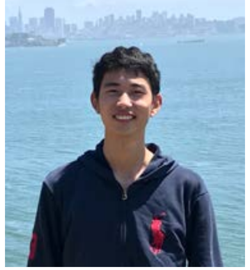
Long-tailed Recognition by Routing Diverse Distribution-Aware Experts

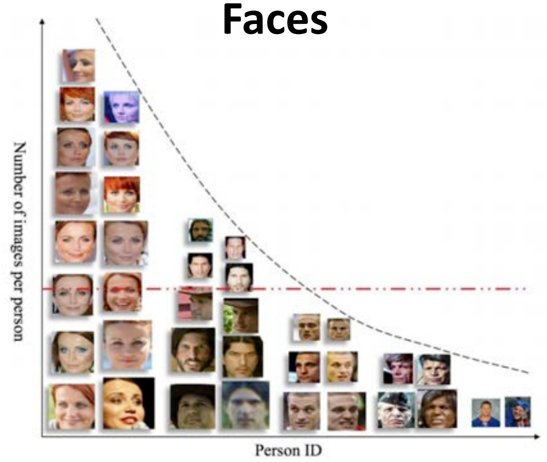
Xudong Wang[1], Long Lian[1], Zhongqi Miao[1], Ziwei Liu[2] and Stella Yu[1]


[1] UC Berkeley / ICSI
[2] Nanyang Technological University

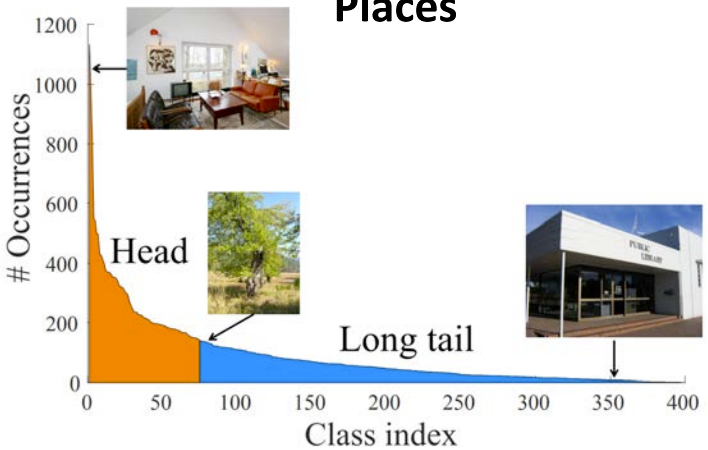
Natural Data Are Often Long-tailed Distributed Over Semantic Classes

Faces



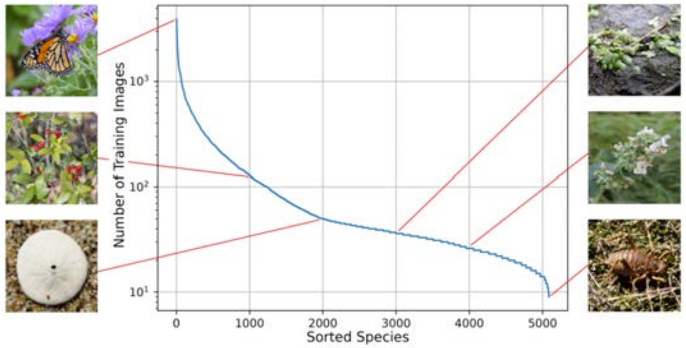
Zhang, Xiao, et al. "Range loss for deep face recognition with long-tailed training data." **[CVPR 2017]**

Places



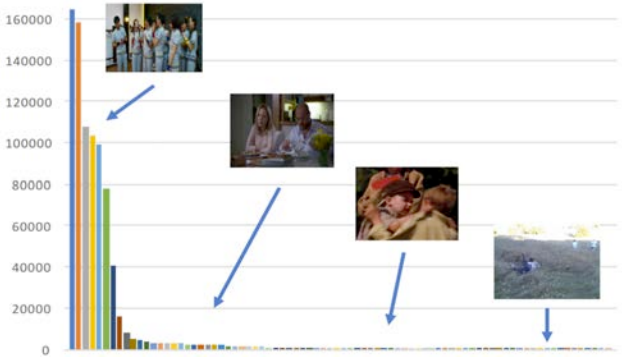
Wang, Yu-Xiong , et al. "Learning to model the tail." **[NeurIPs 2017]**

Species



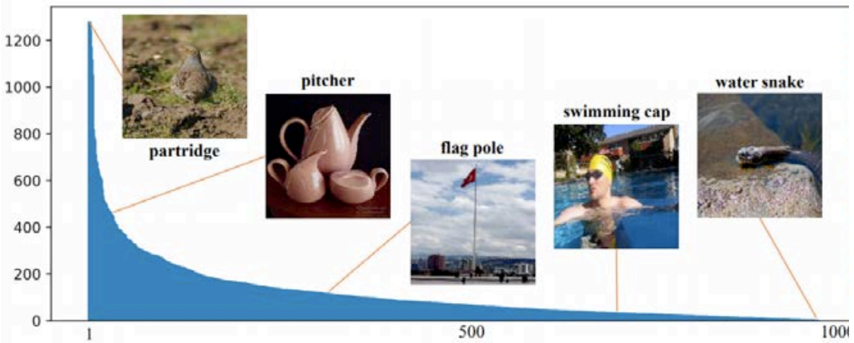
Van Horn, et al. "The inaturalist species classification and detection dataset." **[CVPR 2018]**

Actions



Zhang, Yubo , et al. ”A study on action detection in the wild." *arXiv preprint arXiv:1904.12993* (2019).

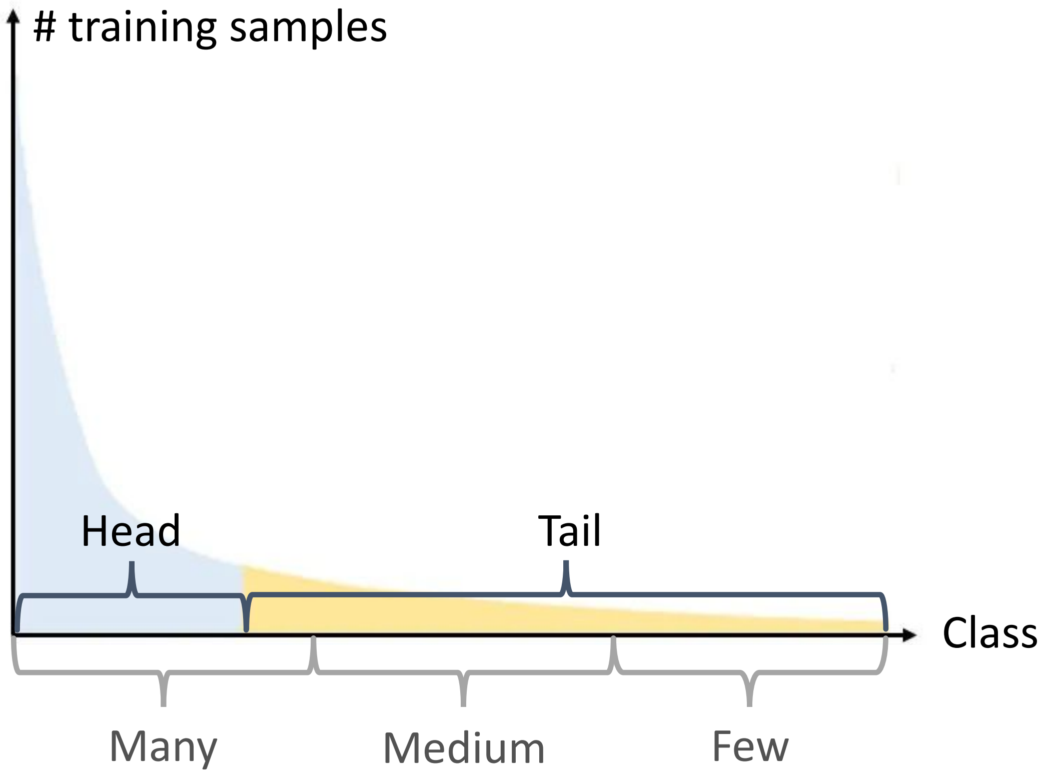
Objects



Liu, Ziwei, et al. "Large-scale long-tailed recognition in an open world." **[CVPR 2019]**

Long-tailed Recognition: Imbalance + Few-shot Learning

- Training set: long-tailed distribution
  - Many-shot: #samples > 100
  - Medium-shot: #samples < 100 & > 20
  - Few-shot: #samples < 20
- Testing set: balanced distribution
- Evaluation:
  - Overall testing set
  - Three splits based on class size



Previous Methods

Methods Overview

1. Instance-wise Balancing (current SOTA)
   - Up/Down sampling tail/head classes. (e.g., Decouple [ICLR 2020], BBN [CVPR 2020])
2. Weighted Loss
   - Assign larger/smaller weights to tail/head classes. (e.g., LDAM [NeurIPS 2019], CB-Loss [CVPR 2019])
3. Feature Enhancement
   - Use the memory enhanced feature learned from both head and tail classes. (e.g., OLTR [CVPR 2018])

Previous Methods

Methods Overview

1. Instance-wise Balancing (current SOTA)
   - Up/Down sampling tail/head classes. (e.g., Decouple [ICLR 2020], BBN [CVPR 2020])
2. Weighted Loss
   - Assign larger/smaller weights to tail/head classes. (e.g., LDAM [NeurIPS 2019], CB-Loss [CVPR 2019])
3. Feature Enhancement
   - Use the memory enhanced feature learned from both head and tail classes. (e.g., OLTR [CVPR 2018])

Caveats

- All these methods generally gain accuracy on tail classes at the cost of performance loss on head classes.

