Leveraging Ensemble Diversity for Robust Self-Training under Sample Selection Bias

Ambroise Odonnat
Huawei Noah's Ark Lab, Inria
Université Rennes 2, CNRS, IRISA

DIG Telecom Seminar

September 24, 2024

Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

Introduction

In some applications, data acquisition is cheaper than labeling . . .

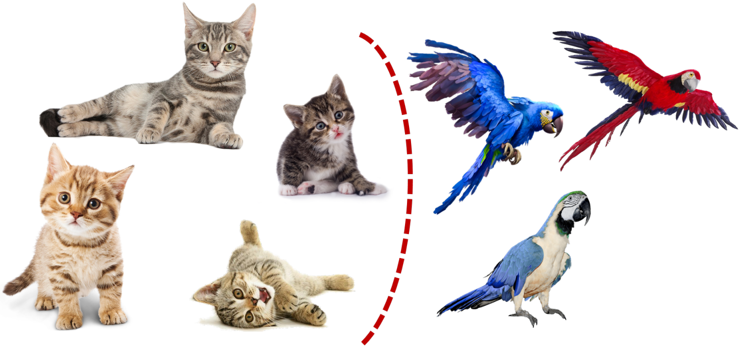
. . . and supervised learning is inefficient.

Semi-Supervised Learning (SSL)

SSL $\rightarrow$ learn from a few labeled and many unlabeled examples.

Semi-Supervised Learning (SSL)

SSL $\rightarrow$ learn from a few labeled and many unlabeled examples.



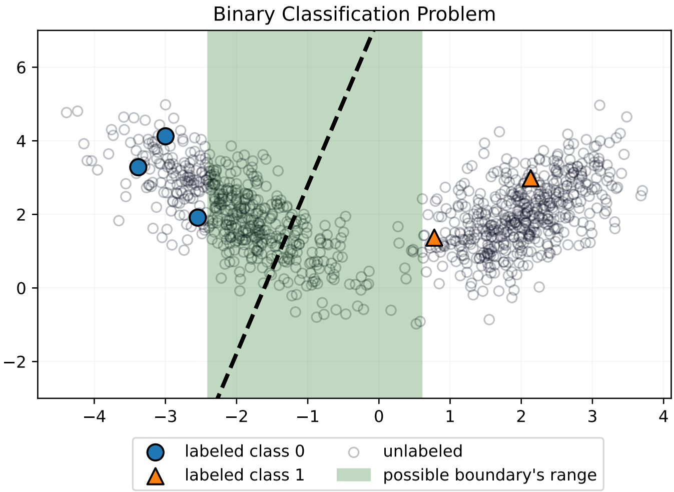
Families of SSL Methods

- Pseudo-labeling (Amini et al., 2023):
  - Unlabeled regularization (Feofanov et al., 2023)
  - Self-training (Feofanov et al., 2019)
- Graph-based algorithms (van Engelen and Hoos, 2020):
  - Label propagation
  - Label spreading
- Unsupervised preprocessing (van Engelen and Hoos, 2020):
  - Cluster-then-label
  - Feature extraction: auto-encoders, PCA
  - Pre-training: self-supervised learning, stacked auto-encoders

