Trustworthy Machine Learning on Imbalance Data

Jiangchao Yao
CMIC, Shanghai Jiao Tong University
Large-scale natural sources are very imbalance, usually following a long-tailed distribution.

A direct decomposition on the risk minimization

$$\min_f R(f) = E_{P(x,y)}[\ell(f(x), y)] = \sum_{k=1}^{K} P(y = k) E_{P(x|y=k)}[\ell(f(x), y)]$$

Minority ("generalized" conceptual) classes have weak importance for training, which will be easily ignored in the early phase especially for overparameterized DNNs [1] (or namely, will be sacrificed first if it is not sufficient for the model to learn).

However, in real applications, the value of classes cannot be absolutely characterized by their quantity, and instead, sometimes less is more for sustainable long-term development.

- **Fairness** w.r.t. diversity e.g., small populations of gender, race and consumers
- **Cost-sensitive scenarios** e.g., medical disease diagnosis and treatment (COVID-19)

Why the Resulted Imbalanced Learning is Special

A critical highlight on the evaluation, different from the ordinary IID learning

The change in evaluation metric induces an statistical consistency problem on applying conventional learning methods, that is,

*What we design during training should be statistically consistent with what we pursue about the evaluation.*
The development of imbalance learning

1998  

2009  
He et al. Learning from Imbalanced Data. TKDE 2009.

2018  

2023  
Zhang et al. Deep Long-tailed Learning: A Survey. TPAMI 2023
SMOTE: Synthetic Minority Over-sampling Technique

Motivation: Replication of the minority class does not cause its decision boundary to spread into the majority class region (but overfitting).

The main idea of SMOTE: augmentation for minority class by interpolation instead of over-sampling with replacement.

Interpolation is limited by the samples. Thus, SMOTE also always runs with the under-sampling for majority class.

Increase minority diversity and decrease majority diversity

Threshold-Moving: adjust the prediction in a post-hoc manner.

Motivation: The over-confident prediction for majority or the low-confident prediction for minority can be calibrated after training.

The threshold-moving algorithm

**Training phase:**
1. Let $S$ be the original training set.
2. Train a neural network from $S$.

**Test phase:**
1. Generate real-value outputs with the trained neural network.
2. For every output, multiply it with the sum of the costs of misclassifying the corresponding class to other classes.
3. Return the class with the biggest output.

Moving function

\[
\hat{p}_k = \frac{p_k \times \sum_{k'=1}^{K} C[k][k']}{\eta}
\]

where $p_k$ is the probabilistic prediction, $C[k][k']$ is the cost mis-predicted from class $k$ to $k'$, and $\eta$ is renormalization parameter.


Sampling methods might not always show promise in multi-class imbalance learning, but threshold-moving way does.

Let $p_k = \frac{e^{\pi_k} \times \sum_{k'=1}^{K} C[k][k']} {\sum_{k'=1}^{K} e^{\pi_k}}$ denote the class prediction. If we set $\sum_{k'=1}^{K} C[k][k'] = e^{-\tau \log \pi_k}$ where $\pi_k$ is the class prior and $\tau$ is the temperature, the threshold-moving method recovers the popular logit adjustment method for long-tailed learning.

Majority classes have the smaller cost than minority classes, e.g., $e^{-\tau \log \pi_k}$ is monotonously decreasing.
COG: Local Decomposition for Rare Class Analysis

Intuition: Quantity imbalance limits the learning pace of minority over majority. We can adjust the quantities by decomposition.

How to properly decompose the majority classes (or including minority classes) into subclasses to balance the training?

Phase I: local clustering
1. for class $i = 1$ to $c$  // “c” represents #classes
2. clusterLabel($i$) = Clustering($D(i)$, $K(i)$);
3. $D(i)* = changeLabel(D(i), clusterLabel(i))$;
4. end for

Phase II: over-sampling (for COG-OS only)
5. for class $j = 1$ to $c$
6. $D(j)** = replicate(D(j)*, r(j))$
7. end for
8. $D** = \bigcup_{j=1}^{c} (D(j)**)$;

Phase III: training
9. $M = train(D**, L)$;

Phase IV: predicting
10. $p' = predict(T, M)$;
11. $p = convertLabel(p')$;

On Statistical Consistency of Binary Classification with Balanced Accuracy

Motivation: The early ERM theory is developed for the instance-wise evaluation, but cannot guarantee the consistency for balanced measure.

Accuracy = $\mathbb{E}_{p(x,y)}[h(x) = y]$

Balanced Accuracy $= \frac{\sum_{k \in \{-1, 1\}} \mathbb{E}_p(x|y=k)[h(x)=k]}{2}$

If we consider the balanced accuracy, how to modify the algorithm to satisfy the statistical consistency?

**Theorem 3.** Let $D$ be a probability distribution on $\mathcal{X} \times \{\pm 1\}$ satisfying Assumption A. Let $\hat{p}_S$ denote any estimator of $p = \mathbb{P}(y = 1)$ satisfying $\hat{p}_S \in (0, 1)$ and $\hat{p}_S \xrightarrow{P} p$. Let $\tilde{\eta}_S: \mathcal{X} \rightarrow [0, 1]$ denote any class probability estimator satisfying $\mathbb{E}_r[|\tilde{\eta}_S(x) - \eta(x)|^r] \xrightarrow{P} 0$ for some $r \geq 1$, and let $h_S(x) = \text{sign}(\tilde{\eta}_S(x) - \hat{p}_S)$. Then $\text{regret}^{AM}_D[h_S] \xrightarrow{P} 0$.

