_u}\sum_{i=n_\ell+1}^{n_\ell+n_u} w_m^\top \mathbf{x}_i w_k^\top \mathbf{x}_i}_{\text{agreement term}}$$

- Under the assumption

$$\forall m \in [\![1, M]\!], \lambda_m > \frac{\gamma(M+1)}{n_u(M-1)}\lambda_{max}(\mathbf{X}_u^\top \mathbf{X}_u)$$

Binary Linear Case

- Consider binary linear classification: $\mathbf{W} = \{\boldsymbol{\omega}_m \in \mathbb{R}^d | 1 \leq m \leq M\}$.

$$\underset{\mathbf{W}\in\mathbb{R}^{d\times M}}{\operatorname{argmin}} \mathcal{L}(\mathbf{W}) := \underbrace{\frac{1}{M}\sum_{m=1}^{M}\frac{1}{n_\ell}\sum_{i=1}^{n_\ell}\left(y_i - \boldsymbol{\omega}_m^\top \mathbf{x}_i\right)^2}_{\text{label fidelity term}} + \underbrace{\frac{1}{M}\sum_{m=1}^{M}\lambda_m \|\boldsymbol{\omega}_m\|^2}_{\text{regularization}}$$

$$+ \underbrace{\frac{\gamma}{M(M-1)}\sum_{m\neq k}\frac{1}{n_u}\sum_{i=n_\ell+1}^{n_\ell+n_u} w_m^\top \mathbf{x}_i w_k^\top \mathbf{x}_i}_{\text{agreement term}}$$

- Under the assumption

$$\forall m \in [\![1, M]\!], \lambda_m > \frac{\gamma(M+1)}{n_u(M-1)}\lambda_{max}(\mathbf{X}_u^\top \mathbf{X}_u)$$

- We proved that $\mathcal{L}$ is continuous, strictly convex and coercive, so the problem admits a unique solution.

Diversity of an Ensemble

- Ensemble diversity:

$$\ell_{\mathrm{div}}(\mathbf{W}, \mathbf{X}_u) = -\frac{1}{n_u M(M-1)} \sum_{m \neq k} \boldsymbol{\omega}_m^\top \mathbf{X}_u^\top \mathbf{X}_u \boldsymbol{\omega}_k.$$

Diversity of an Ensemble

- Ensemble diversity:

$$\ell_{\text{div}}(\mathbf{W}, \mathbf{X}_u) = -\frac{1}{n_u M(M-1)} \sum_{m \neq k} \boldsymbol{\omega}_m^\top \mathbf{X}_u^\top \mathbf{X}_u \boldsymbol{\omega}_k.$$

Theorem (Connection b/w optimal loss and cov. matrix of $\mathbf{X}_\ell$)

$\ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u)$ *is lower-bounded as follows:*

$$\gamma \ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u) \geq \frac{1}{2M}\left(\lambda + \frac{1}{n_\ell}\lambda_{\min}\left(\mathbf{X}_\ell^\top \mathbf{X}_\ell\right)\right)\|\mathbf{W}^*\|_{\mathrm{F}}^2.$$

Diversity of an Ensemble

- Ensemble diversity:

$$\ell_{\text{div}}(\mathbf{W}, \mathbf{X}_u) = -\frac{1}{n_u M(M-1)} \sum_{m \neq k} \boldsymbol{\omega}_m^\top \mathbf{X}_u^\top \mathbf{X}_u \boldsymbol{\omega}_k.$$

Theorem (Connection b/w optimal loss and cov. matrix of $\mathbf{X}_\ell$)

$\ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u)$ *is lower-bounded as follows:*

$$\gamma \ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u) \geq \frac{1}{2M}\left(\lambda + \frac{1}{n_\ell}\lambda_{\min}\left(\mathbf{X}_\ell^\top \mathbf{X}_\ell\right)\right)\|\mathbf{W}^*\|_{\text{F}}^2.$$

- Optimal diversity is determined by variance within labeled data.