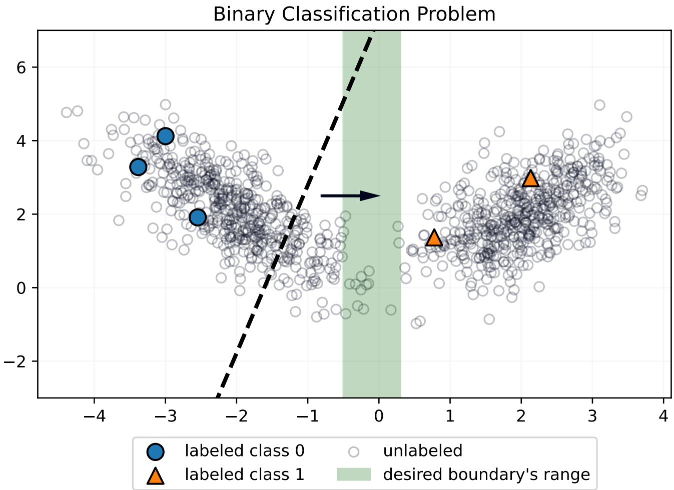
Families of SSL Methods

- **Pseudo-labeling** (Amini et al., 2023):
  - Unlabeled regularization (Feofanov et al., 2023)
  - Self-training(Feofanov et al., 2019)
- Graph-based algorithms (van Engelen and Hoos, 2020):
  - Label propagation
  - Label spreading
- Unsupervised preprocessing (van Engelen and Hoos, 2020):
  - Cluster-then-label
  - Feature extraction: auto-encoders, PCA
  - Pre-training: self-supervised learning, stacked auto-encoders

Range of possible supervised classifiers is vast: we need to make assumptions.



Low Density Separation (LDS) assumption: push boundary away from regions of unlabeled data.

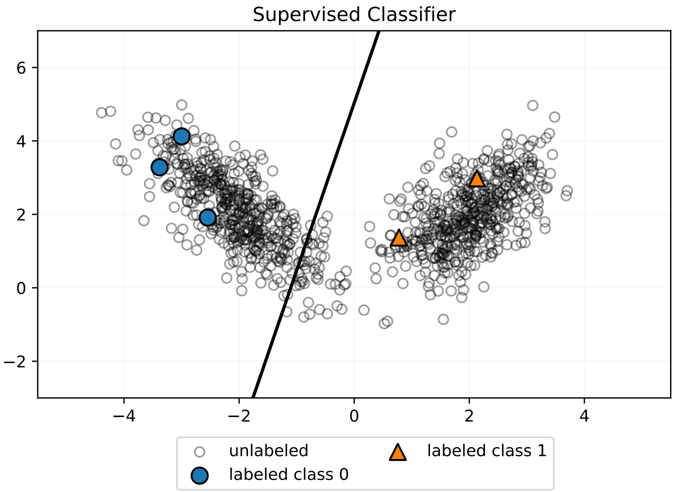


Family of SSL Methods

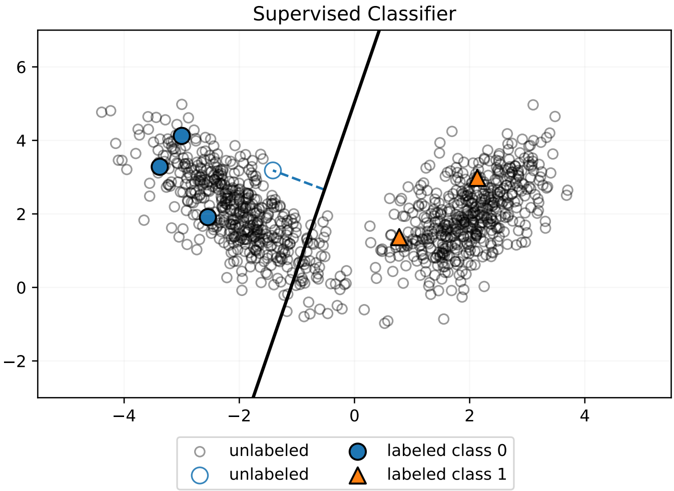
- **Pseudo-labeling** (Amini et al., 2023):
  - Unlabeled regularization (Feofanov et al., 2023)
  - **Self-training** (Feofanov et al., 2019)
- Graph-based algorithms (van Engelen and Hoos, 2020):
  - Label propagation
  - Label spreading
- Unsupervised preprocessing (van Engelen and Hoos, 2020):
  - Cluster-then-label
  - Feature extraction: auto-encoders, PCA
  - Pre-training: self-supervised learning, stacked auto-encoders