Previous Methods

Methods Overview

1. Instance-wise Balancing (current SOTA)
   - Up/Down sampling tail/head classes. (e.g., Decouple [ICLR 2020], BBN [CVPR 2020])
2. Weighted Loss
   - Assign larger/smaller weights to tail/head classes. (e.g., LDAM [NeurIPS 2019], CB-Loss [CVPR 2019])
3. Feature Enhancement
   - Use the memory enhanced feature learned from both head and tail classes. (e.g., OLTR [CVPR 2018])

Caveats

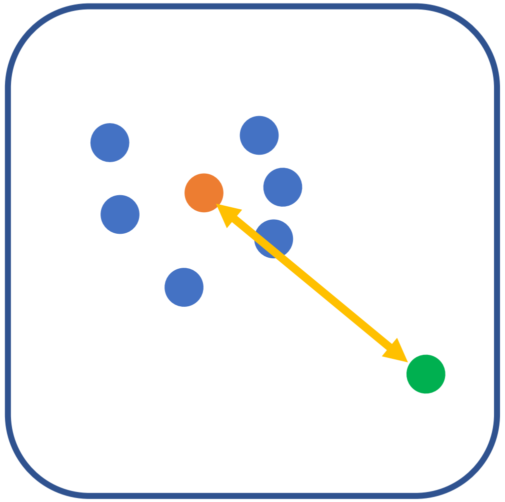
- All these methods generally gain accuracy on tail classes at the cost of performance loss on head classes.

In order to understand the cause of caveats, we decoupled the model error with bias-variance decomposition.

Bias-variance Decomposition with Respect to the Variation in Dataset D

$$\text{Error}(x;h) = \text{Bias}(h)^2 + \text{Variance}(h) + \text{irreducible error}.$$

Bias



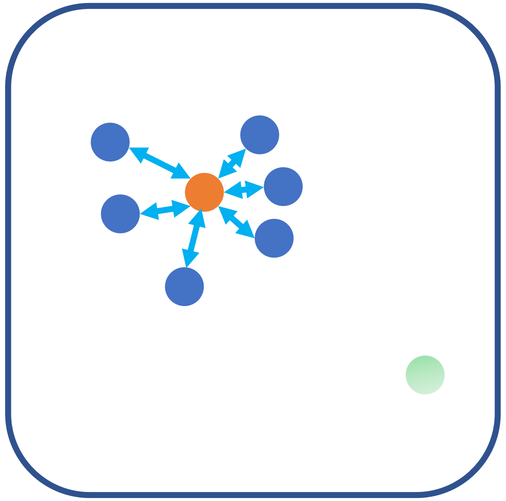
Bias $\mathrm{E}[\mathrm{E}[\widehat{y_d}] - y]$

Ground truth y

Prediction $\widehat{y_d}$

Mean $\mathrm{E}[\widehat{y_d}]$

Variance



Variance $\mathrm{E}[(\mathrm{E}[\widehat{y_d}] - y_d)^2]$

Ground truth y (not used)