Diversity of an Ensemble

- Ensemble diversity:

$$\ell_{\text{div}}(\mathbf{W}, \mathbf{X}_u) = -\frac{1}{n_u M(M-1)} \sum_{m \neq k} \boldsymbol{\omega}_m^\top \mathbf{X}_u^\top \mathbf{X}_u \boldsymbol{\omega}_k.$$

Theorem (Connection b/w optimal loss and cov. matrix of $\mathbf{X}_\ell$)

$\ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u)$ *is lower-bounded as follows:*

$$\gamma \ell_{\text{div}}(\mathbf{W}^*, \mathbf{X}_u) \geq \frac{1}{2M}\left(\lambda + \frac{1}{n_\ell}\lambda_{\min}\left(\mathbf{X}_\ell^\top \mathbf{X}_\ell\right)\right) \|\mathbf{W}^*\|_{\mathrm{F}}^2.$$

- Optimal diversity is determined by variance within labeled data.
- Theorem shows importance of representation learning.

Performance Results in SSL+SSB

Dataset	ERM	$\mathrm{PL}_{\theta=0.8}$		$\mathrm{CSTA}_{\Delta=0.4}$		MSTA	
		softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity
Cod-RNA	74.51 ± 8.86	74.75 ± 8.14	$\mathbf{80.06 \pm 3.55}$	73.39 ± 7.36	$\mathbf{78.39 \pm 4.66}$	75.28 ± 8.79	$\mathbf{76.88 \pm 7.67}$
COIL-20	84.54 ± 2.19	$\mathbf{84.69 \pm 3.56}$	84.57 ± 2.85	84.38 ± 3.05	$\mathbf{84.57 \pm 3.16}$	$\mathbf{84.32 \pm 2.34}$	84.07 ± 2.85
Digits	75.68 ± 4.59	$\mathbf{80.47 \pm 3.8}$	78.2 ± 3.34	78.4 ± 3.28	$\mathbf{79.14 \pm 3.5}$	78.02 ± 5.15	$\mathbf{79.8 \pm 5.92}$
DNA	78.82 ± 2.31	$\mathbf{80.29 \pm 2.24}$	79.06 ± 2.31	80.12 ± 2.08	$\mathbf{80.76 \pm 2.24}$	80.89 ± 2.64	$\mathbf{84.09 \pm 1.7}$
DryBean	64.6 ± 3.89	$\mathbf{65.6 \pm 4.18}$	61.55 ± 4.91	$\mathbf{64.91 \pm 3.72}$	64.6 ± 3.53	66.24 ± 4.31	$\mathbf{67.0 \pm 3.96}$
HAR	82.57 ± 1.96	82.87 ± 3.02	$\mathbf{83.12 \pm 2.27}$	82.19 ± 2.61	$\mathbf{83.53 \pm 3.77}$	$\mathbf{81.35 \pm 2.54}$	81.16 ± 1.63
Mnist	50.74 ± 2.25	51.08 ± 2.55	$\mathbf{52.69 \pm 2.42}$	51.7 ± 3.52	$\mathbf{54.26 \pm 1.82}$	51.6 ± 2.58	$\mathbf{54.18 \pm 2.34}$
Mushrooms	69.45 ± 7.29	59.53 ± 10.46	$\mathbf{71.36 \pm 6.63}$	62.98 ± 7.25	$\mathbf{77.55 \pm 7.65}$	72.16 ± 7.59	$\mathbf{76.16 \pm 13.04}$
Phishing	67.42 ± 3.55	66.08 ± 5.66	$\mathbf{77.41 \pm 3.93}$	66.88 ± 5.64	$\mathbf{76.17 \pm 8.58}$	69.48 ± 4.37	$\mathbf{75.83 \pm 7.52}$
Protein	57.57 ± 6.33	57.45 ± 6.36	$\mathbf{57.61 \pm 6.23}$	56.09 ± 5.61	$\mathbf{57.74 \pm 7.8}$	58.81 ± 6.54	$\mathbf{59.88 \pm 6.29}$
Rice	79.19 ± 5.12	80.54 ± 4.31	$\mathbf{81.1 \pm 4.28}$	79.88 ± 4.48	$\mathbf{81.56 \pm 3.61}$	80.35 ± 4.89	$\mathbf{82.63 \pm 5.63}$
Splice	66.13 ± 4.47	67.14 ± 2.62	$\mathbf{67.45 \pm 2.53}$	67.28 ± 2.07	$\mathbf{68.05 \pm 2.17}$	66.08 ± 4.98	$\mathbf{66.32 \pm 4.73}$
Svmguide1	70.89 ± 10.98	70.35 ± 11.74	$\mathbf{81.07 \pm 5.39}$	69.84 ± 11.06	$\mathbf{74.46 \pm 7.23}$	71.04 ± 11.11	$\mathbf{73.13 \pm 8.82}$