Self-Training

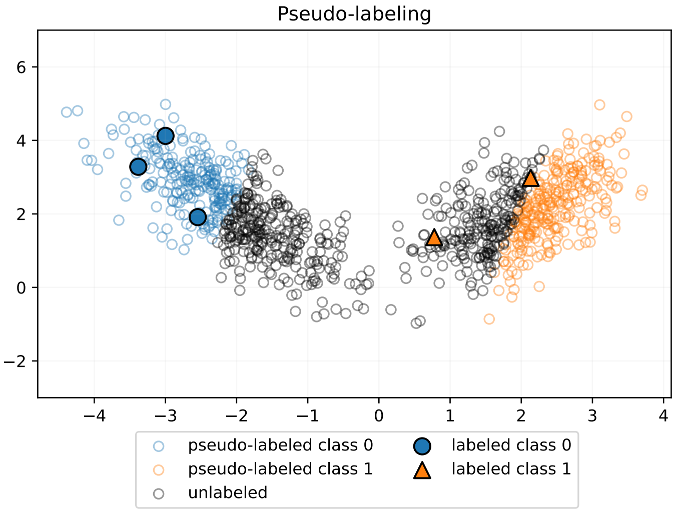
Start from a supervised classifier trained on the labeled set.



Predict labels and confidence scores for unlabeled data.



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



Self-Training

Retrain the model and repeat the same procedure again.

Self-Training

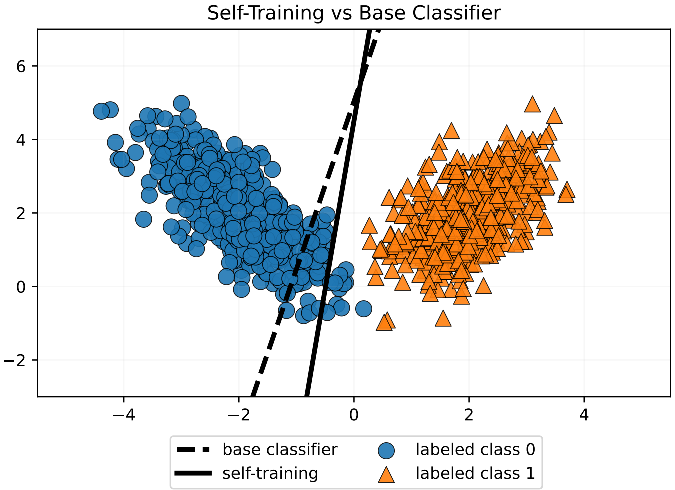
And again...

Self-Training

Until there are no data to pseudo-label.

Self-Training

Self-training pushed the boundary away from the confident data.



Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

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 select unlabeled data for pseudo-labeling at each iteration?

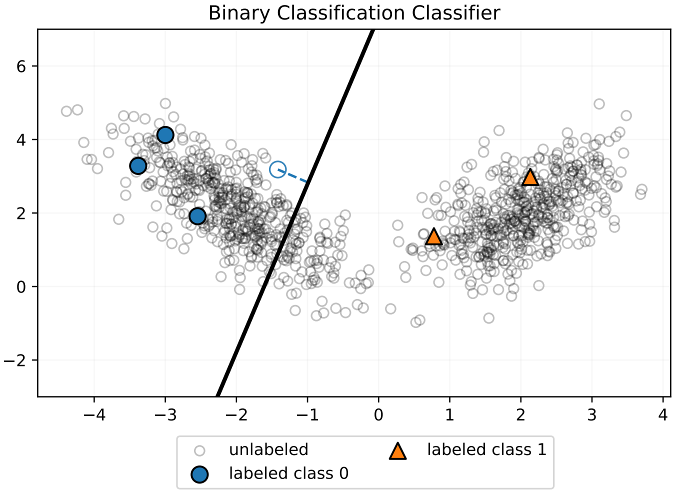
Two Fundamental Questions

1. *Confidence Estimation* $\rightarrow$ How to rank unlabeled data?
2. *Pseudo-Labeling Policy* $\rightarrow$ How to select unlabeled data for pseudo-labeling at each iteration?