**Algorithm 1** Plug-in with Empirical Threshold

1. **Input:** $S = ((x_1, y_1), \ldots, (x_n, y_n)) \in (\mathcal{X} \times \{\pm 1\})^n$
2. **Select:** (a) Proper (composite) loss $\ell: \{\pm 1\} \times \mathbb{R} \rightarrow \mathbb{R}_+^+$, with link function $\psi: [0, 1] \rightarrow \mathbb{R}$; (b) RKHS $\mathcal{F}_K$ with positive definite kernel $K: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$; (c) regularization parameter $\lambda_n > 0$
3. $f_S \in \arg\min_{f \in \mathcal{F}_K} \left\{ \frac{1}{n} \sum_{i=1}^n \ell(y_i, f(x_i)) + \lambda_n \|f\|_K^2 \right\}$
4. $\tilde{\eta}_S = \psi^{-1} \circ f_S$
5. $\hat{p}_S = (\text{as in Eq. (2)})$
6. **Output:** Classifier $h_S(x) = \text{sign}(\tilde{\eta}_S(x) - \hat{p}_S)$

Summary of imbalanced learning in the early years

Imbalanced Learning

- Re-Sampling
  - Control sample quantity from the input augmentation perspective
  - Adjust penalty preference from the loss perspective.

- Loss Adjustment
  - Control learning complexity from the label manipulation perspective

- Re-Labeling
  - The statistically-consistent loss family for imbalanced binary classification
What is the new of this topic in the recent years?
The recent advances of imbalance learning powered by deep learning

- imbalance
  - binary Imbalance
  - multi-label Imbalance
- long tail
  - supervised long-tailed learning
  - weakly-supervised long-tailed learning
  - self-supervised long-tailed learning
Supervised Long-tailed Learning

It has been contributed with very broad explorations

loss re-weighting by effective number

- **Intuition:** Non-overlapping sample number, instead of the vanilla quantity number, playing the role of imbalance

- **Effective Number:** The effective number of examples is the expected volume of samples.
  \[
  E_n = \frac{(1-\beta^n)}{(1-\beta)}
  \]
  where \( \beta = \frac{(N - 1)}{N} \)
  \[
  \lim_{\beta \to 1} E_n = n
  \]

- **Class-Balanced Loss:** Training from imbalanced data by introducing a weighting factor that is inversely proportional to the effective number of samples.

  The class-balanced loss term can be applied to a wide range of deep networks and loss functions.

Supervised Long-tailed Learning

- **Class-Balanced Loss**: The class-balanced (CB) loss can be written as:

\[
CB(p, y) = \frac{1}{E_n \cdot y} \mathcal{L}(p, y) = \frac{1 - \beta}{1 - \beta \cdot n_y} \mathcal{L}(p, y)
\]

\[
CB_{\text{softmax}}(z, y) = -\frac{1 - \beta}{1 - \beta \cdot n_y} \log \left( \frac{\exp(z_y)}{\sum_{j=1}^{C} \exp(z_j)} \right)
\]

Class-Balanced loss can also combined with sigmoid cross-entropy loss, focal loss, etc.

---

Class-wise margin calibration

➢ **Motivation:** Re-weighting and re-sampling often cause over-fitting. The authors propose to regularize the minority classes more strongly than the frequent classes to improve the generalization error of minority classes without sacrificing the model’s ability to fit the frequent classes.

➢ **Class-distribution-aware margin trade-off:** Enforcing bigger margins can improve generalization, but may also hurt the margins of the frequent classes.

The generalization error is proportional to the following:

\[
\frac{1}{\gamma_1 \sqrt{n_1}} + \frac{1}{\gamma_2 \sqrt{n_2}} \quad \gamma_1 + \gamma_2 = \gamma
\]

For multi-class classification tasks, the margin of a sample is defined as:

\[
\gamma(x, y) = f(x)_y - \max_{j \neq y} f(x)_j
\]

The margin of a class is defined as the minimum value of the spacing between all its samples:

\[
\gamma_j = \min_{i \in S_j} \gamma(x_i, y_i) \quad \gamma_j = \frac{C}{n_j^{1/4}}
\]

**LDAM**: The authors define their loss function as:

$$\mathcal{L}_{\text{LDAM-HG}}((x, y); f) = \max\left(\max_{j \neq y}\{z_j\} - z_y + \Delta_y, 0\right)$$

where \(\Delta_j = \frac{C}{n_j^{1/4}}\) for \(j \in \{1, \ldots, k\}\)

The smooth relaxation of the hinge loss is the following cross-entropy loss with enforced margins:

$$\mathcal{L}_{\text{LDAM}}((x, y); f) = -\log \frac{e^{z_y - \Delta_y}}{e^{z_y - \Delta_y} + \sum_{j \neq y} e^{z_j}}$$

where \(\Delta_j = \frac{C}{n_j^{1/4}}\) for \(j \in \{1, \ldots, k\}\)