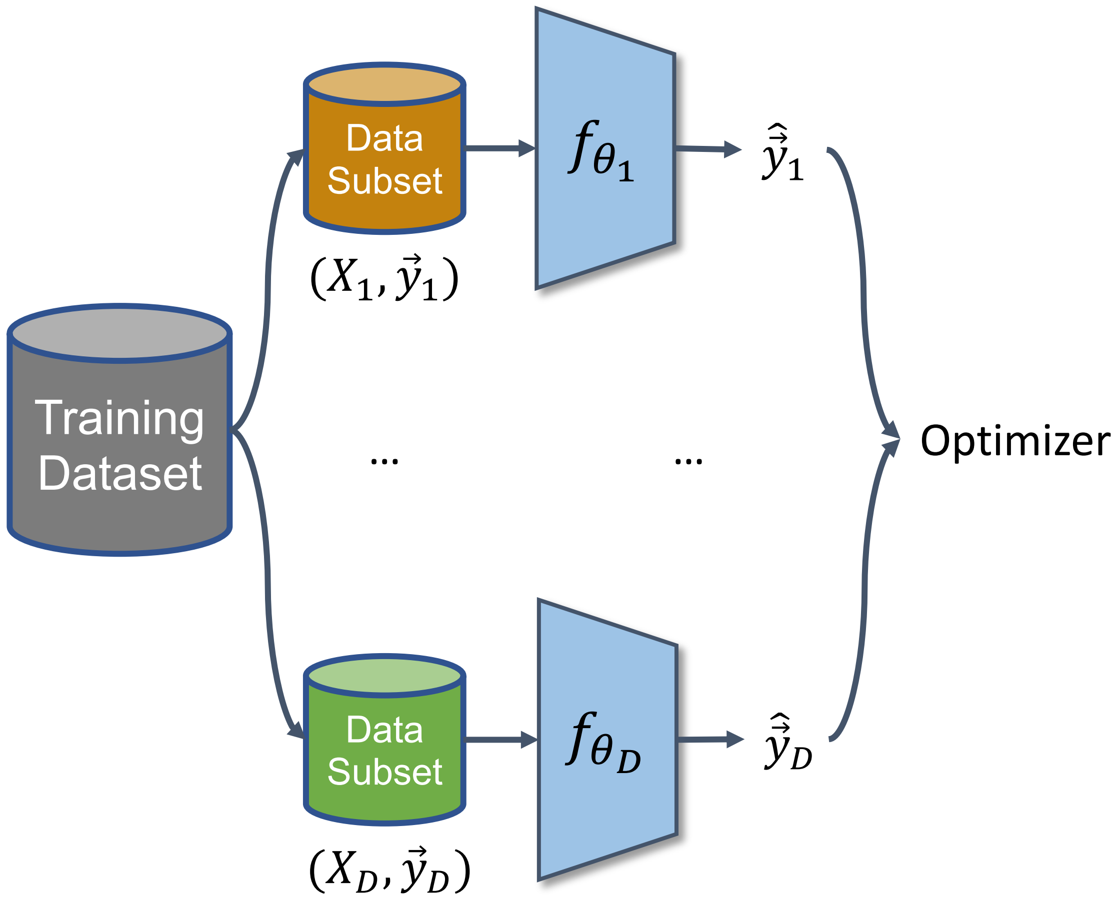
Prediction $\hat{y}$

Mean $\mathrm{E}[\widehat{y_d}]$

Pedro Domingos. A unified bias-variance decomposition. ICML, 2000

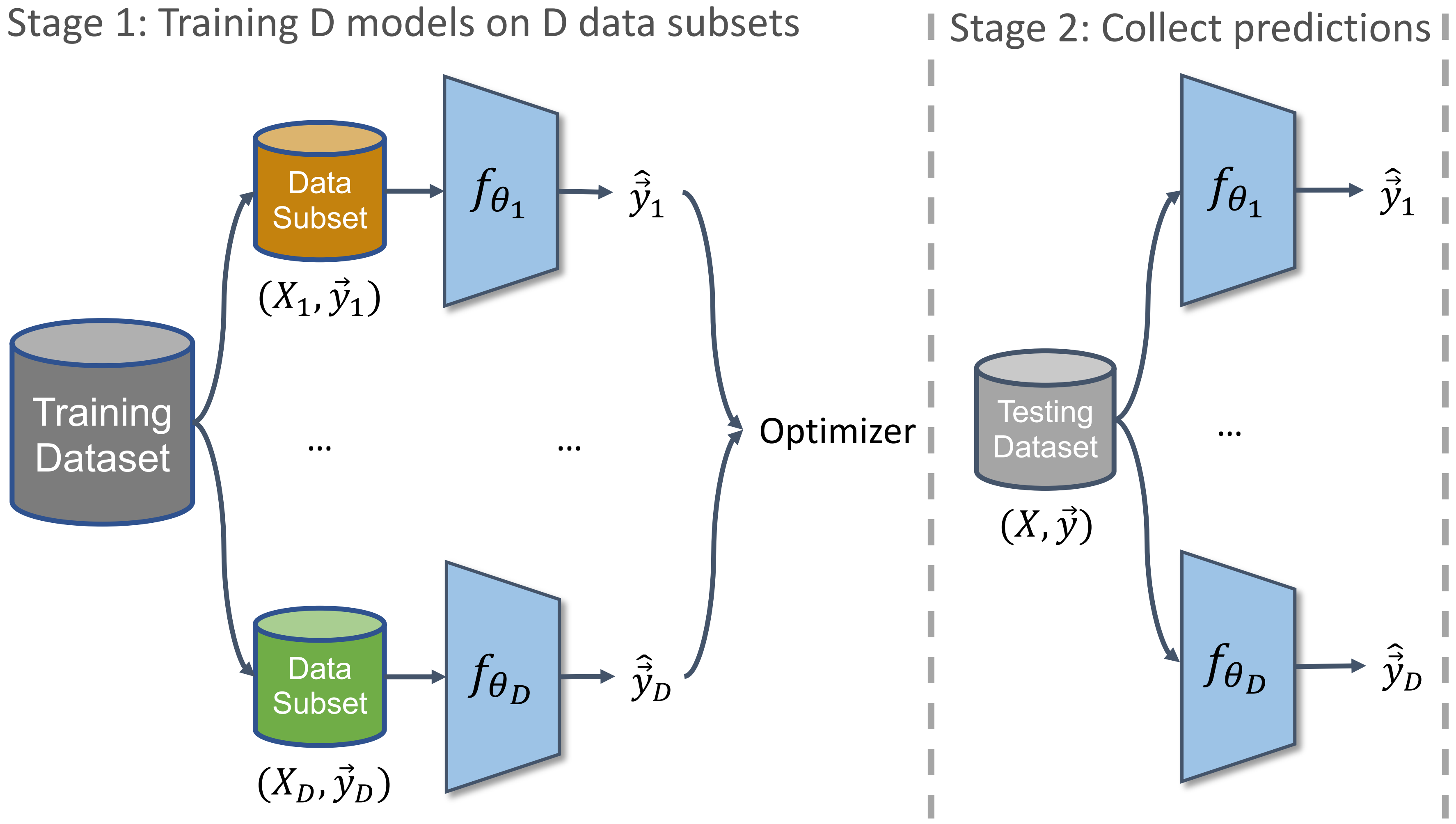
How to Obtain Bias and Variance of Each Method?

Stage 1: Training D models on D data subsets



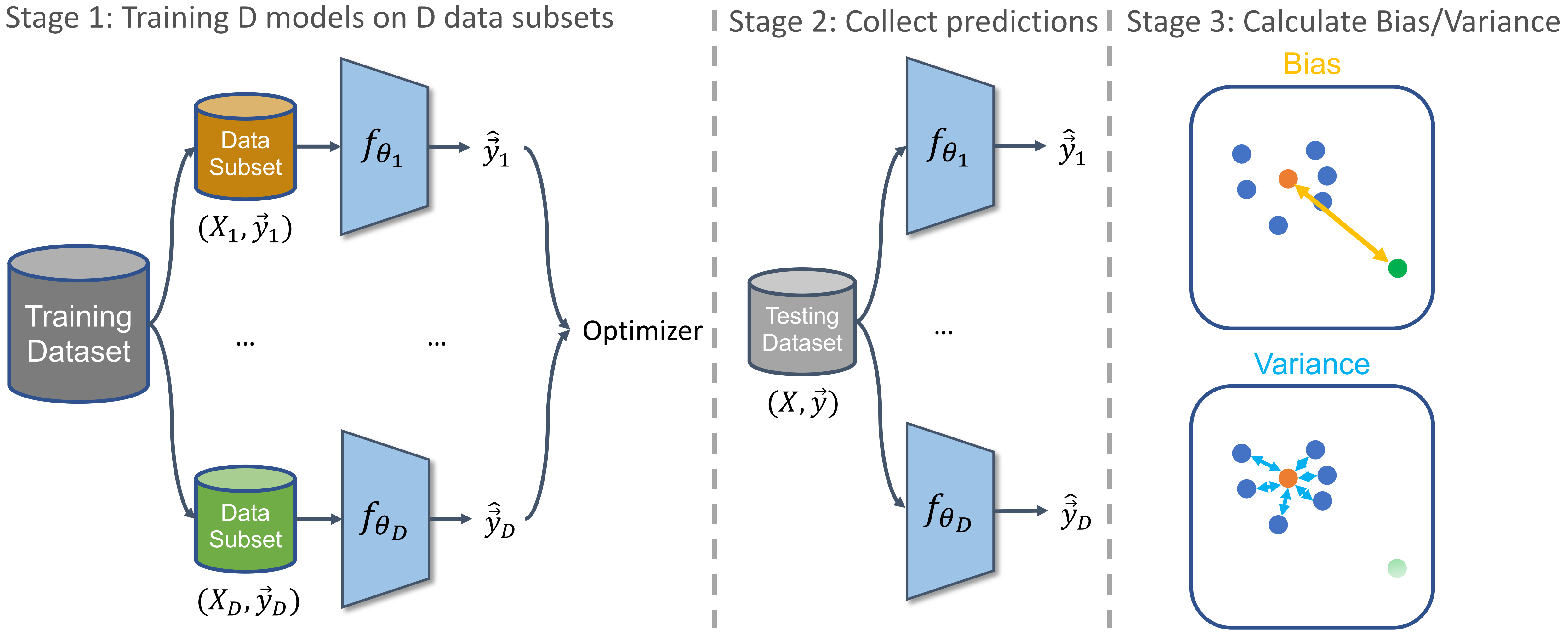
Pedro Domingos. A unified bias-variance decomposition. ICML, 2000

How to Obtain Bias and Variance of Each Method?



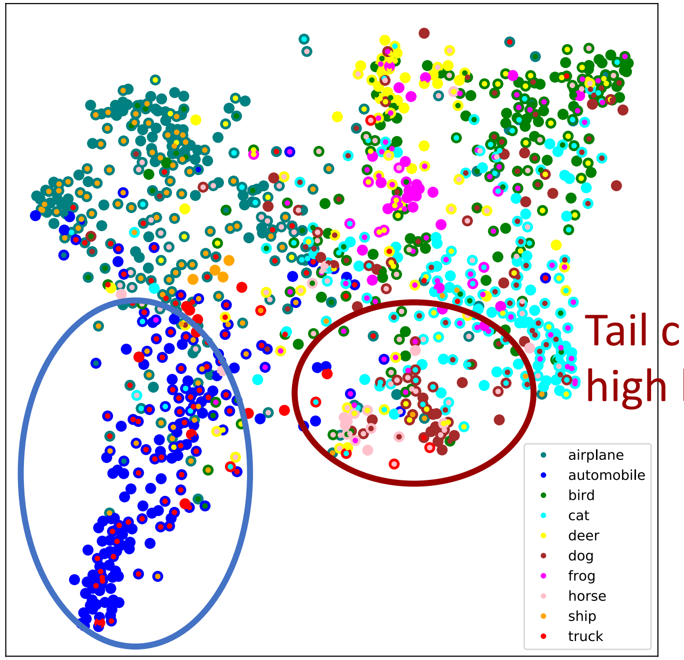
Pedro Domingos. A unified bias-variance decomposition. ICML, 2000

How to Obtain Bias and Variance of Each Method?



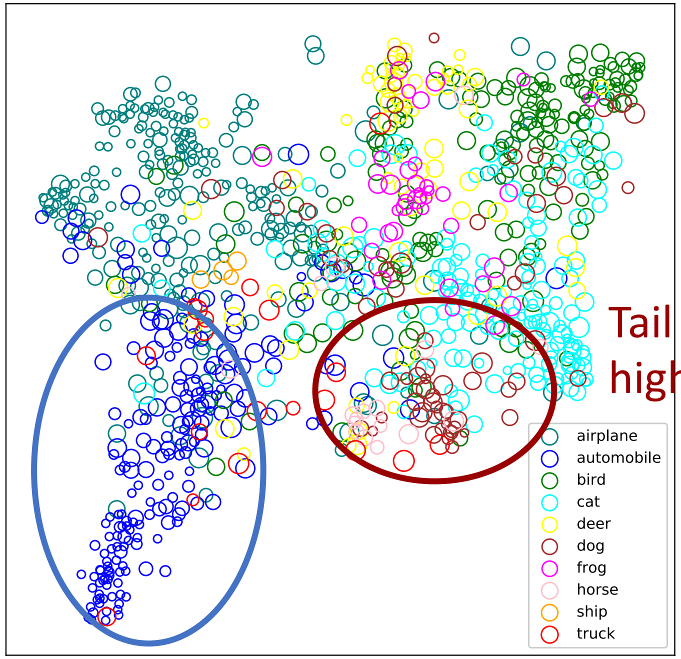
Pedro Domingos. A unified bias-variance decomposition. ICML, 2000

How to Obtain Bias and Variance of Each Method?