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

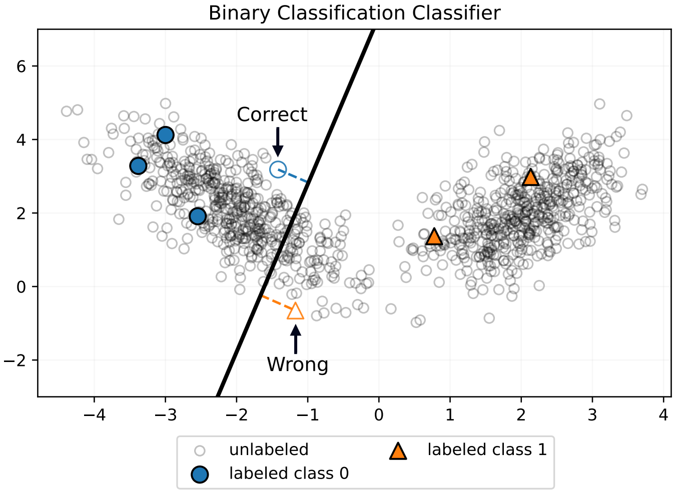
Failure Cases

Requirements → trust the classifier's predictions.



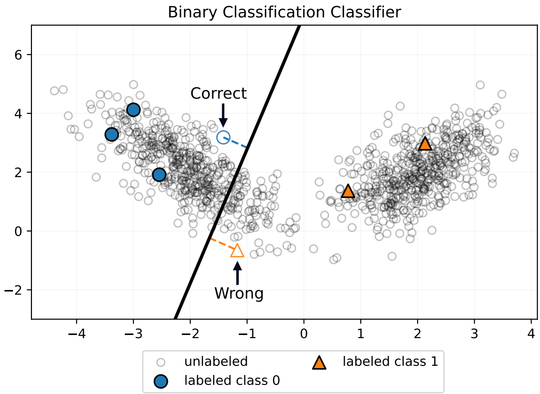
Failure Cases

Problem $\rightarrow$ not safe since the prediction can be wrong.



Failure Cases

Biased prediction confidence $\Rightarrow$ wrong direction can be chosen.
$\rightarrow$ This can occur when there is a distribution shift in the data.



Sample Selection Bias

- SSL assumption: labeled and unlabeled data are i.i.d.
- Confidence can be biased when this assumption does not hold

Sample Selection Bias

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

Sample Selection Bias

- SSL assumption: labeled and unlabeled data are i.i.d.
- Confidence can be biased when this assumption does not hold
- **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.

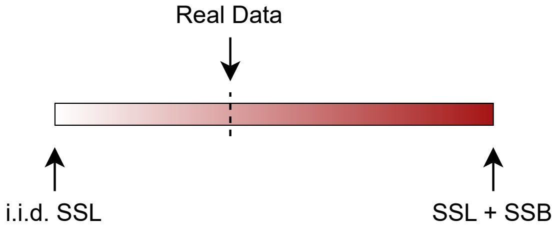
Sample Selection Bias

- SSL assumption: labeled and unlabeled data are i.i.d.
- Confidence can be biased when this assumption does not hold
- **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.
- SSB has been studied but not in the case of SSL.

SSL under Sample Selection Bias

SSL + SSB combines SSL and Sample Selection Bias (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

Select the labeled set to violate the i.i.d. assumption.

- Binary selection variable s_i for each $\mathbf{x}_i$;
- $s_i = 1$ if $\mathbf{x}_i$ is labeled, $s_i = 0$ otherwise;
- Model $\mathbb{P}(s_i = 1 | \mathbf{x}_i, y_i)$ to violate i.i.d. assumption.

Implementation of SSL + SSB

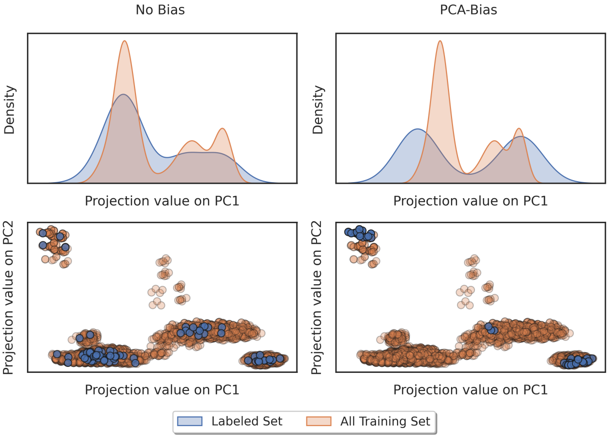
Select the labeled set to violate the i.i.d. assumption.