- $\mathcal{T}$-similarity is better overall;

Performance Results in SSL+SSB

Dataset	ERM	$\mathrm{PL}_{\theta=0.8}$		$\mathrm{CSTA}_{\Delta=0.4}$		MSTA	
		softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity
Cod-RNA	74.51 ± 8.86	74.75 ± 8.14	$\mathbf{80.06 \pm 3.55}$	73.39 ± 7.36	$\mathbf{78.39 \pm 4.66}$	75.28 ± 8.79	$\mathbf{76.88 \pm 7.67}$
COIL-20	84.54 ± 2.19	$\mathbf{84.69 \pm 3.56}$	84.57 ± 2.85	84.38 ± 3.05	$\mathbf{84.57 \pm 3.16}$	$\mathbf{84.32 \pm 2.34}$	84.07 ± 2.85
Digits	75.68 ± 4.59	$\mathbf{80.47 \pm 3.8}$	78.2 ± 3.34	78.4 ± 3.28	$\mathbf{79.14 \pm 3.5}$	78.02 ± 5.15	$\mathbf{79.8 \pm 5.92}$
DNA	78.82 ± 2.31	$\mathbf{80.29 \pm 2.24}$	79.06 ± 2.31	80.12 ± 2.08	$\mathbf{80.76 \pm 2.24}$	80.89 ± 2.64	$\mathbf{84.09 \pm 1.7}$
DryBean	64.6 ± 3.89	$\mathbf{65.6 \pm 4.18}$	61.55 ± 4.91	$\mathbf{64.91 \pm 3.72}$	64.6 ± 3.53	66.24 ± 4.31	$\mathbf{67.0 \pm 3.96}$
HAR	82.57 ± 1.96	82.87 ± 3.02	$\mathbf{83.12 \pm 2.27}$	82.19 ± 2.61	$\mathbf{83.53 \pm 3.77}$	$\mathbf{81.35 \pm 2.54}$	81.16 ± 1.63
Mnist	50.74 ± 2.25	51.08 ± 2.55	$\mathbf{52.69 \pm 2.42}$	51.7 ± 3.52	$\mathbf{54.26 \pm 1.82}$	51.6 ± 2.58	$\mathbf{54.18 \pm 2.34}$
Mushrooms	69.45 ± 7.29	59.53 ± 10.46	$\mathbf{71.36 \pm 6.63}$	62.98 ± 7.25	$\mathbf{77.55 \pm 7.65}$	72.16 ± 7.59	$\mathbf{76.16 \pm 13.04}$
Phishing	67.42 ± 3.55	66.08 ± 5.66	$\mathbf{77.41 \pm 3.93}$	66.88 ± 5.64	$\mathbf{76.17 \pm 8.58}$	69.48 ± 4.37	$\mathbf{75.83 \pm 7.52}$
Protein	57.57 ± 6.33	57.45 ± 6.36	$\mathbf{57.61 \pm 6.23}$	56.09 ± 5.61	$\mathbf{57.74 \pm 7.8}$	58.81 ± 6.54	$\mathbf{59.88 \pm 6.29}$
Rice	79.19 ± 5.12	80.54 ± 4.31	$\mathbf{81.1 \pm 4.28}$	79.88 ± 4.48	$\mathbf{81.56 \pm 3.61}$	80.35 ± 4.89	$\mathbf{82.63 \pm 5.63}$
Splice	66.13 ± 4.47	67.14 ± 2.62	$\mathbf{67.45 \pm 2.53}$	67.28 ± 2.07	$\mathbf{68.05 \pm 2.17}$	66.08 ± 4.98	$\mathbf{66.32 \pm 4.73}$
Svmguide1	70.89 ± 10.98	70.35 ± 11.74	$\mathbf{81.07 \pm 5.39}$	69.84 ± 11.06	$\mathbf{74.46 \pm 7.23}$	71.04 ± 11.11	$\mathbf{73.13 \pm 8.82}$

- $\mathcal{T}$-similarity is better overall;
- `Mushrooms` and `Phishing`: from degradation to improvement.

