Trustworthy Machine Learning on Imbalance Data

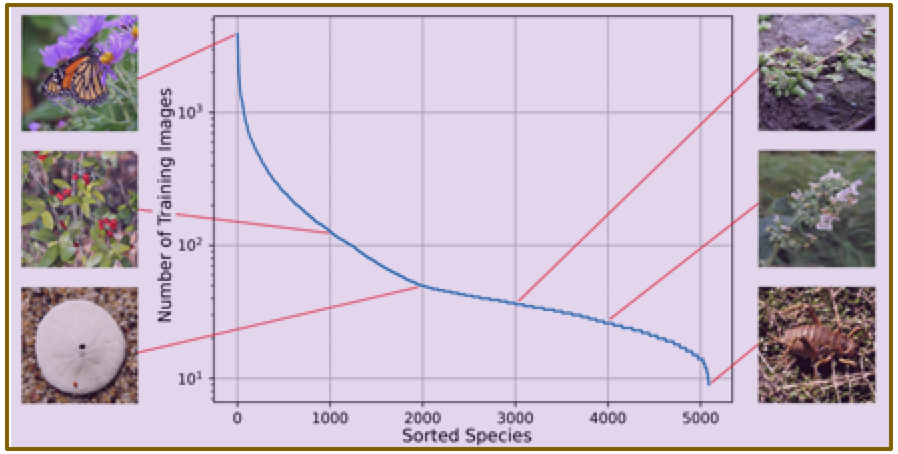
Jiangchao Yao

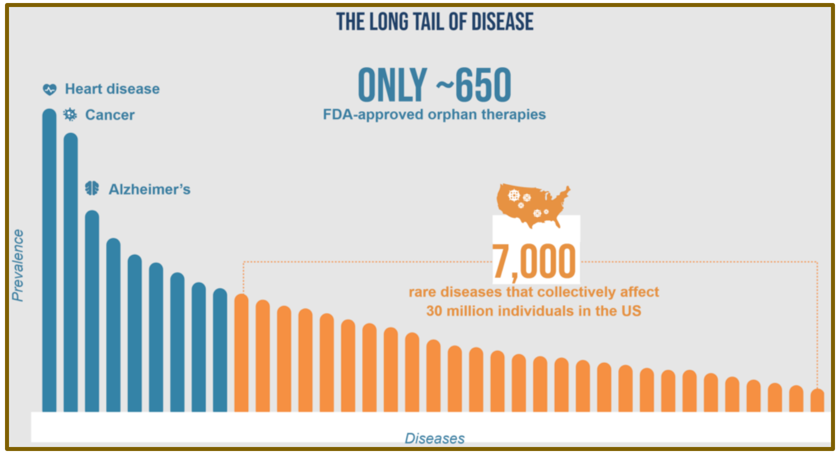
CMIC, Shanghai Jiao Tong University

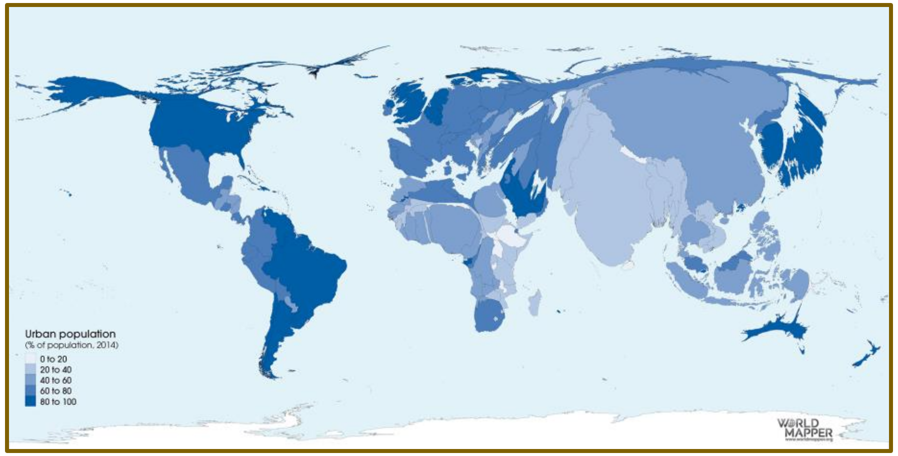
饮 水 思 源 • 爱 国 荣 校

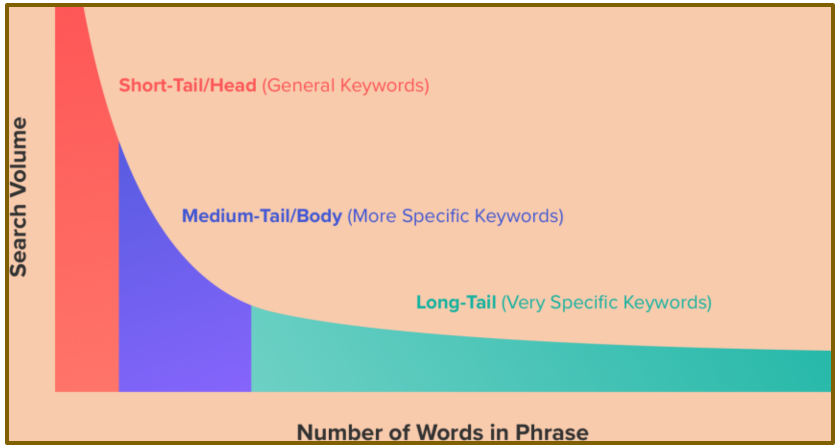
The Imbalance Nature of Data

Large-scale natural sources are very imbalance, usually following a long-tailed distribution.









[1] Van Horn et al. The iNaturalist Species Classification and Detection Dataset. CVPR 2018.

[2] Gregory et al. CXR-LT challenge. ICCV CVAMD 2023.

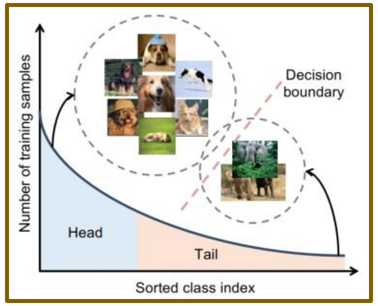
[3] https://worldmapper.org/maps/urban-population-relative-2014/

[4] https://seopressor.com/blog/short-tail-or-long-tail-keywords/

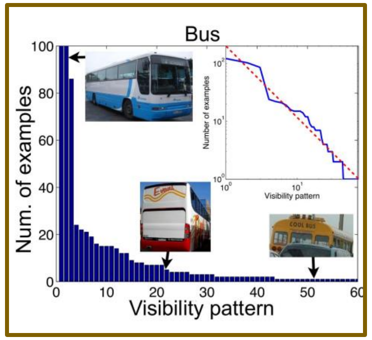
How Imbalance Data Matters in Machine Learning

A direct decomposition on the risk minimization

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



Minority ("generalized" conceptual) classes have weak importance for training, which can 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)*.



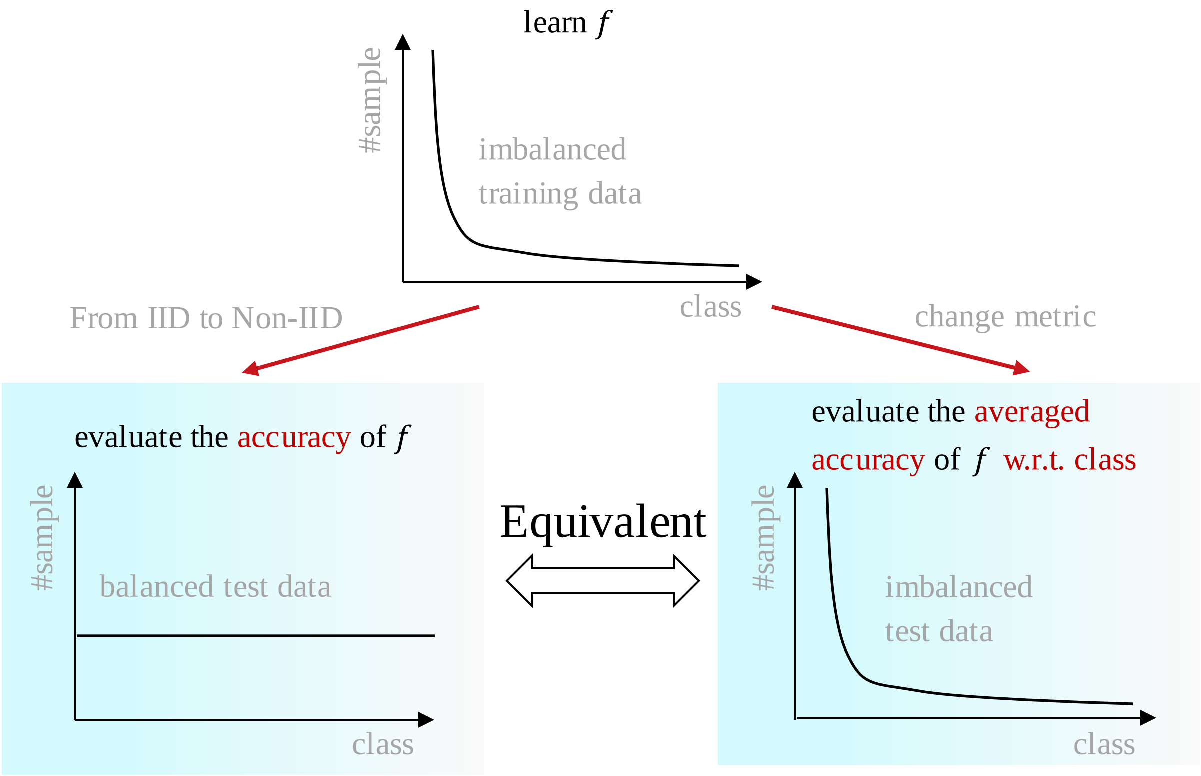
However, in real applications, the value of classes cannot be absolutely characterized by their quantity, and instead, sometimes **less is more** for sustainable long-term development.