`PCA-Bias` $\rightarrow$ for each class c,

1. Apply `PCA` on training data of class c;
2. Compute $\mathrm{proj}_1(\mathbf{x}_i)$, projection value on PC1;
3. $\mathbb{P}(s_i = 1 | \mathbf{x}_i, y_i = c) \propto \exp(r \cdot |\mathrm{proj}_1(\mathbf{x}_i)|), \quad r > 0.$

Implementation of SSL + SSB

Select the labeled set to violate the i.i.d. assumption.



Pseudo-Labeling Policies

- ERM corresponds to supervised learning on the labeled set
- $\texttt{PL}_{\theta=0.8}$ uses a fixed threshold $\theta = 0.8$ (Lee, 2013)
- $\texttt{CSTA}_{\Delta=0.4}$ takes $\Delta\%$ most confident (Cascante-Bonilla et al., 2020)
- MSTA optimizes the threshold to balance the error and the amount of data pseudo-labeled (Feofanov et al., 2019)

Failure of Self-Training under SSL + SSB

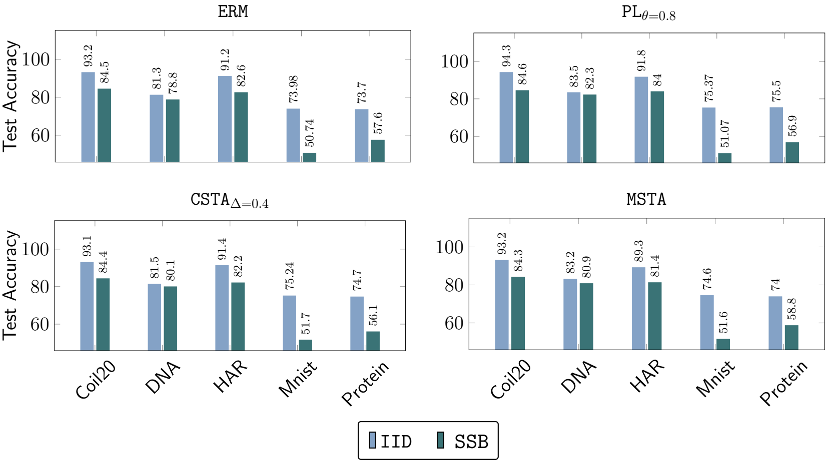


Figure: Test accuracies of the different baselines on 5 datasets. Full results to be found here.

Unreliable Model Selection

LOO over-optimistic w.r.t. generalization performance (Figure 1).

- Leave one labeled point out;
- Train on the remaining $n_\ell - 1$;
- Test on the one left out;
- Repeat for each labeled point.

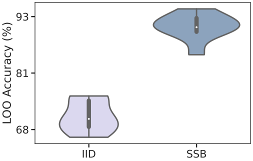


Figure: LOO on Mnist.

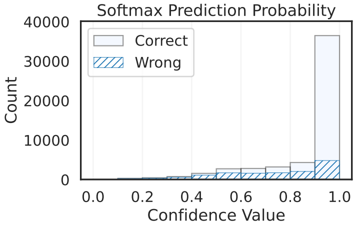
Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

Motivation

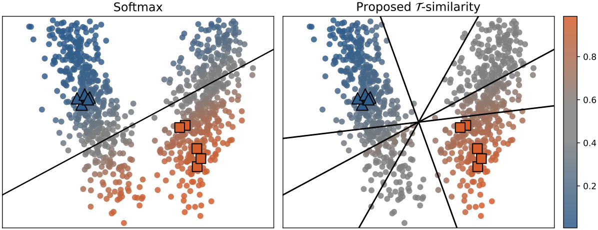
softmax-based confidence measure is unreliable in SSL + SSB.

- NNs are overconfident;
- softmax predictions biased towards the labeled set.