---

The table below shows the comparison of different methods for both imbalanced CIFAR-10 and CIFAR-100 datasets:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Imbalanced CIFAR-10</th>
<th>Imbalanced CIFAR-100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imbalance Type</td>
<td>long-tailed</td>
<td>step</td>
</tr>
<tr>
<td>Imbalance Ratio</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERM</td>
<td>29.64</td>
<td>13.61</td>
</tr>
<tr>
<td>Focal [Lin et al., 2017]</td>
<td>29.62</td>
<td>13.34</td>
</tr>
<tr>
<td>LDAM</td>
<td>26.65</td>
<td>13.04</td>
</tr>
<tr>
<td>CB RW [Cui et al., 2019]</td>
<td>29.45</td>
<td>13.21</td>
</tr>
<tr>
<td>CB Focal [Cui et al., 2019]</td>
<td>25.43</td>
<td>12.90</td>
</tr>
<tr>
<td>HG-DRS</td>
<td>27.16</td>
<td>14.03</td>
</tr>
<tr>
<td>LDAM-HG-DRS</td>
<td>24.42</td>
<td>12.72</td>
</tr>
<tr>
<td>M-DRW</td>
<td>24.94</td>
<td>13.57</td>
</tr>
<tr>
<td>LDAM-DRW</td>
<td><strong>22.97</strong></td>
<td><strong>11.84</strong></td>
</tr>
</tbody>
</table>

---

Class-wise logit adjustment

➢ **Motivation:** Design a consistent loss function that allows for a relatively elastic margin in the logit for head and tail.

➢ **Balanced error:** Under class imbalance, to measure balanced error:

\[
BER(f) = \frac{1}{L} \sum_{y \in [L]} \mathbb{P}_{x \mid y} \left( y \notin \arg\max_{y' \in [y]} f_{y'}(x) \right)
\]

Under Bayes-optimal prediction, if

\[
\mathbb{P}^{\text{bal}}(y \mid x) \propto \mathbb{P}(y \mid x) / \mathbb{P}(y)
\]

Then

\[
\arg\max_{y \in [L]} \mathbb{P}^{\text{bal}}(y \mid x) = \arg\max_{y \in [L]} \exp(s^*_y(x)) / \mathbb{P}(y) = \arg\max_{y \in [L]} s^*_y(x) - \ln \mathbb{P}(y)
\]

The logit adjusted softmax cross-entropy

\[
\ell(y, f(x)) = - \log \frac{e^{f_y(x) + \tau \cdot \log \pi_y}}{\sum_{y' \in [L]} e^{f_{y'}(x) + \tau \cdot \log \pi_{y'}}} = \log \left[ 1 + \sum_{y' \neq y} \left( \frac{\pi_{y'}}{\pi_y} \right)^\tau \cdot e^{(f_{y'}(x) - f_y(x))} \right]
\]

Post-hoc logit adjustment

\[
w_1^\top \Phi(x)/\pi_1 < w_2^\top \Phi(x)/\pi_2 \Leftrightarrow \exp(w_1^\top \Phi(x))/\pi_1 < \exp(w_2^\top \Phi(x))/\pi_2.
\]

Post-hoc logit adjustment

\[
\arg\max_{y \in [L]} \exp(w_y^\top \Phi(x))/\pi_y^\top = \arg\max_{y \in [L]} f_y(x) - \tau \cdot \log \pi_y
\]

An remarkable point on the statistical consistency of long-tailed multi-class classification

\[
\ell(y, f(x)) = \alpha_y \cdot \log \left[ 1 + \sum_{y' \neq y} e^{\Delta_{yy'}} \cdot e^{(f_{y'}(x) - f_y(x))} \right]
\]

**Theorem 1.** For any \( \delta \in \mathbb{R}^L_+ \), the pairwise loss in (11) is Fisher consistent with weights and margins

\[
\alpha_y = \delta_y / \mathbb{P}(y) \quad \Delta_{yy'} = \log \left( \delta_{y'} / \delta_y \right).
\]

Letting \( \delta_y = \pi_y \), we immediately deduce that the logit-adjusted loss of (10) is consistent, provided our \( \pi_y \) is a consistent estimate of \( \mathbb{P}(y) \). Similarly, \( \delta_y = 1 \) recovers the classic result that the balanced loss is consistent. While Theorem 1 only provides a sufficient condition in multi-class setting, one can provide a necessary and sufficient condition that rules out other choices of \( \Delta \) in the binary case.

Proposition 3 (Data-Dependent Bound for the VS Loss). Given the function set $\mathcal{F}$ and the VS loss $L_{VS}$, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the training set $\mathcal{S}$, the following generalization bound holds for all $f \in \mathcal{F}$:

$$
\mathcal{R}_{bag}(f) \preceq \Phi(L_{VS}, \delta) + \frac{\hat{C}_S(\mathcal{F})}{C \pi_C} \sum_{y=1}^{C} \alpha_y \beta_y \sqrt{\pi_y} \left[1 - \text{softmax}_{y} (\beta_y B_{y}(f) + \Delta_y) \right].
$$

Dynamic adjustment based on a fine-grained generalization bound

Algorithm 1: Principled Learning Algorithm induced by the Theoretical Insights

Require: Training set $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^N$ and a model $f$ parameterized by $\Theta$.