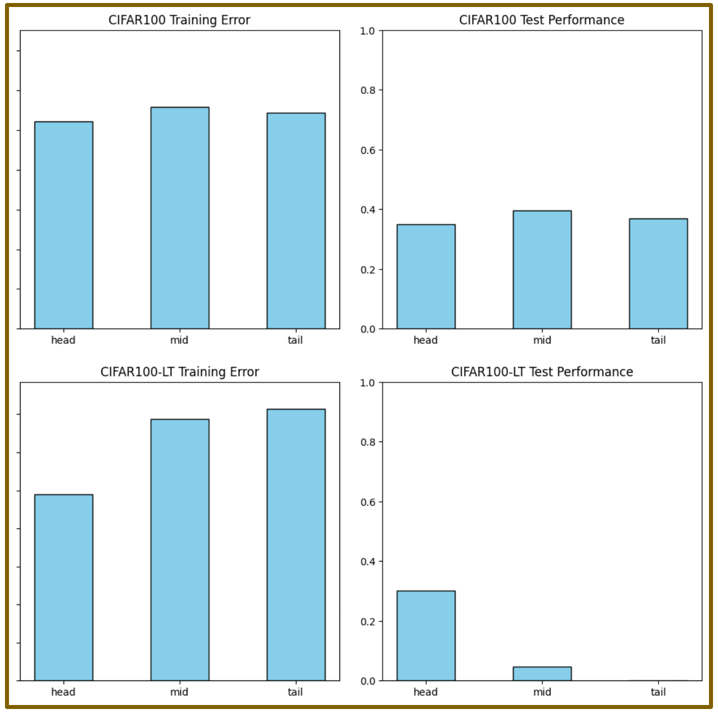
- **Fairness** w.r.t. diversity e.g., small populations of gender, race and consumers
- **Cost-sensitive scenarios** e.g., medical disease diagnosis and treatment

[1] Vitaly Feldman. "Does Learning Require Memorization? A Short Tale about a Long Tail." SIGACT 2020.

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 Historical Development

The development of imbalance learning

1998 — 2009 — 2018 — 2023


Optimizing Classifiers for Imbalanced Training Sets

Grigoris Karakoulas
Global Analytics Group
Canadian Imperial Bank of Commerce
161 Bay St., BCE-11,
Toronto ON, Canada M5J 2S8

John Shawe-Taylor
Department of Computer Science
Royal Holloway, University of London
Egham, TW20 0EX
England

Abstract

1 Introduction

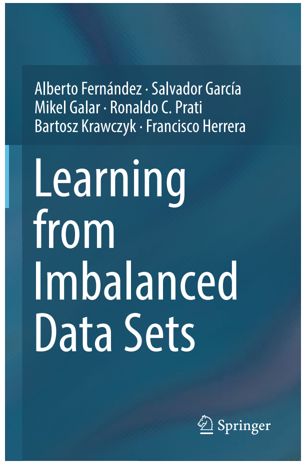
methods


Learning from Imbalanced Data

Haibo He, Member, IEEE, and Edwardo A. Garcia

1 Introduction

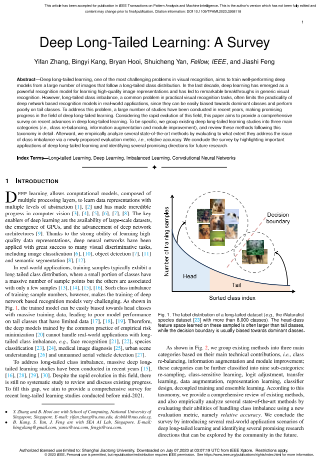
Applications


Alberto Fernández · Salvador García
Mikel Galar · Ronaldo C. Prati
Bartosz Krawczyk · Francisco Herrera

Learning from Imbalanced Data Sets

Springer

Deep learning


Deep Long-Tailed Learning: A Survey

Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, Fellow, IEEE, and Jiashi Feng

1 Introduction

[1] Karakoulas et al. Optimizing Classifers for Imbalanced Training Sets. NIPS 1998.

[2] He et al. Learning from Imbalanced Data. TKDE 2009.

[3] Fernández et al. Learning from Imbalanced Data Sets. Springer, 2018.

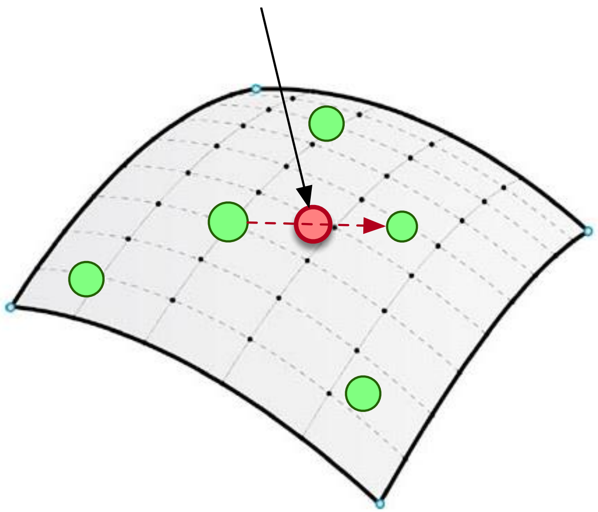
[4] Zhang et al. Deep Long-tailed Learning: A Survey. TPAMI 2023.

Retrospection-I: Re-Sampling

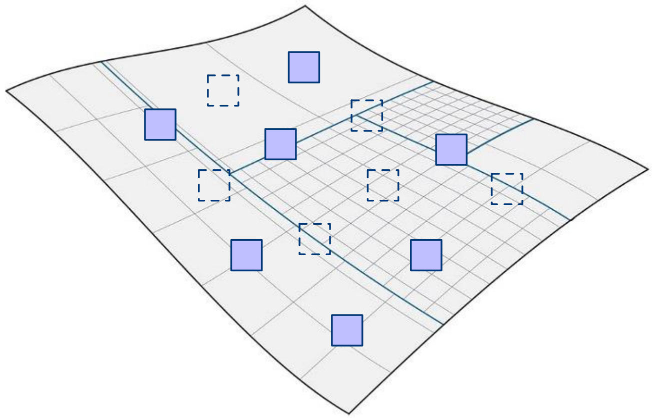
SMOTE: Synthetic Minority Over-sampling Technique

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

Interpolation on minority manifold



Under-sampling in the majority class



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

Chawla, Nitesh V., et al. "SMOTE: Synthetic Minority Over-Sampling Technique." *JAIR 2002*.

Retrospection-II: Threshold-Moving Method

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

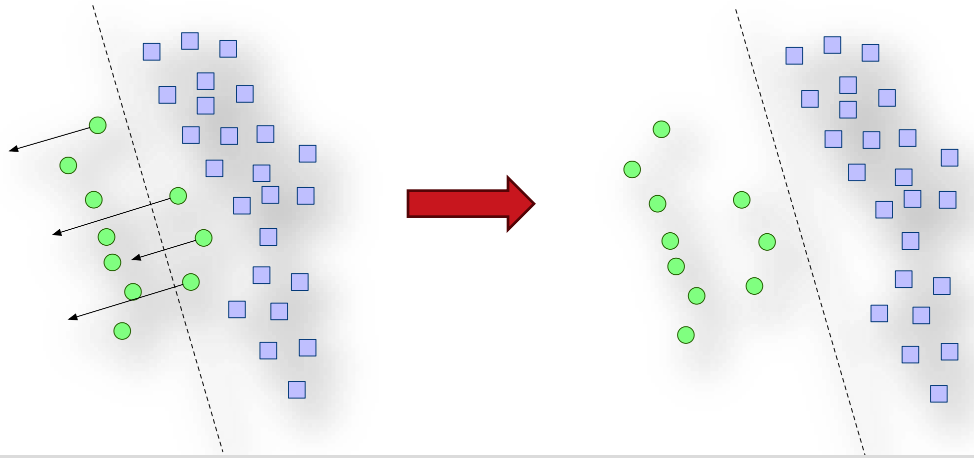
Motivation: The over-confident prediction for majority or the low-confident prediction for minority can be calibrated after training.