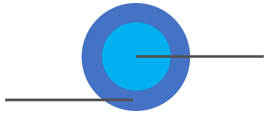
Head class: low bias

Tail class: high bias



Head class: low variance

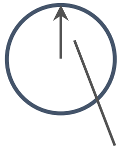
Tail class: high variance



Ground Truth

Prediction

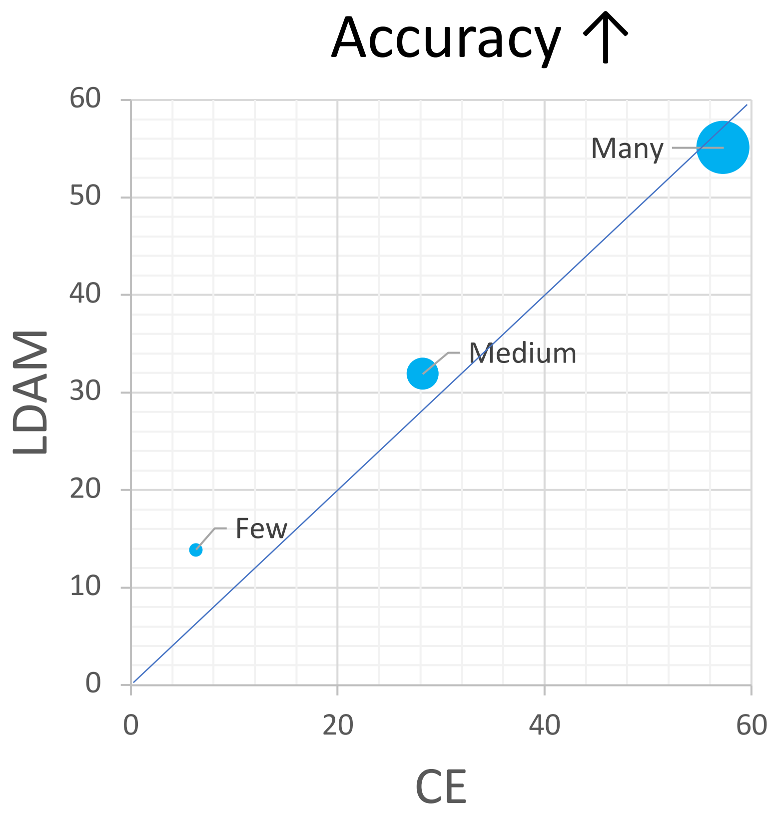
Bias (= 0 if Prediction = GT)



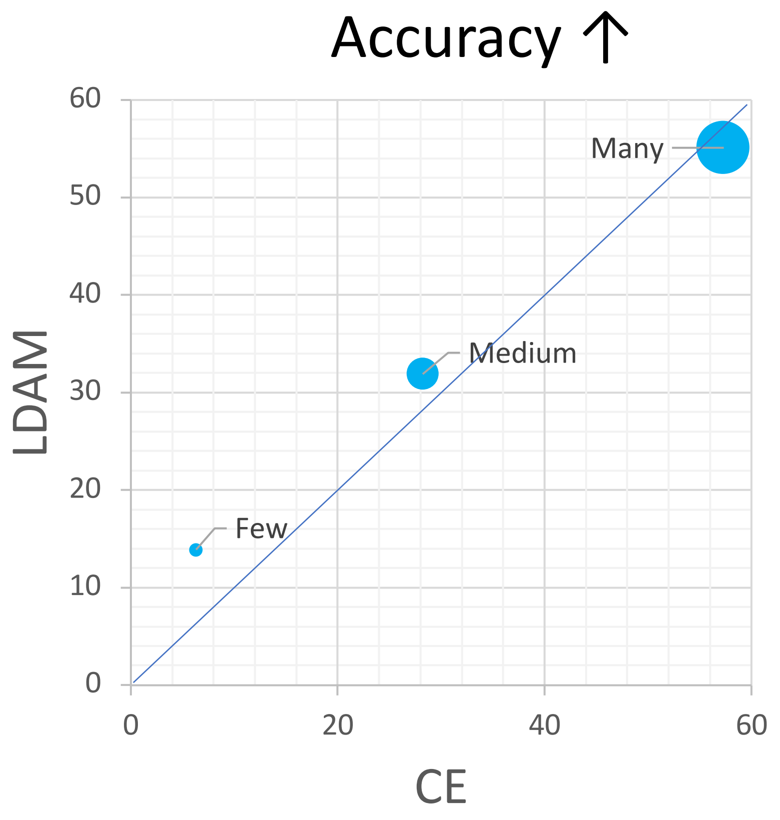
Variance (=radius)

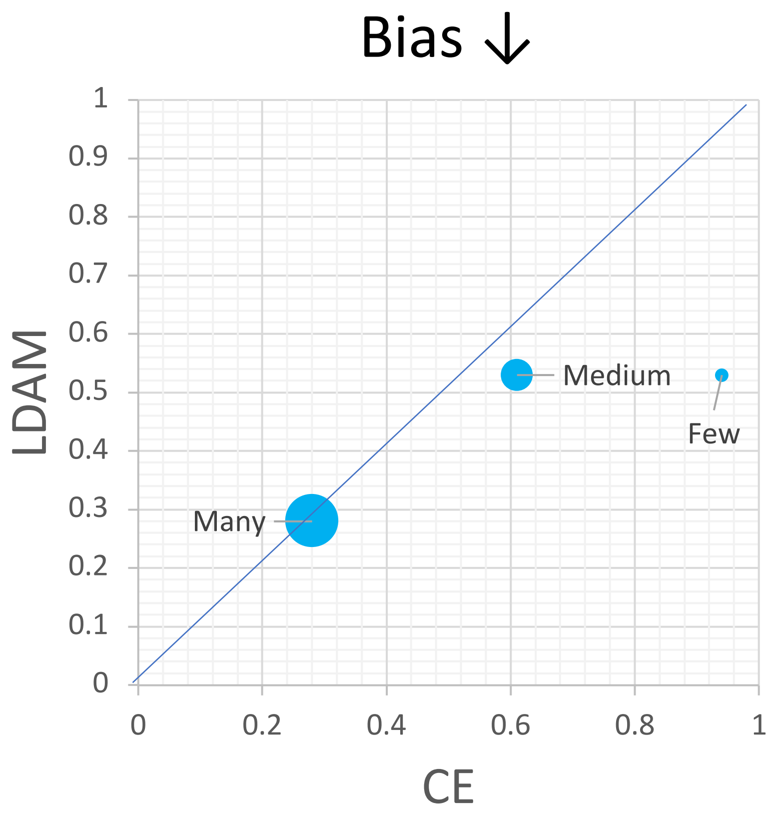
Pedro Domingos. A unified bias-variance decomposition. ICML, 2000