1: Initialize the model parameters $\Theta$ randomly.
2: for $t = 1, 2, \cdots, T$ do
3: \hspace{1cm} $B \leftarrow \text{SampleMiniBatch}(\mathcal{S}, m)$ \hspace{1cm} $\triangleright$ A mini-batch of $m$ samples
4: if $t < T_0$ then
5: \hspace{1cm} Set $\alpha = 1, \beta_y, \Delta_y$ \hspace{1cm} $\triangleright$ Adjust logits during the initial phase
6: \hspace{1cm} else
7: \hspace{1.5cm} Set $\alpha_y \propto \pi_y^{-\nu}, \beta_y = 1, \Delta_y, \nu > 0$ \hspace{1cm} $\triangleright$ TLA and ADRW
8: \hspace{1cm} end if
9: \hspace{1cm} $L(f, B) \leftarrow \frac{1}{m} \sum_{(x, y) \in B} L_{VS}(f(x), y)$ \hspace{1cm} $\triangleright$ Calculate the loss
10: \hspace{1cm} $\Theta \leftarrow \Theta - \eta \nabla_{\Theta} L(f, B)$ \hspace{1cm} $\triangleright$ One SGD step
11: \hspace{1cm} Optional: anneal the learning rate $\eta$. \hspace{1cm} $\triangleright$ Required when $t = T_0$
12: end for

Both representation and classifier matter

- **Motivation:** Representation and classifier are in the different learning pace, and can be treated differently during training.

- **The problem behind long-tail:**
  
  Classification performance = Representation Quality + Classifier Quality

- **Ways of learning classifiers:**
  
  Classifier Re-training (cRT)

Nearest Class Mean classifier (NCM)

$\tilde{w}_i = \frac{w_i}{||w_i||^\tau}$

$\tilde{w}_i = f_i \ast w_i$, where $f_i = \frac{1}{||w_i||^\tau}$

The special constant classifier geometry

Motivation: In the training on an imbalanced dataset, the classifier vectors of minor classes will be merged, termed as minority collapse, which breaks up the ETF structure and deteriorates the performance on test data.

Simplex Equiangular Tight Frame:

\[
M = \sqrt{\frac{K}{K-1}} U \left( I_K - \frac{1}{K} 1_K 1_K^T \right)
\]

\[
m_i^T m_j = \frac{K}{K-1} \delta_{i,j} - \frac{1}{K-1}, \forall i, j \in [1, K]
\]

ETF Classifier:

\[
\min_H \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \mathcal{L}_{CE}(h_{k,i}, W^*)
\]

\[
w_k^* w_{k'}^* = E_W \left( \frac{K}{K-1} \delta_{k,k'} - \frac{1}{K-1} \right), \forall k, k' \in [1, K]
\]

Dot-Regression Loss:

Squared loss function:

\[ \mathcal{L}_{DR}(h, W^*) = \frac{1}{2\sqrt{E_WE_H}} \left( w_c^T h - \sqrt{E_WE_H} \right)^2 \]

\[ \frac{\partial \mathcal{L}_{DR}}{\partial h} = -(1 - \cos \angle(h, w_c^*)) \cdot w_c^* \]

Table:

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Methods</th>
<th>Acc. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>90</td>
<td>Learnable Classifier + CE</td>
<td>34.6</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>41.8</td>
</tr>
<tr>
<td>120</td>
<td>Learnable Classifier + CE</td>
<td>41.9</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>43.2</td>
</tr>
<tr>
<td>150</td>
<td>Learnable Classifier + CE</td>
<td>42.5</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>43.8</td>
</tr>
<tr>
<td>180</td>
<td>Learnable Classifier + CE</td>
<td>44.3</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>44.7</td>
</tr>
</tbody>
</table>

Backbone | Methods                     | Acc. (%) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-34</td>
<td>Learnable Classifier + CE</td>
<td>82.2</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>83.0</td>
</tr>
<tr>
<td>ResNet-50</td>
<td>Learnable Classifier + CE</td>
<td>85.5</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>86.1</td>
</tr>
<tr>
<td>ResNet-101</td>
<td>Learnable Classifier + CE</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>ETF Classifier + DR</td>
<td>87.0</td>
</tr>
</tbody>
</table>

Table:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.005</td>
<td>0.01</td>
<td>0.02</td>
<td>0.005</td>
</tr>
<tr>
<td>Learnable Classifier + CE</td>
<td>67.3</td>
<td>72.8</td>
<td>78.6</td>
<td>38.7</td>
</tr>
<tr>
<td>ETF Classifier + DR</td>
<td>71.9</td>
<td>76.5</td>
<td>81.0</td>
<td>40.9</td>
</tr>
<tr>
<td>Improvements</td>
<td>+4.6</td>
<td>+3.7</td>
<td>+2.4</td>
<td>+2.2</td>
</tr>
<tr>
<td>DenseNet</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learnable Classifier + CE</td>
<td>71.1</td>
<td>77.7</td>
<td>84.1</td>
<td>40.3</td>
</tr>
<tr>
<td>ETF Classifier + DR</td>
<td>72.9</td>
<td>78.5</td>
<td>83.4</td>
<td>40.4</td>
</tr>
<tr>
<td>Improvements</td>
<td>+1.8</td>
<td>+0.8</td>
<td>-0.7</td>
<td>-0.2</td>
</tr>
</tbody>
</table>