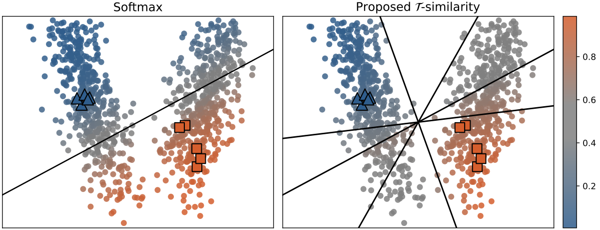
$\rightarrow$ We propose a novel confidence measure for NNs.

Leveraging Ensemble Diversity



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

Leveraging Ensemble Diversity

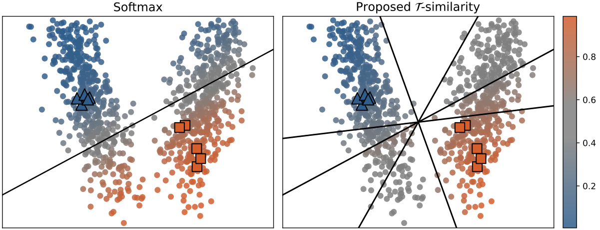


$$\min_{\mathcal{T}} \frac{1}{M} \sum_{h \in \mathcal{T}} \underbrace{\left(\frac{1}{n_\ell} \sum_{(\mathbf{x},y) \in \mathbf{X}_\ell \times \mathbf{y}_\ell} \ell(h(\mathbf{x}), y) \right)}_{\text{supervised loss}} + \underbrace{\frac{\gamma}{n_u M(M-1)} \sum_{h \neq \tilde{h} \in \mathcal{T}} \sum_{\mathbf{x}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

We jointly train the ensemble to

1. Fit very well the labeled data
2. Disagree as much as possible on unlabeled data

Leveraging Ensemble Diversity



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

We jointly train the ensemble to

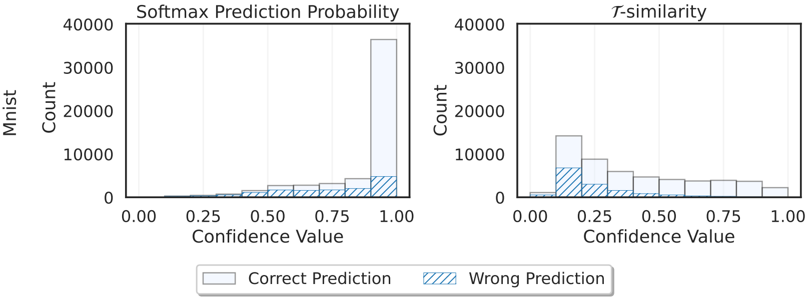
1. Fit very well the labeled data
2. Disagree as much as possible on unlabeled data

$\mathcal{T}$-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 $0 \leq s_{\mathcal{T}}(\mathbf{x}) \leq 1$.



Practical Implementation

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

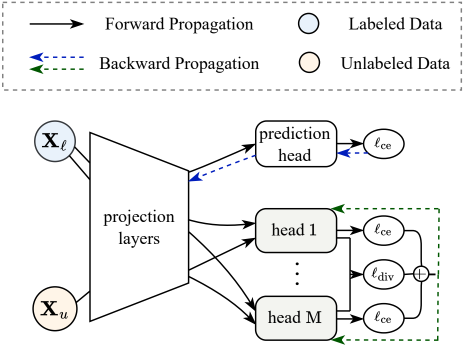


Figure: Architecture of the model.