Few-shot Accuracy Gain at The Cost of Many-shot Drop

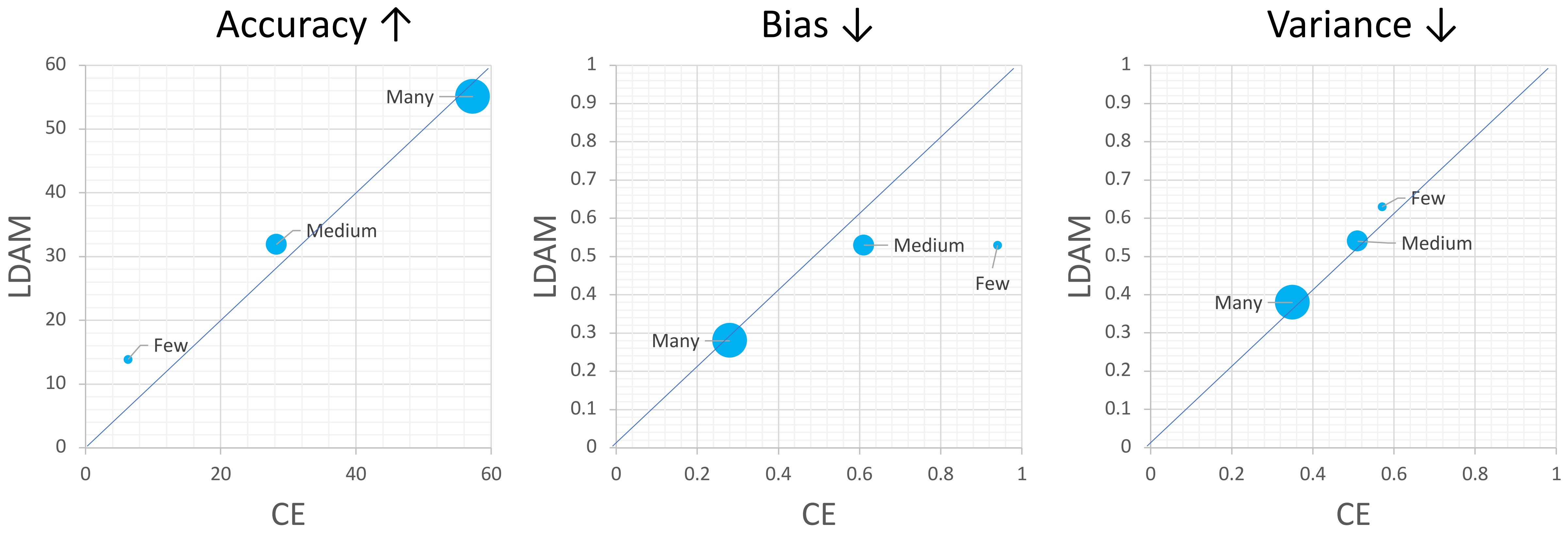


Bias Reduction Tends to Be Greater for Tail Classes





Variance Is Increased Throughout The Class Spectrum



Our Key Insights

	Head Classes			Tail Classes		
	Acc	Bias	Variance	Acc	Bias	Variance
Current SOTAs	Worse	Comparable	Worse	Better	Better	Worse

Why previous methods get worse accuracy on many-shot classes?

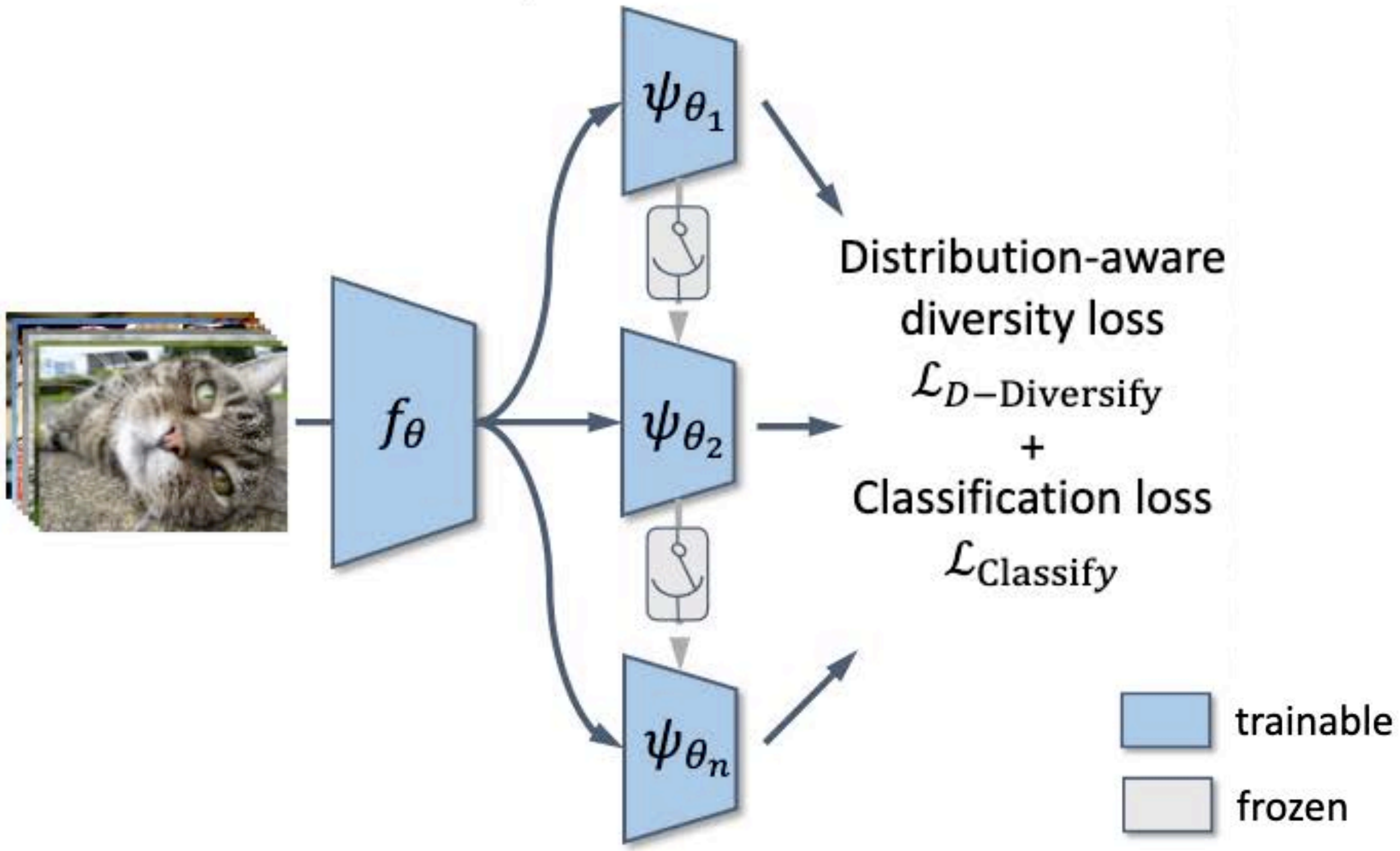
The increased variance leads to a worse bias-variance trade-off.

How to further improve the performance on few-shot classes?

Obtaining the optimal bias-variance trade-off by further reducing variance *and* bias.

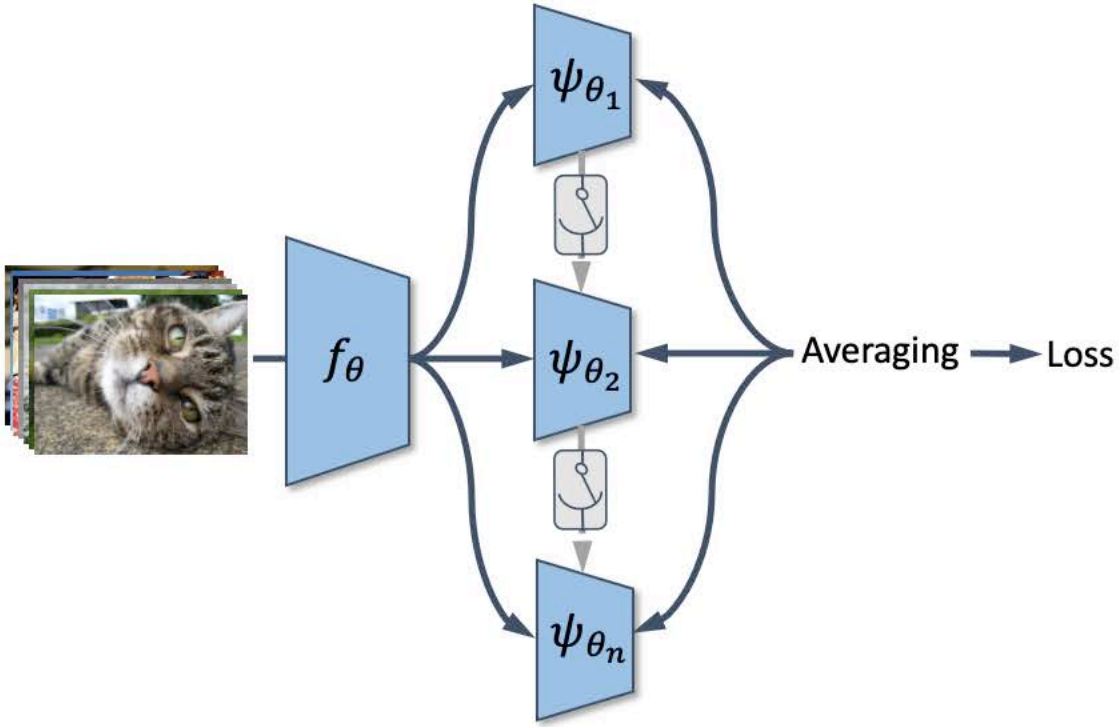
Reducing Model Variance with Multi-expert Framework