Summary: supervised long-tailed learning in the context of deep learning, should take more factors (loss, representation etc.) into account to improve the performance of imbalanced learning.
Weakly-Supervised Long-tailed Learning

Weak supervision

- Incomplete supervision: **semi-supervised learning**
- Inexact supervision: **partial label learning, multi-instance learning**
- Inaccurate supervision: **label-noise learning**

Target: *low labeling cost but high accuracy*.

Pseudo Labeling

- Generate more precise annotations for weakly supervised data

Transfer weakly-supervised learning → supervised learning

Weakly-Supervised Long-tailed Learning

When long-tailed distribution meets weak supervision

- Positive feedback property of pseudo-labeling: head get better and more

(a) Imbalanced class distribution  
(b) Bias of pseudo-labels  
(c) Accuracy gain from SSL

- Distributional corrections need to go deeper into training and pseudo-labeling

Motivation: Label propagation on unlabeled data biased toward the majority classes.

Main idea: Refine the original, biased pseudo-labels so that their distribution can match the true class distribution of unlabeled data, while constraining the refined pseudo-labels to be close from the original ones.

Formulation: An optimization problem for constructing refined pseudo-labels: minimizing the distortion from the original pseudo-labels, while matching the true class distribution.

\[
\begin{align*}
\text{minimize} & \quad \sum_{m=1}^{M} w_m D_{KL}(\hat{y}_m \parallel \hat{y}_{m}^{\text{unlabeled}}) \\
\text{subject to} & \quad \sum_{m=1}^{M} \hat{y}_m(k) = M_k, \quad \forall k, \quad \sum_{k=1}^{K} \hat{y}_m(k) = 1, \quad \forall m, \quad \hat{y}_m(k) \in [0, 1], \quad \forall m, k
\end{align*}
\]

Algorithm 1: DualCoordinateAscent: Coordinate ascent algorithm for dual of (1)

Require: \(\{\hat{y}_m^{\text{unlabeled}}\}_{m=1}^{M}, \{w_m\}_{m=1}^{M}, \{M_k\}_{k=1}^{K}, T\)
Ensure: The unique solution of (1)

1: \(\hat{y}_m^0 \leftarrow \hat{y}_m^{\text{unlabeled}}, \quad \alpha_m^0 \leftarrow 1, \quad \beta_k^0 \leftarrow 1, \quad \forall m, k\)
2: \(\textbf{for } t = 1 \textbf{ to } T \textbf{ do}\)
3: \(\textbf{if } t \text{ is odd or } t = T \text{ then}\)
4: \(\beta_k^t \leftarrow \beta_k^{t-1}, \quad \alpha_m^t \leftarrow \left(\sum_{k=1}^{K} \hat{y}_m^0(k) \beta_k^{t-1} \frac{1}{w_m}\right)^{-1}, \quad \forall m, k,\)
5: \(\textbf{else}\)
6: \(\alpha_m^t \leftarrow \alpha_m^{t-1}, \quad \beta_k^t \leftarrow \text{Solve}_{Z \geq 0} \left(\sum_{m=1}^{M} \hat{y}_m^0(k) \alpha_m^{t-1} Z \frac{1}{w_m} - M_k\right), \quad \forall m, k\)
7: \(\textbf{end if}\)
8: \(\textbf{end for}\)
9: \(\hat{y}_m^{\text{out}}(k) \leftarrow \hat{y}_m^0(k) \alpha_m^T(\beta_k^T \frac{1}{w_m}), \quad \forall m, k\)
ABC: Auxiliary Balanced Classifier [1]

- Motivation: high-quality representations can be learned even the classifier is biased [2].
- Train an Auxiliary Balanced Classifier (ABC) by resampling a balanced subset while using the high-quality representations learned from all data.

Long-tailed Partial Label Learning

Dataset $\mathcal{X} = \{(x_i, y_i, S_i): i \in (1, 2, 3, \ldots, N)\}$, where any $x_i$ is associated with a candidate label set $S_i$ and its ground truth $y_i \in S_i$ is invisible.

- The sample number of $L$ classes in descending order
  - $N_1 \geq N_2 \geq \cdots \geq N_L$, Imbalance ratio $\gamma = \frac{N_1}{N_L} \gg 1$

- Main challenges:
  - Tail samples cannot be correctly recognized even in training
  - No available class prior
**RECORDS: Rebalancing for dynamic bias**

- Observation: even after applying an oracle class distribution prior in the training, existing long-tailed techniques underperform in LT-PLL and even fail in some cases.
- Existing long-tailed techniques: leverage a constant class distribution prior to rebalance the training and does not consider the dynamic of label disambiguation.
- A dynamic rebalancing mechanism friendly to the training dynamic.