Performance Results in SSL+SSB

Dataset	ERM	$\text{PL}_{\theta=0.8}$		$\text{CSTA}_{\Delta=0.4}$		MSTA	
		softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity	softmax	$\mathcal{T}$-similarity
Cod-RNA	74.51 ± 8.86	74.75 ± 8.14	$\mathbf{80.06 \pm 3.55}$	73.39 ± 7.36	$\mathbf{78.39 \pm 4.66}$	75.28 ± 8.79	$\mathbf{76.88 \pm 7.67}$
COIL-20	84.54 ± 2.19	$\mathbf{84.69 \pm 3.56}$	84.57 ± 2.85	84.38 ± 3.05	$\mathbf{84.57 \pm 3.16}$	$\mathbf{84.32 \pm 2.34}$	84.07 ± 2.85
Digits	75.68 ± 4.59	$\mathbf{80.47 \pm 3.8}$	78.2 ± 3.34	78.4 ± 3.28	$\mathbf{79.14 \pm 3.5}$	78.02 ± 5.15	$\mathbf{79.8 \pm 5.92}$
DNA	78.82 ± 2.31	$\mathbf{80.29 \pm 2.24}$	79.06 ± 2.31	80.12 ± 2.08	$\mathbf{80.76 \pm 2.24}$	80.89 ± 2.64	$\mathbf{84.09 \pm 1.7}$
DryBean	64.6 ± 3.89	$\mathbf{65.6 \pm 4.18}$	61.55 ± 4.91	$\mathbf{64.91 \pm 3.72}$	64.6 ± 3.53	66.24 ± 4.31	$\mathbf{67.0 \pm 3.96}$
HAR	82.57 ± 1.96	82.87 ± 3.02	$\mathbf{83.12 \pm 2.27}$	82.19 ± 2.61	$\mathbf{83.53 \pm 3.77}$	$\mathbf{81.35 \pm 2.54}$	81.16 ± 1.63
Mnist	50.74 ± 2.25	51.08 ± 2.55	$\mathbf{52.69 \pm 2.42}$	51.7 ± 3.52	$\mathbf{54.26 \pm 1.82}$	51.6 ± 2.58	$\mathbf{54.18 \pm 2.34}$
Mushrooms	69.45 ± 7.29	59.53 ± 10.46	$\mathbf{71.36 \pm 6.63}$	62.98 ± 7.25	$\mathbf{77.55 \pm 7.65}$	72.16 ± 7.59	$\mathbf{76.16 \pm 13.04}$
Phishing	67.42 ± 3.55	66.08 ± 5.66	$\mathbf{77.41 \pm 3.93}$	66.88 ± 5.64	$\mathbf{76.17 \pm 8.58}$	69.48 ± 4.37	$\mathbf{75.83 \pm 7.52}$
Protein	57.57 ± 6.33	57.45 ± 6.36	$\mathbf{57.61 \pm 6.23}$	56.09 ± 5.61	$\mathbf{57.74 \pm 7.8}$	58.81 ± 6.54	$\mathbf{59.88 \pm 6.29}$
Rice	79.19 ± 5.12	80.54 ± 4.31	$\mathbf{81.1 \pm 4.28}$	79.88 ± 4.48	$\mathbf{81.56 \pm 3.61}$	80.35 ± 4.89	$\mathbf{82.63 \pm 5.63}$
Splice	66.13 ± 4.47	67.14 ± 2.62	$\mathbf{67.45 \pm 2.53}$	67.28 ± 2.07	$\mathbf{68.05 \pm 2.17}$	66.08 ± 4.98	$\mathbf{66.32 \pm 4.73}$
Svmguide1	70.89 ± 10.98	70.35 ± 11.74	$\mathbf{81.07 \pm 5.39}$	69.84 ± 11.06	$\mathbf{74.46 \pm 7.23}$	71.04 ± 11.11	$\mathbf{73.13 \pm 8.82}$

- $\mathcal{T}$-similarity is better overall;
- Mushrooms and Phishing: from degradation to improvement.
- Results on SSL i.i.d.: no significant improvement nor degradation.

Thanks for your attention !