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.

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



Moving function

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

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

Let $p_k = \frac{e^{z_k}}{\sum_{k'}^{K} e^{z_{k'}}}$ denote the class prediction. If we set $\sum_{k'=1}^{K} C[k][k'] = e^{-\tau \log \pi_k}$ where π_k is the class prior and τ 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.

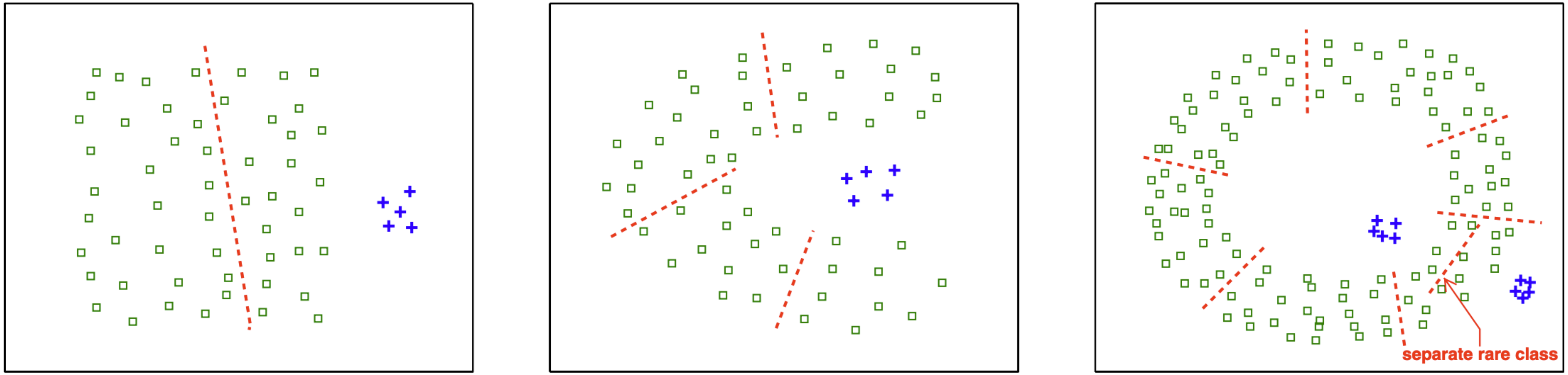
Zhou, Z. H., & Liu, X. Y. Training Cost-Sensitive Neural Networks with Methods Addressing the Class Imbalance Problem. *TKDE, 2005*.

Retrospection-III: Pseudo Labeling

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

```
for class i = 1 to c    // "c" represents #classes
    clusterLabel(i) = Clustering(D(i), K(i));
    D(i)* = changeLabel(D(i), clusterLabel(i));
end for
```

Phase II: over-sampling (for COG-OS only)

```
for class j = 1 to c
    D(j)** = replicate(D(j)*, r(j));
end for
D** = ∪_{j=1}^{c} (D(j)**);
```

Phase III: training

```
M = train(D**, L);
```

Phase IV: predicting

```
p' = predict(T, M);
p = convertLabel(p');
```

Wu. J. J. et al. COG: Local Decomposition for Rare Class Analysis. *DMKD, 2010*.

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.

$$\text{Accuracy} = \mathrm{E}_{p(x,y)}[h(x) = y] \Rightarrow \text{Balanced Accuracy} = \frac{\sum_{k\in\{-1,1\}} \mathrm{E}_{p(x|y=k)}[h(x)=k]}{2}$$

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 $\widehat{p}_S$ denote any estimator of $p = \mathbf{P}(y = 1)$ satisfying $\widehat{p}_S \in (0, 1)$ and $\widehat{p}_S \xrightarrow{P} p$. Let $\widehat{\eta}_S : \mathcal{X} \to [0, 1]$ denote any class probability estimator satisfying $\mathbf{E}_x\left[|\widehat{\eta}_S(x) - \eta(x)|^r\right] \xrightarrow{P} 0$ for some $r \geq 1$, and let $h_S(x) = \boxed{\mathrm{sign}(\widehat{\eta}_S(x) - \widehat{p}_S).}$ Then*

$$\mathrm{regret}_D^{\mathrm{AM}}[h_S] \xrightarrow{P} 0\,.$$

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 $\ell : \{\pm 1\} \times \bar{\mathbb{R}} \to \bar{\mathbb{R}}_+$, with link function $\psi : [0, 1] \to \bar{\mathbb{R}}$; (b) RKHS $\mathcal{F}_K$ with positive definite kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$; (c) regularization parameter $\lambda_n > 0$
3: $f_S \in \mathrm{argmin}_{f \in \mathcal{F}_K} \left\{\frac{1}{n} \sum_{i=1}^n \ell(y_i, f(x_i)) + \lambda_n \|f\|_K^2\right\}$
4: $\widehat{\eta}_S = \psi^{-1} \circ f_S$
5: $\widehat{p}_S =$ (as in Eq. (2))
6: **Output:** Classifier $h_S(x) = \mathrm{sign}(\widehat{\eta}_S(x) - \widehat{p}_S)$

Menon, Aditya, et al. "On the Statistical Consistency of Algorithms for Binary Classification under Class Imbalance." *ICML 2013*.

Summary

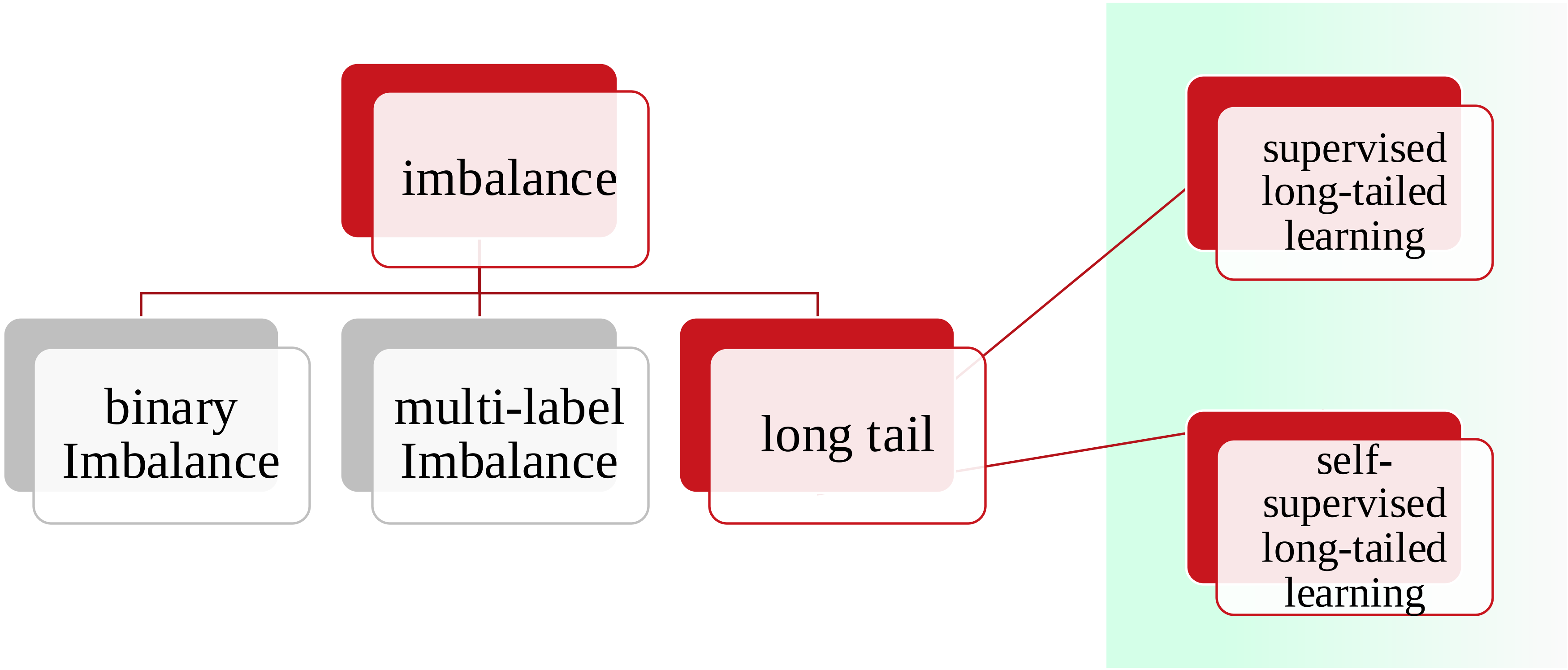
Summary of imbalanced learning in the early years

- Imbalanced Learning
  - Re-Sampling
    - Control sample quantity from the input augmentation perspective
  - Loss Adjustment
    - Adjust penalty preference from the loss perspective.
  - Re-Labeling
    - Control learning complexity from the label manipulation perspective
  - Statistical Consistency
    - The statistically-consistent loss family for imbalanced binary classification

What is the new of this topic in the recent years?