**Constant Rebalancing**

\[
z_{\text{uni}}^y(x) = z^y(x) - \log P_{\text{train}}(y|\Theta) = z^y(x) - \log \text{softmax}(g^y(F; W))
\]

**Dynamic Rebalancing**

\[
z_{\text{uni}}^y(x) = z^y(x) - \log P_{\text{train}}(y|\Theta) = z^y(x) - \log \text{softmax}(g^y(F; W))
\]

Match the training dynamic

---

**RECORDS: Rebalancing for dynamic bias**

- Theory: alongside the label disambiguation, the dynamic estimation can **progressively approach to the oracle class distribution**.

**Proposition 1.** Let \( \eta = \sup_{(x,y) \in \mathcal{X} \times \mathcal{Y} \setminus \mathcal{Y}} P_{\mathcal{S}}(x,y | y \in S) \) denote the ambiguity degree, \( d_H \) be the Natarajan dimension of the hypothesis space \( H \), \( \hat{h} = h_{\Theta} \) be the optimal classifier on the basis of the label disambiguation, where \( \Theta = \arg \min_{\Theta} R_{\text{train}}(\Theta) \). If the small ambiguity degree condition (Cour et al., 2011a; Liu & Dietterich, 2014) satisfies, namely, \( \eta \in [0, 1) \), then for \( \forall \delta > 0 \), the \( L_2 \) distance between \( P_{\text{train}}(y) \) and \( P_{\text{train}}(y | \Theta) \) given \( \hat{h} \) is bounded as

\[
L_2(\hat{h}) < \frac{4}{(\ln 2 - \ln(1 + \eta))N} (d_H(\ln 2N + 2 \ln C) - \ln \delta + \ln 2)
\]

with probability at least \( 1 - \delta \), where \( N \) is the sample number and \( C \) is the category number.

<table>
<thead>
<tr>
<th></th>
<th>CIFAR-10-LT</th>
<th>CIFAR-100-LT</th>
<th>PASCAL VOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imbalance ratio ( \rho )</td>
<td>0.3</td>
<td>0.5</td>
<td>0.7</td>
</tr>
<tr>
<td><strong>Ambiguity q</strong></td>
<td>71.31</td>
<td>39.69</td>
<td>25.23</td>
</tr>
<tr>
<td>LW</td>
<td>57.77</td>
<td>41.68</td>
<td>21.32</td>
</tr>
<tr>
<td>CAVL</td>
<td>74.22</td>
<td>56.25</td>
<td>45.11</td>
</tr>
<tr>
<td>PRODEN</td>
<td>77.53</td>
<td>58.35</td>
<td>45.66</td>
</tr>
<tr>
<td>CORR</td>
<td>83.88</td>
<td>76.55</td>
<td>54.61</td>
</tr>
<tr>
<td>SoLar</td>
<td>LW+RECORDS</td>
<td>77.22</td>
<td>59.88</td>
</tr>
<tr>
<td>CAVL+RECORDS</td>
<td>68.93</td>
<td>63.59</td>
<td>44.61</td>
</tr>
<tr>
<td>PRODEN+RECORDS</td>
<td>81.23</td>
<td>78.82</td>
<td>68.03</td>
</tr>
<tr>
<td>CORR+RECORDS</td>
<td><strong>84.25</strong></td>
<td><strong>82.5</strong></td>
<td><strong>71.24</strong></td>
</tr>
</tbody>
</table>

(a) \( L_2 \) distance during training (b) Estimated class distribution

RCAL: Representation Calibration

- Motivation: The representations of unsupervised contrastive learning are not influenced by corrupted labels and thus naturally robust.

- Main spirit: Decoupling the imbalance from the noise labels with the help of self-supervised representation learning methods.

- Based upon the achieved representations, the calibration can be performed. Distributional representation calibration estimate the robust feature distribution considering data imbalance and perform balanced sampling. Individual representation calibration constrain the distance between supervised training representations and pre-training representations.

**RCAL: Representation Calibration**

**Algorithm 1** Algorithm of the proposed method RCAL

**Require:** the training dataset $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{n}$, regularization strength $\beta$, scalar temperature $\tau$, confidence weight $\gamma$, the pre-training epochs $T_p$, max epochs $T_m$.  

1: for $t = 1, ..., T_p$ do
2: **Pre-train** the encoder network $f$ with MoCo [20].
3: end for

4: Extract deep representations of instances with $z = f(x)$.  
5: for $c = 1, ..., K$ do
6: **Perform** the LOF algorithm for the $c$-th class and obtain preserved examples $\mathcal{S}_c'$.  
7: **Build** the multivariate Gaussian distribution $N(f(x)|\mu_c, \Sigma_c)$ for $c$-th class using $\mathcal{S}_c'$.  
8: end for

9: **Calibrate** the multivariate Gaussian distributions of tail classes with the statistics of head classes.
10: **Sample** data points from achieved multivariate Gaussian distributions of all classes.