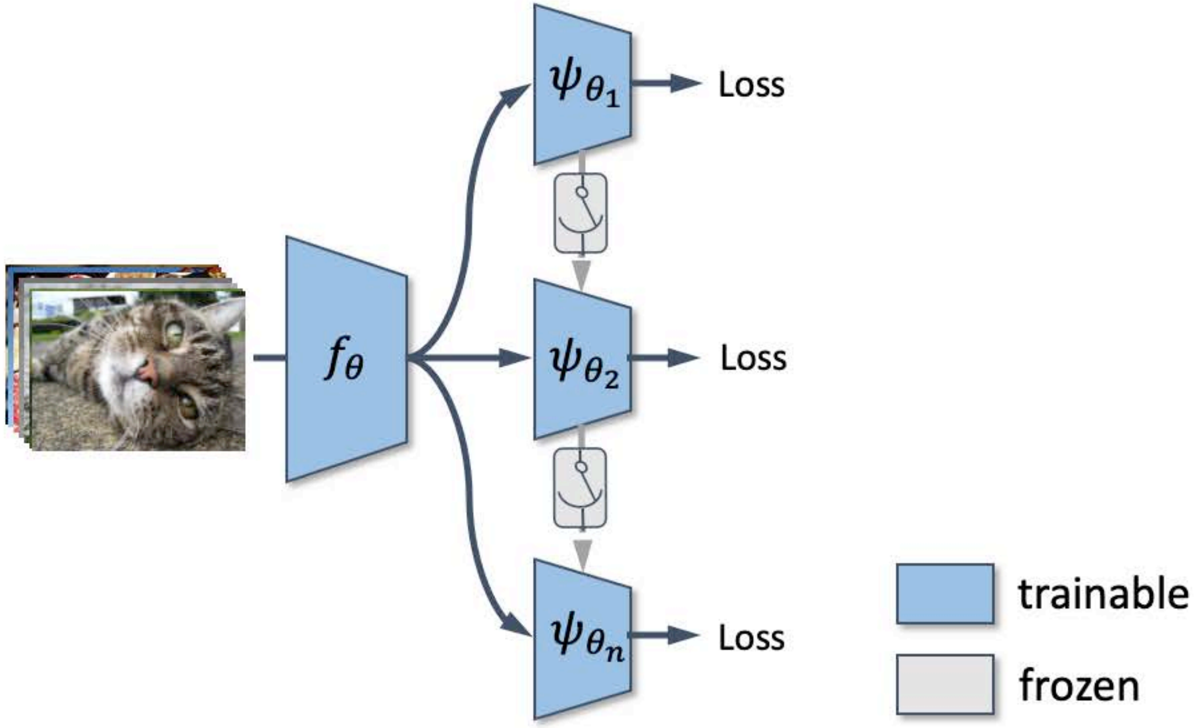
Stage One: Jointly Optimize Diverse Distribution-aware Experts



Reducing Model Bias with Individual Loss

Using Individual Loss Instead of Collaborative Loss





Collaborative loss leads to correlated experts ✗

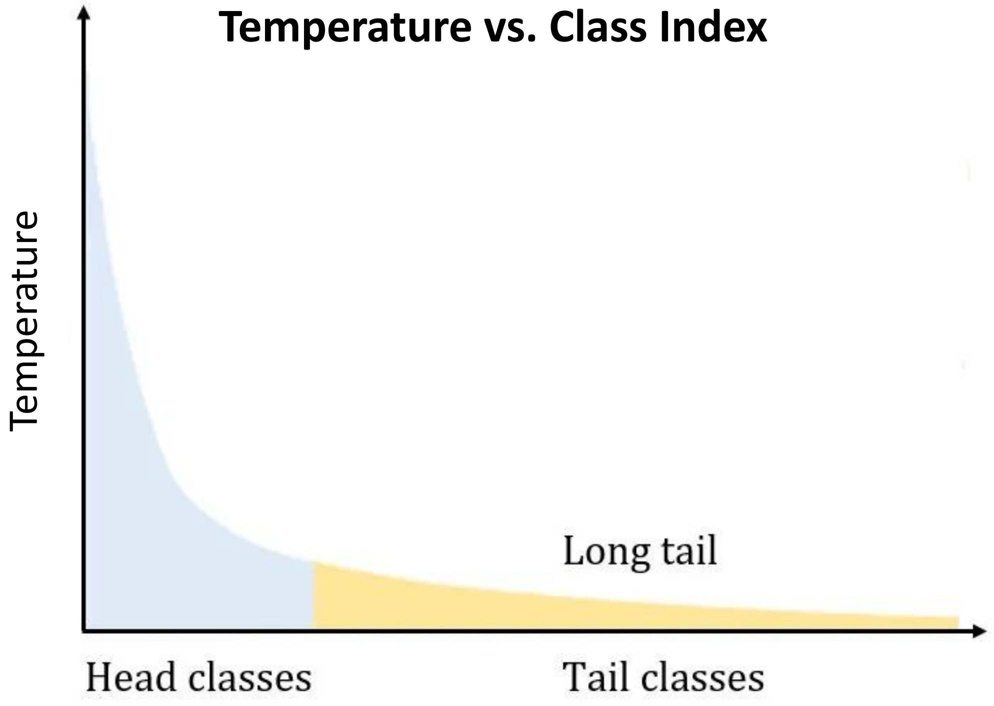
Individual loss decorrelates experts ✓

Further Reducing Model Bias with Distribution-aware Diversity Loss

The distribution-aware diversity loss is proposed to penalize the inter-expert correlation, formulated as:

$$\mathcal{L}^{i}_{\text{D-Diversify}} = -\frac{\lambda}{k-1}\sum_{j\neq i}^{n} \boxed{\mathcal{D}_{KL}}\left(\phi^{i}(\vec{x},\vec{T}), \boxed{\phi^{j}(\vec{x},\vec{T})}\right)$$

KL divergence   Softmax with temperature



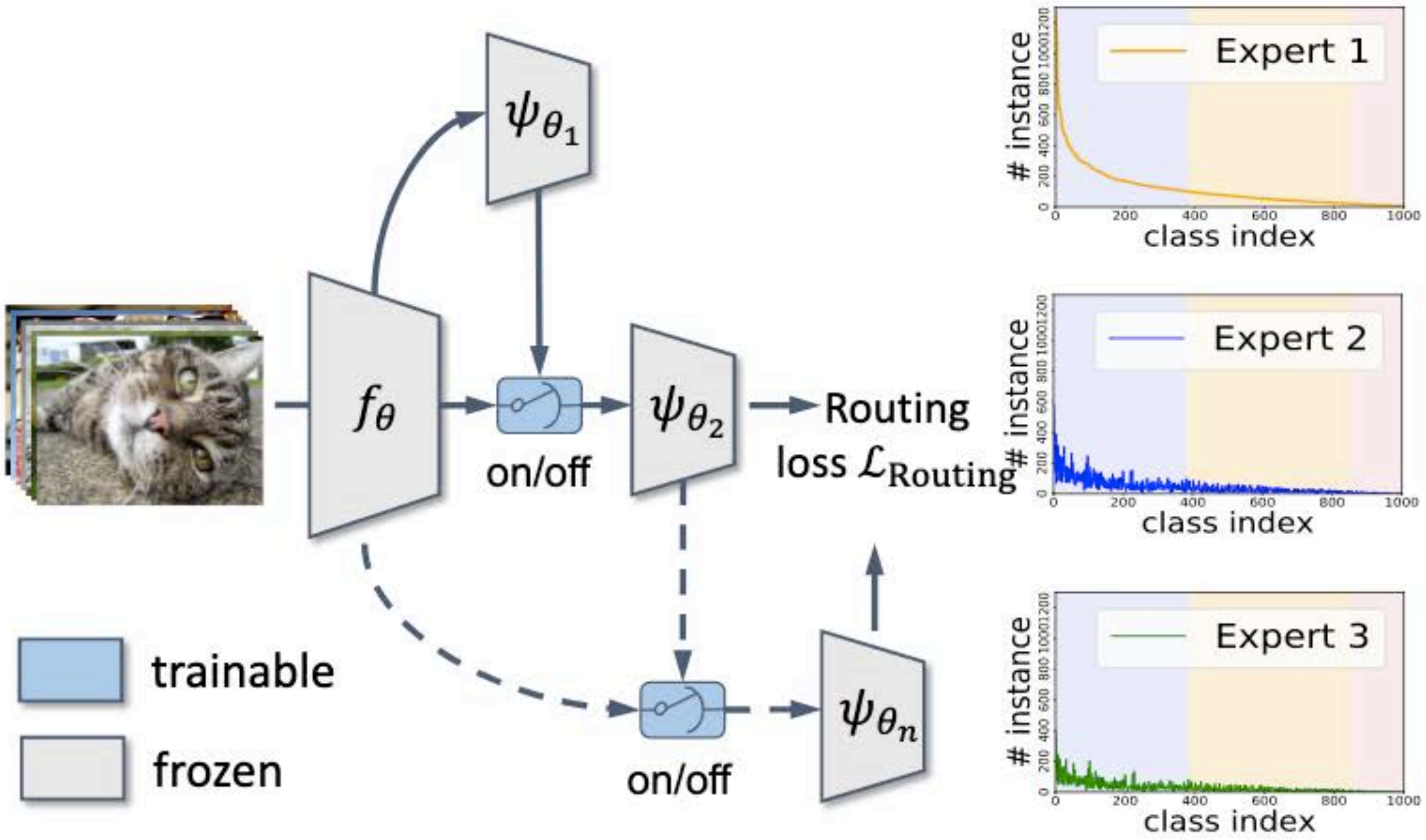
Total Loss for Stage One

$$\mathcal{L}^i_{\text{Total}} = \mathcal{L}^i_{\text{Classify}}(\phi^i(\vec{x}), y) - \frac{\lambda}{n-1}\sum_{j\neq i}^{n} D_{\text{KL}}(\phi^i(\vec{x}, \vec{T}), \phi^j(\vec{x}, \vec{T}))$$

where i is the expert index, $\mathcal{L}^i_{\text{Classify}}(.,.)$ can be LDAM loss, focal loss, etc., depending on the training mechanisms we choose.