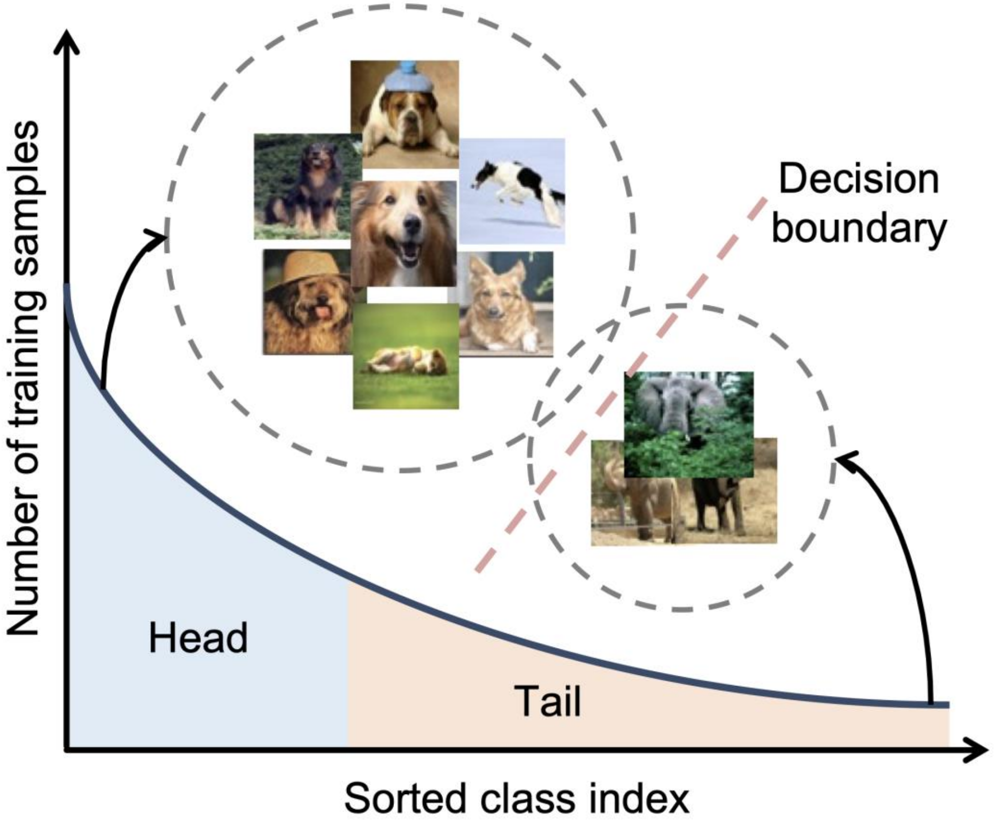
The Following Part in This Tutorial

The recent advances of imbalance learning powered by deep learning



Supervised Long-tailed Learning

It has been contributed with very broad explorations



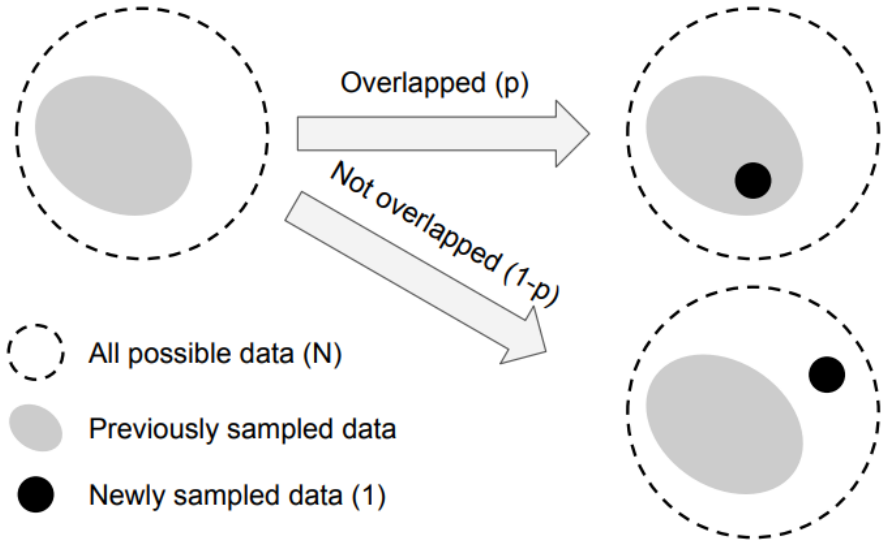
Method	Year	Class Re-balancing			Augmentation		Module Improvement				Target Aspect
		Re-sampling	CSL	LA	TL	Aug	RL	CD	DT	Ensemble	
LMLE [89]	2016						✓				feature
HFL [90]	2016						✓				feature
Focal loss [54]	2017		✓								objective
Range loss [21]	2017						✓				feature
CRL [50]	2017						✓				feature
MetaModelNet [91]	2017				✓						
DSTL [92]	2018				✓						
DCL [93]	2019	✓									sample
Meta-Weight-Net [94]	2019		✓								objective
LDAM [18]	2019		✓								objective
CB [16]	2019		✓								objective
UML [95]	2019		✓								feature
FTL [96]	2019				✓	✓					feature
Unequal-training [48]	2019						✓				feature
OLTR [15]	2019						✓				feature
Balanced Meta-Softmax [97]	2020	✓	✓								sample, objective
Decoupling [32]	2020	✓	✓				✓	✓	✓		feature, classifier
LST [98]	2020	✓			✓						sample
Domain adaptation [28]	2020		✓								objective
Equalization loss (ESQL) [19]	2020		✓								objective
DBM [22]	2020		✓								objective
Distribution-balanced loss [37]	2020		✓								objective
UNO-IC [99]	2020			✓							prediction
De-confound-TDE [45]	2020			✓				✓			prediction
M2m [100]	2020				✓	✓					sample
LEAP [49]	2020				✓	✓	✓				feature
OFA [101]	2020				✓	✓			✓		feature
SSP [102]	2020				✓		✓				feature
LFME [103]	2020				✓					✓	sample, model
IEM [104]	2020						✓				feature
Deep-RTC [105]	2020							✓			classifier
SimCal [34]	2020								✓	✓	sample, model
BBN [44]	2020									✓	sample, model
BAGS [56]	2020									✓	sample, model
VideoLT [38]	2021	✓									sample
LOCE [33]	2021	✓	✓								sample, objective
DARS [26]	2021	✓	✓		✓						sample, objective
CReST [106]	2021	✓			✓						sample
GIST [107]	2021	✓			✓			✓			classifier
FASA [58]	2021	✓				✓					feature
Equalization loss v2 [108]	2021		✓								objective
Seesaw loss [109]	2021		✓								objective
ACSL [110]	2021		✓								objective
IB [111]	2021		✓								objective
PML [51]	2021		✓								objective
VS [112]	2021		✓								objective
LADE [31]	2021		✓	✓							objective, prediction
RoBal [113]	2021		✓	✓				✓			objective, prediction
DisAlign [29]	2021		✓	✓					✓		objective, classifier
MiSLAS [114]	2021		✓			✓			✓		objective, feature, classifier
Logit adjustment [14]	2021			✓							prediction
Conceptual 12M [115]	2021				✓						
DiVE [116]	2021				✓						
MosaicOS [117]	2021				✓						
RSG [118]	2021				✓	✓					feature
SSD [119]	2021				✓				✓		
RIDE [17]	2021				✓					✓	model
MetaSAug [120]	2021					✓					sample
PaCo [121]	2021						✓				feature
DRO-LT [122]	2021						✓				feature
Unsupervised discovery [35]	2021						✓				feature
Hybrid [123]	2021						✓				feature
KCL [13]	2021						✓		✓		feature
DT2 [61]	2021								✓		feature, classifier
LTML [46]	2021									✓	sample, model
ACE [124]	2021									✓	sample, model
ResLT [125]	2021									✓	sample, model
SADE [30]	2021									✓	objective, model

[1] Zhang et al. "Deep Long-Tailed Learning: A Survey" TPAMI 2023.

Supervised Long-tailed Learning

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.

$$E_n = (1-\beta^n)/(1-\beta)$$

$$\text{where } \beta = (N-1)/N$$

$$\lim_{\beta \to 1} E_n = n$$

- **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.