11: for $t = T_p + 1, ..., T_m$ do
12: Add distance constraints between learned representations and representations brought by contrastive learning.
13: **Adopt** the mixup technology to original examples.
14: **Train** the encoder $f$ and the linear head $h$ simultaneously on the training dataset and sample data points with the training loss in Eq. (2).
15: end for

16: return The robust classifier $h(f(x))$ for testing.

---

**Table**: Performance comparison of different methods on the CIFAR-10 dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Imbalance Ratio</th>
<th>10</th>
<th>100</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Noise Rate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERN</td>
<td>80.41</td>
<td>75.61</td>
<td>71.94</td>
<td>70.13</td>
</tr>
<tr>
<td>ERN-DRW</td>
<td>81.72</td>
<td>77.61</td>
<td>71.94</td>
<td>70.13</td>
</tr>
<tr>
<td>LDAM</td>
<td>84.59</td>
<td>82.37</td>
<td>77.48</td>
<td>71.41</td>
</tr>
<tr>
<td>LDAM-DRW</td>
<td>85.94</td>
<td>83.73</td>
<td>80.20</td>
<td>74.87</td>
</tr>
<tr>
<td>CRT</td>
<td>80.22</td>
<td>76.15</td>
<td>74.17</td>
<td>70.05</td>
</tr>
<tr>
<td>NCM</td>
<td>82.33</td>
<td>74.73</td>
<td>74.76</td>
<td>68.43</td>
</tr>
<tr>
<td>MSLAS</td>
<td>87.58</td>
<td>85.21</td>
<td>83.39</td>
<td>76.16</td>
</tr>
<tr>
<td>Co-teaching</td>
<td>80.30</td>
<td>78.54</td>
<td>68.71</td>
<td>57.10</td>
</tr>
<tr>
<td>CDR</td>
<td>81.68</td>
<td>78.09</td>
<td>73.86</td>
<td>68.12</td>
</tr>
<tr>
<td>Sel-CL+</td>
<td>86.47</td>
<td>85.11</td>
<td>84.41</td>
<td>80.35</td>
</tr>
<tr>
<td>HAR-DRW</td>
<td>84.09</td>
<td>82.43</td>
<td>80.41</td>
<td>77.43</td>
</tr>
<tr>
<td>RoLT</td>
<td>85.68</td>
<td>85.43</td>
<td>83.50</td>
<td>80.92</td>
</tr>
<tr>
<td>RoLT-DRW</td>
<td>86.24</td>
<td>85.49</td>
<td>81.99</td>
<td>81.99</td>
</tr>
<tr>
<td>RCAL (Ours)</td>
<td>88.09</td>
<td>86.46</td>
<td>84.58</td>
<td>83.43</td>
</tr>
</tbody>
</table>

**Table**: Performance comparison of different methods on the CIFAR-100 dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Imbalance Ratio</th>
<th>CIFAR-10</th>
<th>CIFAR-100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise Rate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
</tr>
<tr>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Summary

- Weak Supervision can **exacerbate the long-tail effect**, impairing the performance of minority classes.
- It is critical to construct a friendly mechanism to decouple *pseudo labeling / sample selection* and *rebalancing*.
Self-supervised learning

- The idea of contrastive learning: push augmented views of the same image closer, and embeddings to random pairs of images far apart.

Is self-supervised learning more robust to data imbalance?

- Supervised learning (SL) only extracts features that are useful for predicting labels ($e_1$)
- Self-supervised learning (SSL) learns task-irrelevant features regardless of the labels, which enables richer and more robust representation ($e_1, e_2$)

Self-supervised learning still suffers from data imbalance

- Performance degeneration: Linear probing on imbalanced data ($D_i$) and balanced data ($D_b$) with same data amount

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Subset</th>
<th>Many</th>
<th>Medium</th>
<th>Few</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR10</td>
<td>$D_b$</td>
<td>77.14 ± 4.64</td>
<td>74.25 ± 6.54</td>
<td>71.47 ± 7.55</td>
<td>74.57 ± 0.65</td>
</tr>
<tr>
<td></td>
<td>$D_i$</td>
<td>76.07 ± 3.88</td>
<td>67.97 ± 5.84</td>
<td>54.21 ± 10.24</td>
<td>67.08 ± 2.15</td>
</tr>
<tr>
<td>CIFAR100</td>
<td>$D_b$</td>
<td>25.48 ± 1.74</td>
<td>25.16 ± 3.07</td>
<td>24.01 ± 1.23</td>
<td>24.89 ± 0.99</td>
</tr>
<tr>
<td></td>
<td>$D_i$</td>
<td>30.72 ± 2.01</td>
<td>21.93 ± 2.61</td>
<td>15.99 ± 1.51</td>
<td>22.96 ± 0.43</td>
</tr>
</tbody>
</table>

- Representation learning disparity: head classes dominate the feature regime but tail classes passively collapse

BCL: Boosted Contrastive Learning

- Motivation I: Memorization effect still holds under long-tailed distribution.
- Motivation II: Stronger information discrepancy motivates tail samples mining

Challenge: how to detect tail data and how to construct the desired information discrepancy

Motivated from the observation that learning speed-based proxy shows strong correlation with the memorization score[1], BCL extends the memorization estimation to self-supervised learning.