Theoretical Analysis

- Fixed representation of dimension d, binary linear classification
- Linear ensemble $\mathbf{W} = \{\boldsymbol{w}_m \in \mathbb{R}^d | 1 \leq m \leq M\}$
- Prediction of $\boldsymbol{\omega}_m$ on $\mathbf{x}$ is $\operatorname{sign}(\boldsymbol{\omega}_m^\top \mathbf{x})$

$$\mathcal{L}(\mathbf{W}) := \underbrace{\frac{1}{M n_\ell} \sum_{m=1}^{M} \sum_{i=1}^{n_\ell} \left(y_i - \boldsymbol{\omega}_m^\top \mathbf{x}_i \right)^2}_{\text{label fidelity term}} + \underbrace{\frac{1}{M} \sum_{m=1}^{M} \lambda_m \|\boldsymbol{\omega}_m\|^2}_{\text{regularization}} + \underbrace{\frac{\gamma}{n_u M (M-1)} \sum_{m \neq k} \sum_{i=n_\ell+1}^{n_\ell+n_u} w_m^\top \mathbf{x}_i w_k^\top \mathbf{x}_i}_{\text{agreement term}}, \qquad \textbf{(P)}$$

where γ controls the influence of the diversity on the learning.

Theoretical Analysis

- Fixed representation of dimension d, binary linear classification
- Linear ensemble $\mathbf{W} = \{\boldsymbol{w}_m \in \mathbb{R}^d | 1 \leq m \leq M\}$
- Prediction of $\boldsymbol{\omega}_m$ on $\mathbf{x}$ is $\mathrm{sign}(\boldsymbol{\omega}_m^\top \mathbf{x})$

Theorem (O., Feofanov, Redko)

1. *Convergence to a stationary point under mild assumption*
2. *Lower-bound on the diversity of stationary points*
3. *Connection to contrastive learning*

Convergence to a Stationary Point

- Labeled set $(\mathbf{X}_\ell, \mathbf{y}_\ell) = (\mathbf{x}_i, y_i)_{i=1}^{n_\ell}$
- Unlabeled set $\mathbf{X}_u = (\mathbf{x}_i)_{i=n_\ell+1}^{n_\ell+n_u}$
- Assumption **A**: $\forall m \in [\![1, M]\!], \lambda_m > \frac{\gamma(M+1)}{n_u(M-1)} \lambda_{\max}(\mathbf{X}_u^\top \mathbf{X}_u)$.

Theorem (O., Feofanov, Redko)

Under Assumption **A**, *$\mathcal{L}$ is strictly convex and coercive on $\mathbb{R}^{d \times M}$. Hence, the optimization problem* (**P**) *admits a unique solution $\mathbf{W}^*$ that verifies*

$$\nabla \mathcal{L}(\mathbf{W}^*) = 0. \tag{1}$$

Ensemble Diversity

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

Theorem (O., Feofanov, Redko)

$$\gamma \ell_{\mathit{div}}(\mathbf{W}^*, \mathbf{X}_u) \geq \frac{1}{2n_\ell M} \sum_{m=1}^{M} \|\mathbf{y}_\ell - \mathbf{X}_\ell \boldsymbol{\omega}_m^*\|_2^2 + \frac{1}{2M} \sum_{m=1}^{M} (\boldsymbol{\omega}_m^*)^\top \left(\lambda_m \mathbf{I}_d + \frac{\mathbf{X}_\ell^\top \mathbf{X}_\ell}{n_\ell}\right) \boldsymbol{\omega}_m^*.$$

1. Trade-off between supervised performance and margin term
2. Assuming orthogonality, the predictors $\boldsymbol{\omega}_m$ span the M directions of largest variance of the labeled data.

Ensemble Diversity

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

Theorem (O., Feofanov, Redko)

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

1. Direction of smallest variance is also important for diversity
2. Theorem shows the importance of representation learning

Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

Diversity provides Calibrated Confidence Measure

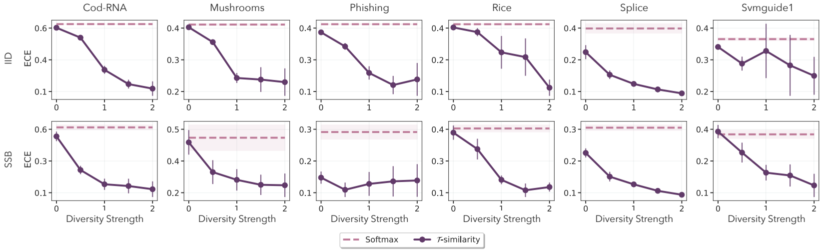


Figure: Increasing the diversity improves the classifier's calibration

Pseudo-Labeling Policies

- $\texttt{ERM}$ corresponds to supervised learning on the labeled set
- $\texttt{PL}_{\theta=0.8}$ uses a fixed threshold $\theta = 0.8$ (Lee, 2013)
- $\texttt{CSTA}_{\Delta=0.4}$ takes $\Delta\%$ most confident (Cascante-Bonilla et al., 2020)
- $\texttt{MSTA}$ optimizes the threshold to balance the error and the amount of data pseudo-labeled (Feofanov et al., 2019)