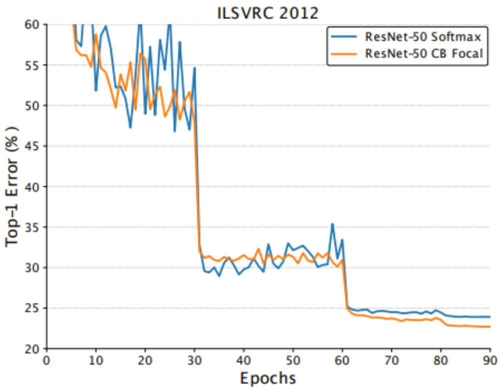
[1] Cui et al. "Class-Balanced Loss Based on Effective Number of Samples" CVPR 2019.

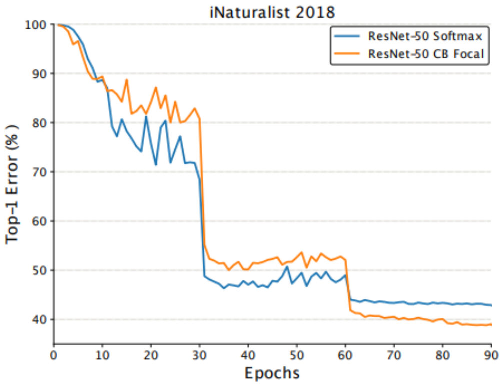
Supervised Long-tailed Learning

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

$$\mathrm{CB}(\mathbf{p}, y) = \frac{1}{E_{n_y}} \mathcal{L}(\mathbf{p}, y) = \frac{1-\beta}{1-\beta^{n_y}} \mathcal{L}(\mathbf{p}, y) \qquad \mathrm{CB}_{\mathrm{softmax}}(\mathbf{z}, y) = -\frac{1-\beta}{1-\beta^{n_y}} \log\left(\frac{\exp(z_y)}{\sum_{j=1}^{C} \exp(z_j)}\right)$$

It can also be combined with **sigmoid cross-entropy loss, focal loss**, etc.





Dataset Name	Long-Tailed CIFAR-10						Long-Tailed CIFAR-100					
Imbalance	200	100	50	20	10	1	200	100	50	20	10	1
Softmax	**34.32**	29.64	25.19	17.77	13.61	6.61	65.16	61.68	56.15	48.86	44.29	29.07
Sigmoid	34.51	**29.55**	23.84	**16.40**	**12.97**	**6.36**	64.39	**61.22**	55.85	48.57	44.73	**28.39**
Focal ($\gamma = 0.5$)	36.00	29.77	**23.28**	17.11	13.19	6.75	65.00	61.31	55.88	48.90	44.30	28.55
Focal ($\gamma = 1.0$)	34.71	29.62	23.29	17.24	13.34	6.60	**64.38**	61.59	**55.68**	**48.05**	**44.22**	28.85
Focal ($\gamma = 2.0$)	35.12	30.41	23.48	16.77	13.68	6.61	65.25	61.61	56.30	48.98	45.00	28.52
Class-Balanced	**31.11**	**25.43**	**20.73**	**15.64**	**12.51**	**6.36***	**63.77**	**60.40**	**54.68**	**47.41**	**42.01**	**28.39***
Loss Type	SM	Focal	Focal	SM	SGM	SGM	Focal	Focal	SGM	Focal	Focal	SGM
β	0.9999	0.9999	0.9999	0.9999	0.9999	-	0.9	0.9	0.99	0.99	0.999	-
γ	-	1.0	2.0	-	-	-	1.0	1.0	-	0.5	0.5	-

The proposed framework provides a non-parametric means of quantifying data overlap.

[1] Cui et al. "Class-Balanced Loss Based on Effective Number of Samples" CVPR 2019.

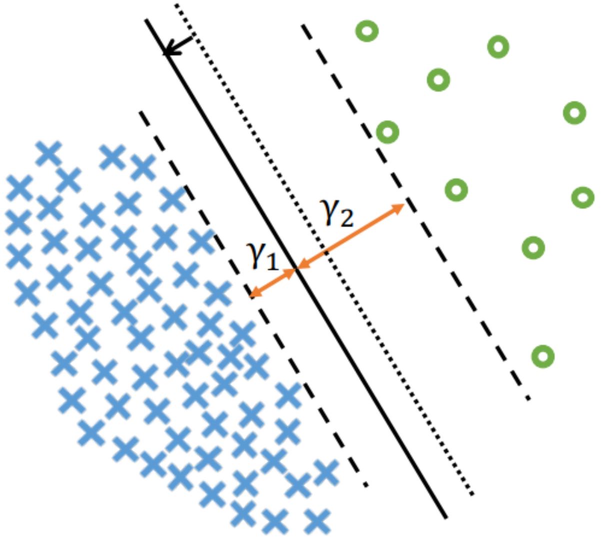
Supervised Long-tailed Learning

Class-wise margin calibration

- **Motivation:** for imbalanced learning, there is a class-distribution-aware margin trade-off for generalization error.

The generalization error is proportional to the following (two classes, and same holds for multiple classes)

$$\frac{1}{\gamma_1\sqrt{n_1}} + \frac{1}{\gamma_2\sqrt{n_2}} \qquad \gamma_1 + \gamma_2 = \gamma$$



The margin definition:

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

[1] Cao et al. "Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss" NeurIPS 2019.

Supervised Long-tailed Learning

➢ **LDAM:** The authors define their hinge loss function (and its relaxed version via Softmax)

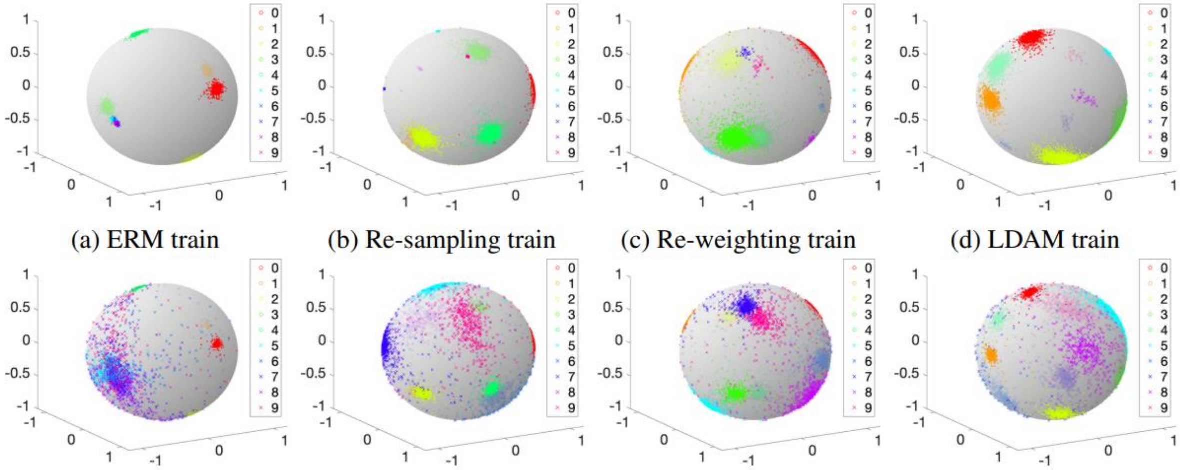
$$\mathcal{L}_{\text{LDAM-HG}}((x,y);f) = \max(\max_{j\neq y}\{z_j\} - z_y + \Delta_y, 0)$$

$$\text{where } \Delta_j = \frac{C}{n_j^{1/4}} \text{ for } j \in \{1,\dots,k\}$$

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

$$\text{where } \Delta_j = \frac{C}{n_j^{1/4}} \text{ for } j \in \{1,\dots,k\}$$

Dataset	Imbalanced CIFAR-10				Imbalanced CIFAR-100			
Imbalance Type	long-tailed		step		long-tailed		step	
Imbalance Ratio	100	10	100	10	100	10	100	10
ERM	29.64	13.61	36.70	17.50	61.68	44.30	61.45	45.37
Focal [Lin et al., 2017]	29.62	13.34	36.09	16.36	61.59	44.22	61.43	46.54
LDAM	26.65	13.04	33.42	15.00	60.40	43.09	60.42	43.73
CB RS	29.45	13.21	38.14	15.41	66.56	44.94	66.23	46.92
CB RW [Cui et al., 2019]	27.63	13.46	38.06	16.20	66.01	42.88	78.69	47.52
CB Focal [Cui et al., 2019]	25.43	12.90	39.73	16.54	63.98	42.01	80.24	49.98
HG-DRS	27.16	14.03	29.93	14.85	-	-	-	-
LDAM-HG-DRS	24.42	12.72	24.53	12.82	-	-	-	-
M-DRW	24.94	13.57	27.67	13.17	59.49	43.78	58.91	44.72
LDAM-DRW	**22.97**	**11.84**	**23.08**	**12.19**	**57.96**	**41.29**	**54.64**	**40.54**


(a) ERM train (b) Re-sampling train (c) Re-weighting train (d) LDAM train
(e) ERM val (f) Re-sampling val (g) Re-weighting val (h) LDAM val

The margin definition is an approximation to the truth value, and whether we should directly add on the logit space?