Reducing the Computational Complexity with Routing Module

Stage Two: Routing Diverse Experts

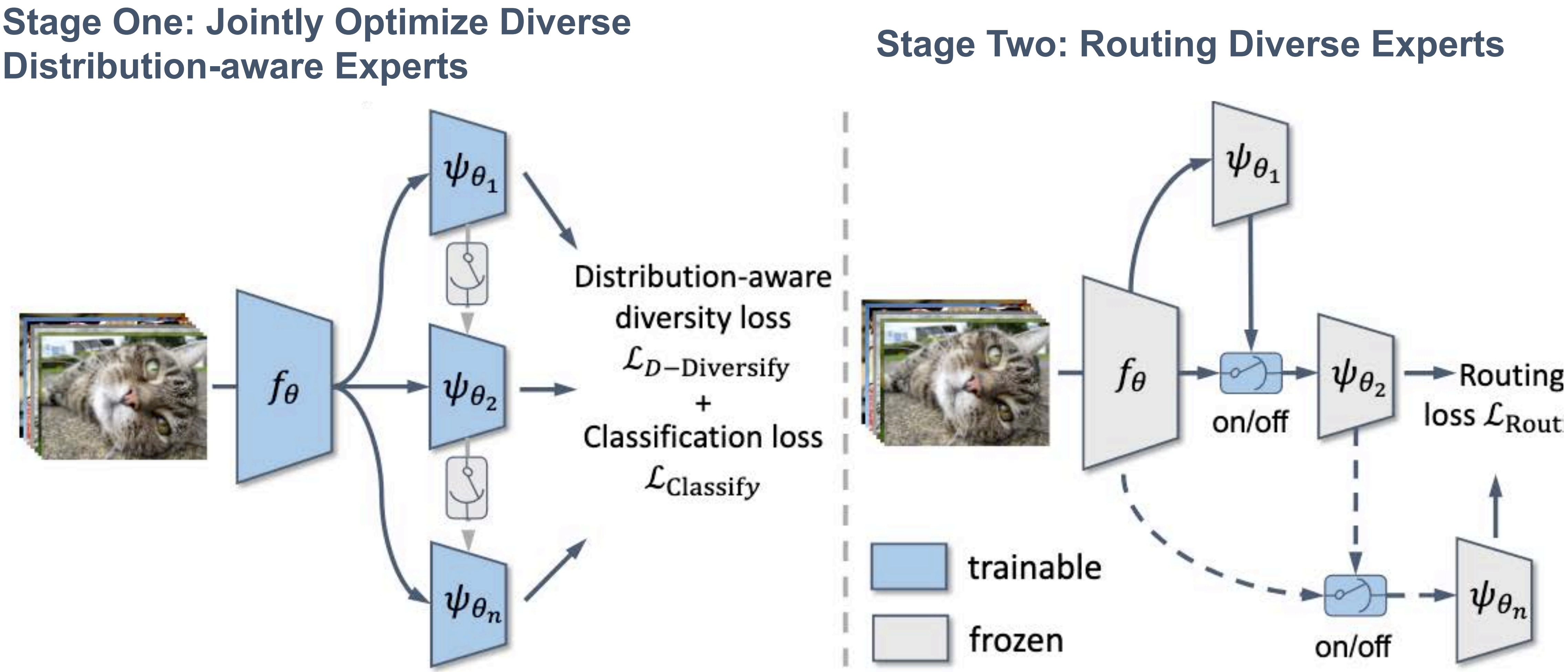


Routing Loss of Expert Assignment

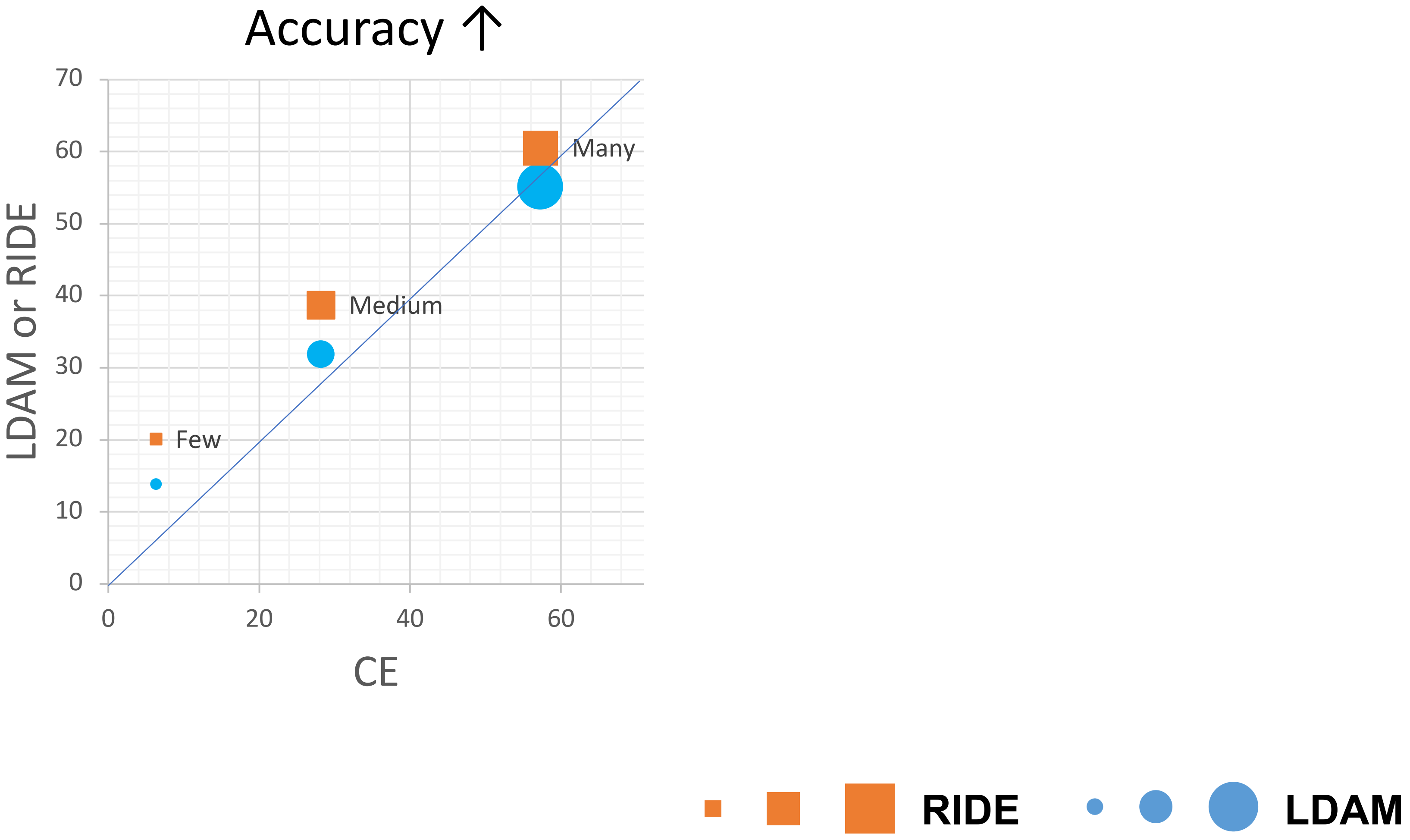
The expert assignment is optimized with the routing loss, a weighted variant of binary cross entropy loss:

$$\mathcal{L}_{\mathrm{Routing}} = -\omega_{\mathrm{p}} y \log(\frac{1}{1+e^{-y_{\mathrm{ea}}}}) - \omega_{\mathrm{n}}(1-y)\log(1-\frac{1}{1+e^{-y_{\mathrm{ea}}}})$$