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;

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;
- Even go from degradation to improvement on 2 datasets.

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;
- Even go from degradation to improvement on 2 datasets;
- Our approach remains similar to `softmax` in i.i.d. SSL.

Outline

1. Introduction
2. Failure of Self-Training
3. Learning with the $\mathcal{T}$-similarity
4. Numerical Experiments
5. Discussion

Discussion

1. Practical and principled framework to study SSL + SSB;
2. Calibrated confidence measure;
3. $\mathcal{T}$-similarity good both in i.i.d. SSL and SSL + SSB.

Future work $\rightarrow$ use $\mathcal{T}$-similarity for iterative self-training, domain adaptation, or uncertainty modeling.

To Know More

This work has been accepted to AISTATS 2024, Valencia, Spain. You may find the links to the paper and the code below. to know more about my research, see my website: ambroiseodt.github.io and feel free to contact me.

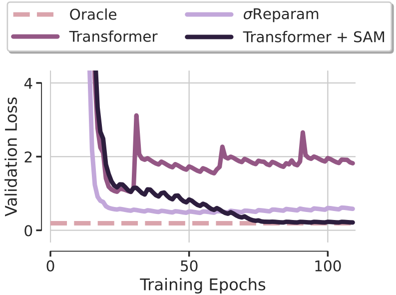
- Paper: https://arxiv.org/abs/2310.14814
- Code: https://github.com/ambroiseodt/tsim

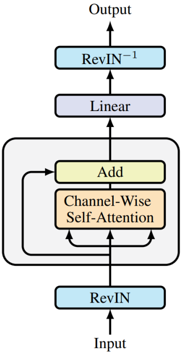
Acknowledgement

I thank my co-authors Vasilii Feofanov and Ievgen Redko who supervised me during the internship and without whom this project would not have existed. I would also like to thank Gabriel Peyré for his insightful comments on early drafts of the paper, as well as Malik Tiomoko, and Aladin Virmaux for the fruitful discussions that led to this work.

Self-Promotion

SAMformer: Unlocking the Potential of Transformers in Time Series Forecasting - Oral ICML 2024





- Paper: https://arxiv.org/pdf/2402.10198
- Code: https://github.com/romilbert/samformer

Thanks for your attention !

References I

Amini, M.-R., Feofanov, V., Pauletto, L., Devijver, E., and Maximov, Y. (2023). Self-Training: A Survey.

Cascante-Bonilla, P., Tan, F., Qi, Y., and Ordonez, V. (2020). Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning.

Feofanov, V., Devijver, E., and Amini, M.-R. (2019). Transductive Bounds for the Multi-Class Majority Vote Classifier. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

References II

Feofanov, V., Tiomoko, M., and Virmaux, A. (2023). Random Matrix Analysis to Balance between Supervised and Unsupervised Learning under the Low Density Separation Assumption.

Lee, D.-H. (2013). Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*.

van Engelen, J. E. and Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440.

Datasets

Dataset	Size	# of lab. examples n_ℓ	Dimension d	# classes C
Cod-RNA	59535	99	8	2
COIL-20	1440	200	1024	20
Digits	1797	99	64	10
DNA	3186	149	180	6
DryBean	13543	104	16	7
HAR	10299	299	561	3
Mnist	70000	100	784	10
Mushrooms	8124	79	112	2
Phishing	11055	99	68	2
Protein	1080	80	77	8
Rice	3810	29	7	2
Splice	3175	39	20	2
Svmguide1	3089	39	4	2

Table: Characteristics of datasets used in our experiments