[1] Cao et al. "Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss" NeurIPS 2019.

Supervised Long-tailed Learning

Class-wise logit adjustment [1]

- **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:

$$\mathrm{BER}(f) \doteq \frac{1}{L} \sum_{y \in [L]} \mathbb{P}_{x|y}\left(y \notin \operatorname{argmax}_{y' \in \mathcal{Y}} f_{y'}(x)\right)$$

Under Bayes-optimal prediction, if $\mathbb{P}^{\mathrm{bal}}(y \mid x) \propto \mathbb{P}(y \mid x) / \mathbb{P}(y)$

Then

$$\operatorname{argmax}_{y \in [L]} \mathbb{P}^{\mathrm{bal}}(y \mid x) = \operatorname{argmax}_{y \in [L]} \exp(s_y^*(x)) / \mathbb{P}(y) = \operatorname{argmax}_{y \in [L]} s_y^*(x) - \ln \mathbb{P}(y)$$

[1] Menon et al. ”Long-tailed Learning via Logit Adjustment" ICLR 2021.

Supervised Long-tailed Learning

- **The logit adjusted softmax cross-entropy**

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

$$\boxed{w_1^\top\Phi(x)/\pi_1 < w_2^\top\Phi(x)/\pi_2 \overset{\not\Rightarrow}{\not\Leftarrow} \exp(w_1^\top\Phi(x))/\pi_1 < \exp(w_2^\top\Phi(x))/\pi_2.}$$

- **Post-hoc logit adjustment**

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

[1] Menon et al. ”Long-tailed Learning via Logit Adjustment" ICLR 2021.

Supervised Long-tailed Learning

➢ **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*

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

Letting $\delta_y = \pi_y$, we immediately deduce that the logit-adjusted loss of (10) is consistent, *provided* our π_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 Δ in the binary case.

[1] Menon et al. ”Long-tailed Learning via Logit Adjustment" ICLR 2021.

Dynamic adjustment based on a fine-grained generalization bound

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 $f \in \mathcal{F}$:*

$$\mathcal{R}^{L}_{bal}(f) \precsim \Phi(L_{\mathrm{VS}}, \delta) + \frac{\hat{\mathfrak{C}}_{\mathcal{S}}(\mathcal{F})}{C\pi_C} \sum_{y=1}^{C} \alpha_y \tilde{\beta}_y \sqrt{\pi_y}\left[1 - \mathit{softmax}\left(\beta_y B_y(f) + \Delta_y\right)\right].$$

$$L_{\mathrm{VS}}(f(\boldsymbol{x}), y) = -\alpha_y \log\left(\frac{e^{\beta_y f(\boldsymbol{x})_y + \Delta_y}}{\sum_{y'} e^{\beta_{y'} f(\boldsymbol{x})_{y'} + \Delta_{y'}}}\right).$$

Algorithm 1: Principled Learning Algorithm induced by the Theoretical Insights

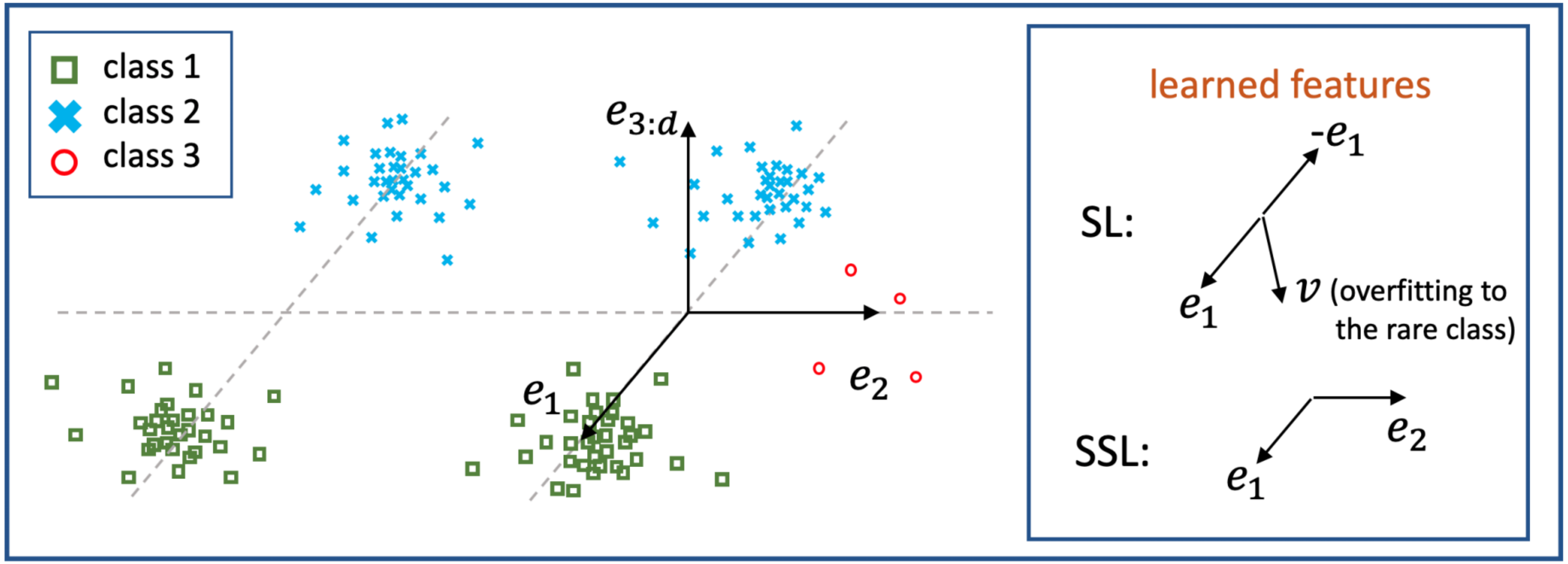
Require: Training set $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{N}$ and a model f parameterized by Θ.

```
Initialize the model parameters Θ randomly.
for t = 1, 2, ··· , T do
    B ← SampleMiniBatch(S, m)                          ▷ A mini-batch of m samples
    if t < T_0 then
        Set α = 1, β_y, Δ_y                            ▷ Adjust logits during the initial phase
    else
        Set α_y ∝ π_y^{-ν}, β_y = 1, Δ_y, ν > 0          ▷ TLA and ADRW
    end if
    L(f, B) ← (1/m) Σ_{(x,y)∈B} L_VS(f(x), y)          ▷ Calculate the loss
    Θ ← Θ − η∇_Θ L(f, B)                               ▷ One SGD step
    Optional: anneal the learning rate η.              ▷ Required when t = T_0
end for
```


[1] Wang et al. "A Unified Generalization Analysis of Re-Weighting and Logit-Adjustment for Imbalanced Learning" NeurIPS 2023.

Self-Supervised Long-tailed Learning

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)

[1] Liu et al. "Self-supervised Learning is More Robust to Dataset Imbalance." ICLR 2021.

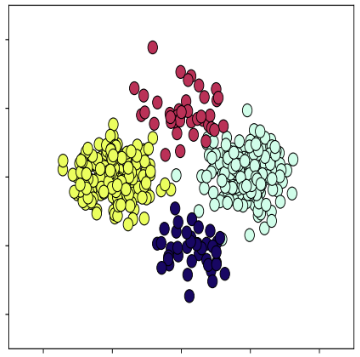
Self-Supervised Long-tailed Learning

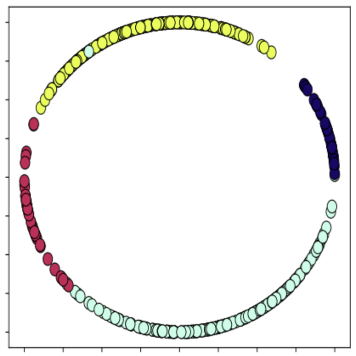
Self-supervised learning still suffers from data imbalance

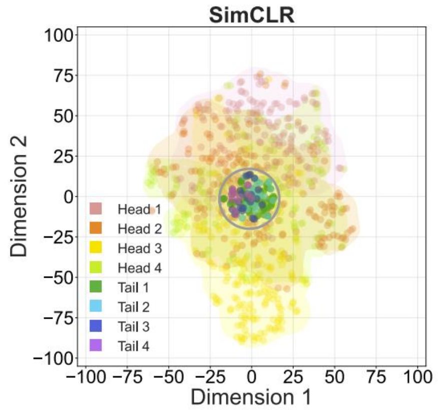
- Performance degeneration: Linear probing on imbalanced data **(D_i)** and balanced data **(D_b)** with same data amount

Dataset	Subset	*Many*	*Medium*	*Few*	All
CIFAR10	D_b	77.14 ± 4.64	74.25 ± 6.54	71.47 ± 7.55	74.57 ± 0.65
	D_i	76.07 ± 3.88	67.97 ± 5.84	54.21 ± 10.24	67.08 ± 2.15
CIFAR100	D_b	25.48 ± 1.74	25.16 ± 3.07	24.01 ± 1.23	24.89 ± 0.99
	D_i	30.72 ± 2.01	21.93 ± 2.61	15.99 ± 1.51	22.96 ± 0.43

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







[1] Jiang et al. "Self-damaging contrastive learning." ICML 2021.