$$\mathcal{L}_{i,0}^m = \mathcal{L}_{i,0}, \quad \mathcal{L}_{i,t}^m = \beta \mathcal{L}_{i,t-1}^m + (1 - \beta) \mathcal{L}_{i,t}$$

$$M_{i,t} = \frac{1}{2} \left( \frac{\mathcal{L}_{i,t}^m - \mathcal{L}_{i,t}^m}{\max \{ |\mathcal{L}_{i,t}^m - \mathcal{L}_{i,t}^m|_{i=0,\ldots,N} + 1 \} } \right)$$

Adaptively assigns the appropriate augmentation strength for the individual sample according to the feedback from the memorization clues

$$\Psi(x_i; A, M_i) = a_1(x_i) \circ \ldots \circ a_k(x_i),$$

$$a_j(x_i) = \begin{cases} A_j(x_i; M_i \zeta) & u \sim \mathcal{U}(0, 1) \& u < M_i \\ x_i & \text{otherwise} \end{cases}$$

$$\mathcal{L}_{BCL} = \frac{1}{N} \sum_{i=1}^{N} - \log \frac{\exp \left( \frac{f(\Psi(x_i))^\top f(\Psi(x_i'))}{\tau} \right)}{\sum_{x' \in X'} \exp \left( \frac{f(\Psi(x_i))^\top f(\Psi(x_i'))}{\tau} \right)}$$


BCL: Boosted Contrastive Learning

- Calculate **memorization scores** based on historical statistics to detect tail.
- Construct **instance-wise augmentations** to enhance representation learning.

DnC: Divide and Contrast: Self-Supervised Learning from Uncurated Data

**Motivation:** Conquer-and-Divide Training to isolate the negative effect on tail classes during training.

SDCLR: Self-damaging Contrastive Learning

- **Intuition:** The sensitivity of head and tail samples to the model pruning, are very different, which helps us to anchor and promote the training of tail samples.

- **Pruning identified exemplars (PIE)*** systematically investigates the model output changes introduced by pruning and finds that certain examples are particularly sensitive to sparsity. They are high likely to be rare and atypical samples, which probably comes from tail classes.

Coverage of embedding space during training. For small $\tau$ the representations are more uniformly distributed.

**Observation:** TS investigates the role of the temperature parameter $\tau$ in the contrastive loss, and find that a large $\tau$ emphasizes group-wise discrimination, whereas a small $\tau$ leads to a higher degree of instance discrimination.

**Temperature Schedules (TS):** alternates between an upper $\tau$ and a lower $\tau$ bound at a fixed period length $T$.

$$\tau_{\cos}(t) = (\tau_+ - \tau_-) \times (1 + \cos(2\pi t/T))/2 + \tau_-$$

---

Representations of a head and a tail class. Red: head class and blue: tail class. Small $\tau = 0.1$ promotes uniformity, while large $\tau = 1.0$ creates dense clusters. Tail classes benefit from instance discrimination.

---

<table>
<thead>
<tr>
<th>Method</th>
<th>CIFAR-10-LT</th>
<th>ImageNet-100-LT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>kNN@10</td>
<td>FS</td>
</tr>
<tr>
<td>SimCLR</td>
<td>60.19</td>
<td>68.29</td>
</tr>
<tr>
<td>SDCLR</td>
<td>60.74</td>
<td>71.03</td>
</tr>
<tr>
<td>SimCLR + TS</td>
<td>62.91</td>
<td>71.86</td>
</tr>
</tbody>
</table>

GH: Geometric Harmonization

- Why the conventional contrastive learning underperforms in self-supervised long-tailed context?

  Conventional contrastive loss motivates *sample-level uniformity*, which is biased towards the head classes.

- Geometric Harmonization aims at achieves *category-level uniformity*, i.e., equal allocation w.r.t. classes.

### Challenges

- **Challenges I:** No guarantee for the desired category-level uniformity
- **Challenges II:** The latent true labels are not available, while the estimated labels are noisy

#### Geometric Uniform Structure

\[
M_i^T \cdot M_j = C, \ \forall i, j \in \{1, 2, \ldots, K\}, \ i \neq j,
\]

Any two vectors in \( M \) have the same angle, namely, the unit space are equally partitioned by the vectors.

#### Surrogate Label Allocation

\[
\min_{\hat{Q} = [\hat{q}_1, \ldots, \hat{q}_N]} \mathcal{L}_{GH} = -\frac{1}{|D|} \sum_{x_i \sim D} \hat{q}_i \log q_i,
\]

s.t. \( \hat{Q} \cdot \mathbb{1}_N = N \cdot \pi, \ \hat{Q}^T \cdot \mathbb{1}_K = \mathbb{1}_N, \)

**Overall objective**

\[
\min_{\theta, Q} \mathcal{L} = \mathcal{L}_{\text{InfoNCE}} + w_{GH} \mathcal{L}_{GH},
\]

---

Summary

- Self-supervised learning is more robust to data imbalance than the supervised counterpart.

- However, self-supervised learning still suffers from the long-tailed distribution, resulting in performance degeneration and representation learning disparity.
Still require more efforts on this way
Thank you

Q & A

Tutorial website: https://tmlr-group.github.io/tutorials/aaai2024.html