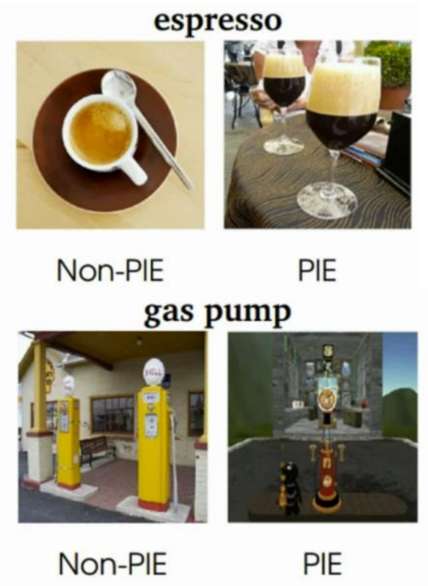
[2] Zhou et al. "Combating Representation Learning Disparity with Geometric Harmonization." NeurIPS 2023.

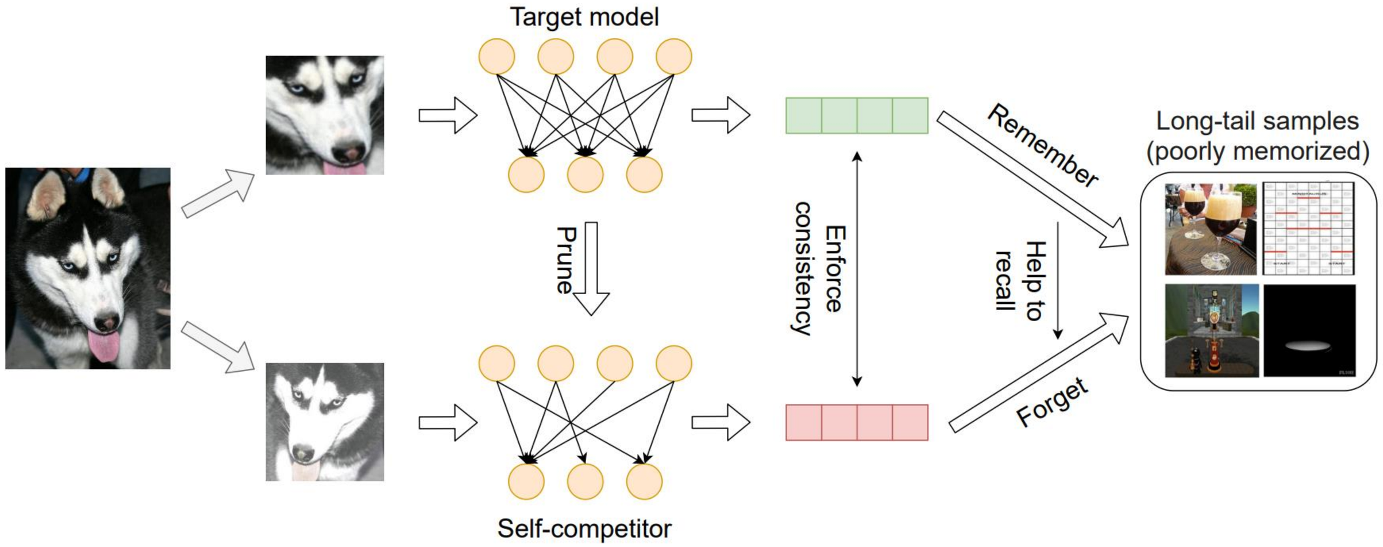
Model-based Self-Supervised Long-tailed Learning

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.



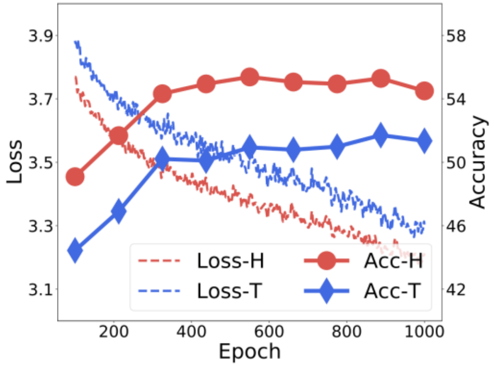


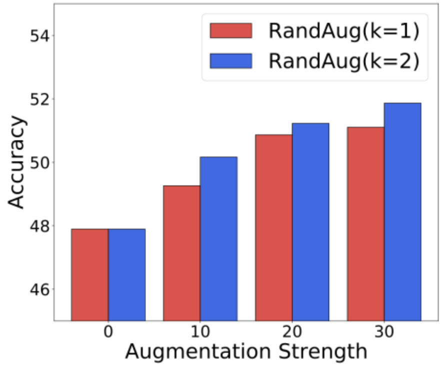
[1] Jiang et al. "Self-damaging contrastive learning." ICML 2021.

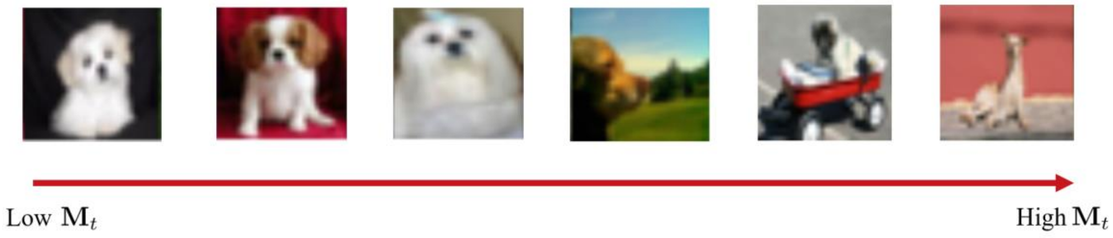
BCL: Boosted Contrastive Learning

- 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*.

$$\mathcal{L}^m_{i,0} = \mathcal{L}_{i,0}, \quad \mathcal{L}^m_{i,t} = \beta\mathcal{L}^m_{i,t-1} + (1-\beta)\mathcal{L}_{i,t} \qquad \mathbf{M}_{i,t} = \frac{1}{2}\left(\frac{\mathcal{L}^m_{i,t} - \bar{\mathcal{L}}^m_t}{\max\left\{\left|\mathcal{L}^m_{i,t} - \bar{\mathcal{L}}^m_t\right|\right\}_{i=0,\ldots,N}} + 1\right)$$

$$\Psi(x_i; \mathcal{A}, \mathbf{M}_i) = a_1(x_i) \circ \ldots \circ a_k(x_i),$$

$$a_j(x_i) = \begin{cases} A_j(x_i; \mathbf{M}_i\zeta) & u \sim \mathcal{U}(0,1)\ \&\ u < \mathbf{M}_i \\ x_i & \text{otherwise} \end{cases} \qquad \mathcal{L}_{\text{BCL}} = \frac{1}{N}\sum_{i=1}^{N} -\log \frac{\exp\left(\frac{f(\Psi(x_i))^\top f(\Psi(x_i^+))}{\tau}\right)}{\sum_{x_i' \in X'} \exp\left(\frac{f(\Psi(x_i))^\top f(\Psi(x_i'))}{\tau}\right)}$$

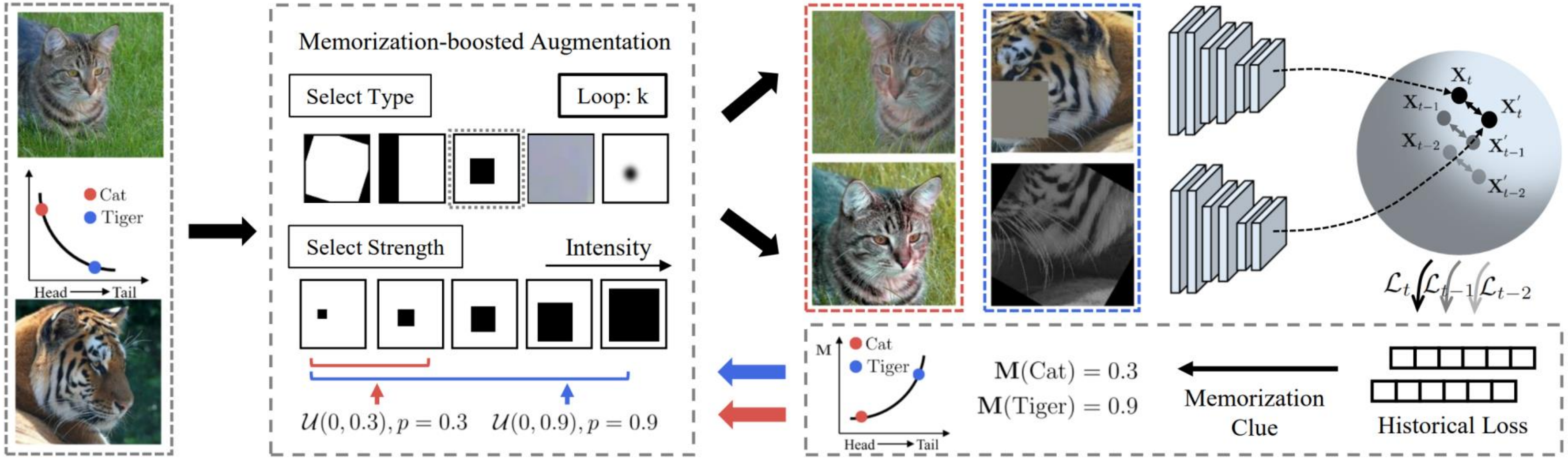
Adaptively assigns the appropriate augmentation strength for the individual sample according to the feedback from the memorization clues

[1] Jiang et al. "Characterizing structural regularities of labeled data in overparameterized models." ICML 2021

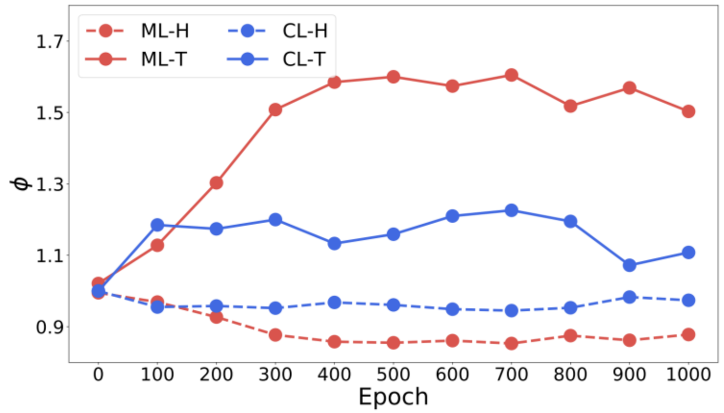
[2] Zhou et al. "Contrastive learning with boosted memorization. " ICML 2022.

Data-based Self-Supervised Long-tailed Learning

BCL: Boosted Contrastive Learning



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



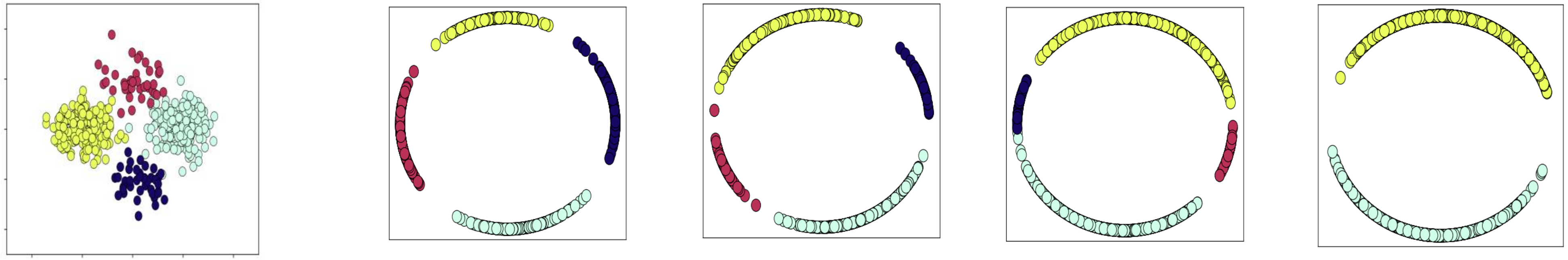
[1] Zhou et al. “Contrastive learning with boosted memorization. " ICML 2022.

Loss-based Self-Supervised Long-tailed Learning

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.



(a) R=1(Balanced) (b) R=4 (c) R=16 (d) R=64

Contrastive learning causes severer representation learning disparity when enlarging the imbalance ratios.

Geometric Uniform Structure

$$\mathbf{M}_i^\top \cdot \mathbf{M}_j = C, \;\; \forall i, j \in \{1, 2, \dots, K\}, \; i \neq j,$$

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

Surrogate Label Allocation

$$\min_{\hat{\mathbf{Q}}=[\hat{\boldsymbol{q}}_1,\dots,\hat{\boldsymbol{q}}_N]} \mathcal{L}_{\mathrm{GH}} = -\frac{1}{|\mathcal{D}|}\sum_{\boldsymbol{x}_i\sim\mathcal{D}} \hat{\boldsymbol{q}}_i \log \boldsymbol{q}_i,$$

$$\text{s.t.} \;\; \hat{\mathbf{Q}} \cdot \mathbb{1}_N = N \cdot \boldsymbol{\pi}, \;\; \hat{\mathbf{Q}}^\top \cdot \mathbb{1}_K = \mathbb{1}_N,$$

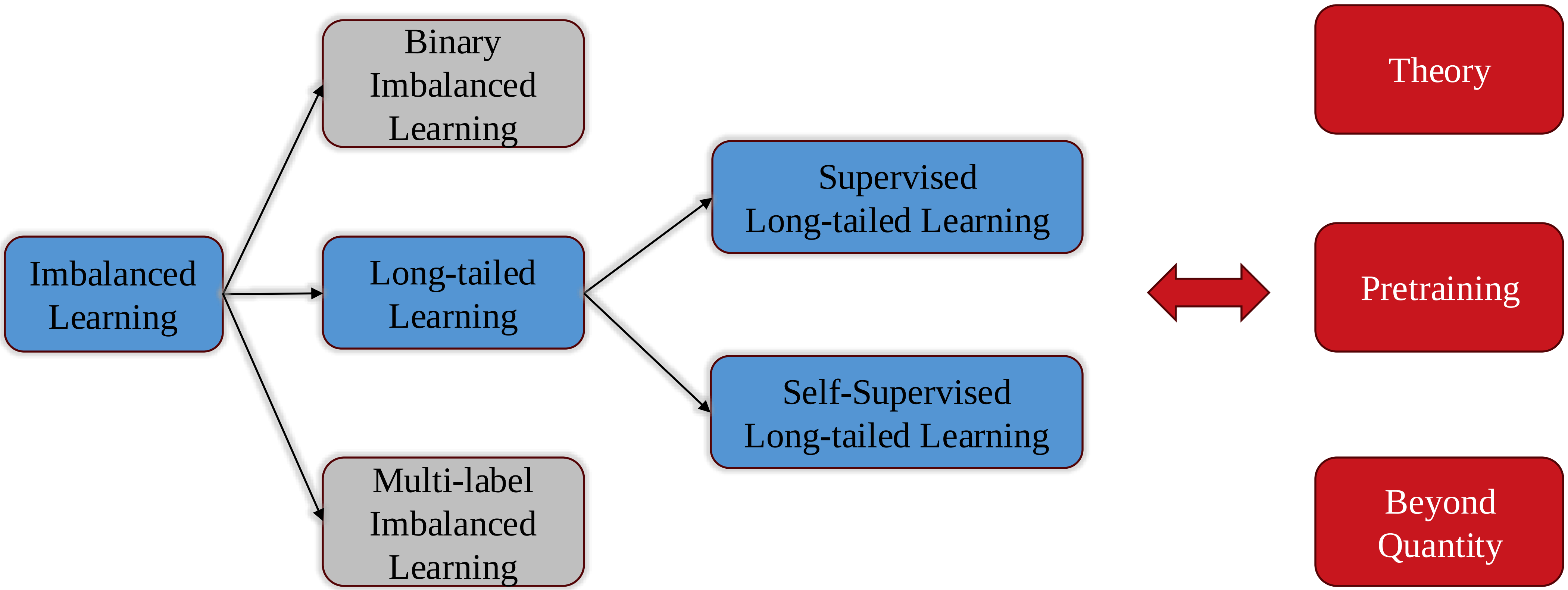
Overall objective

$$\min_{\theta, \hat{\mathbf{Q}}} \mathcal{L} = \mathcal{L}_{\mathrm{InfoNCE}} + w_{\mathrm{GH}}\mathcal{L}_{\mathrm{GH}},$$

[1] Zhou et al. "Combating Representation Learning Disparity with Geometric Harmonization." NeurIPS 2023.

Summary

Still require more efforts on this way



Thank you	Q & A