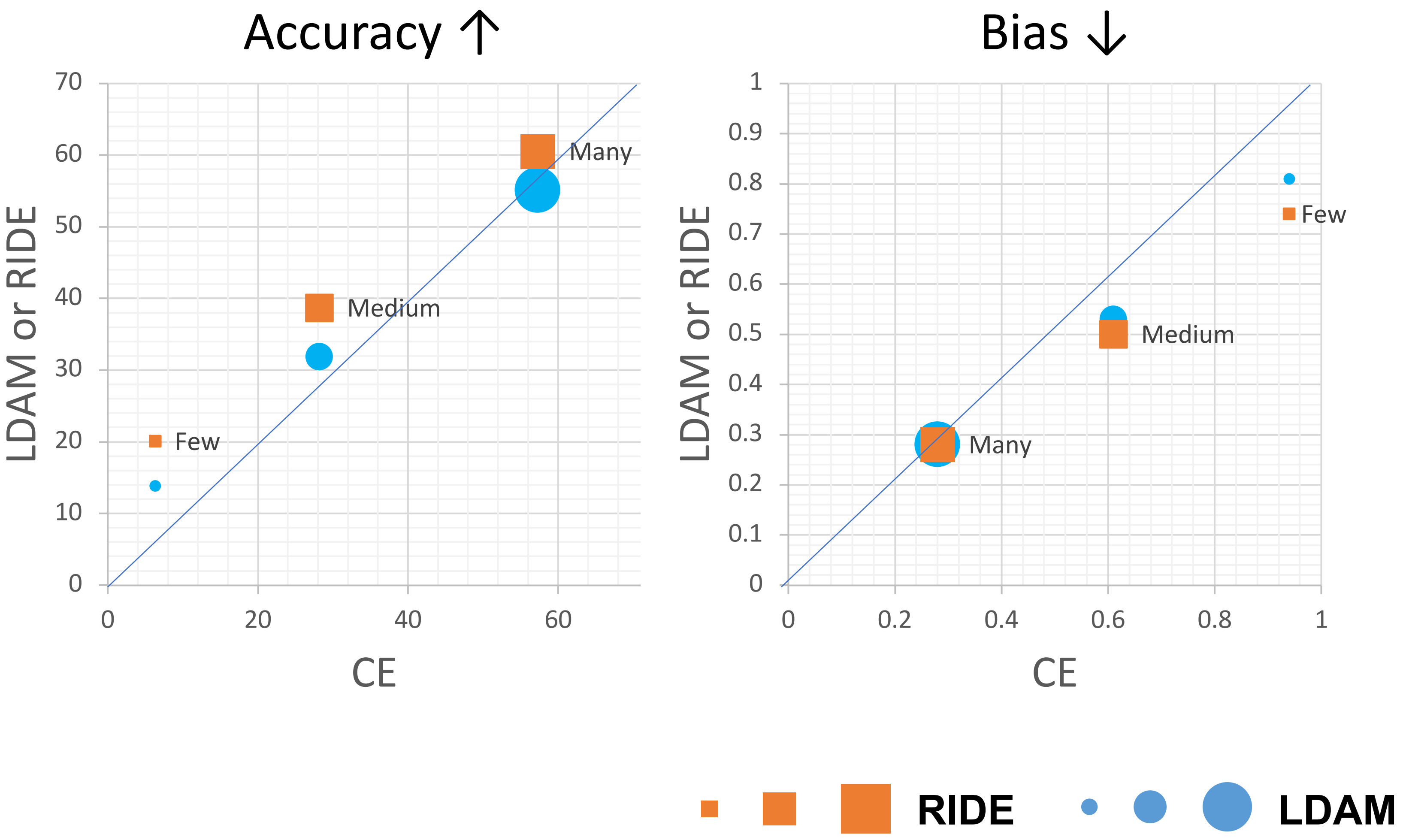
Method Overview



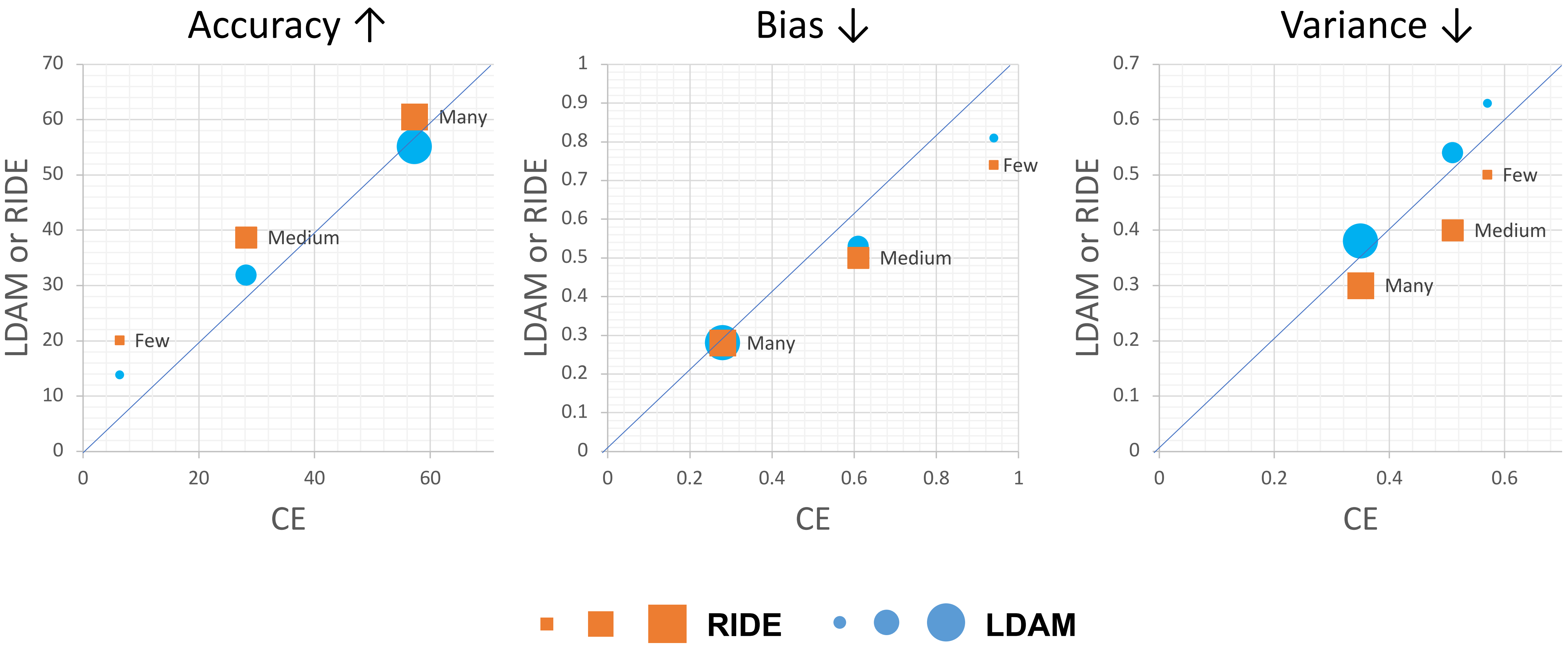
Improving Few-shot Acc. *Without* Sacrificing Many-shot Acc.



RIDE Decreases Bias More Than Other Methods on Few-shot Classes



RIDE Reduces Variances Throughout the Class Spectrum



RIDE vs Current SOTAs

	Head Classes			Tail Classes		
	Acc	Bias	Variance	Acc	Bias	Variance
Current SOTAs	Worse	Comparable	Worse	Better	Better	Worse
RIDE	Better	Better	Better	Better	Better	Better

Better accuracy for all splits.
Better bias-variance trade-off for all splits.

CIFAR100-LT (100 Classes)

SOTA performance on few-shot classes with 5.8% improvements.

Methods	MFlops	Acc. (%)	Many	Med	Few
Cross Entropy (CE) ‡	69.5 (1.0x)	38.3	-	-	-
Cross Entropy (CE) †	69.5 (1.0x)	39.1	66.1	37.3	10.6
Focal Loss ‡ (Lin et al., 2017)	69.5 (1.0x)	38.4	-	-	-
OLTR † (Liu et al., 2019)	-	41.2	61.8	41.4	17.6
LDAM + DRW (Cao et al., 2019)	69.5 (1.0x)	42.0	-	-	-
LDAM + DRW † (Cao et al., 2019)	69.5 (1.0x)	42.0	61.5	41.7	$\boxed{20.2}$
BBN (Zhou et al., 2020)	74.3 (1.1x)	42.6	-	-	-
τ-norm † (Kang et al., 2020)	69.5 (1.0x)	43.2	65.7	43.6	17.3
cRT † (Kang et al., 2020)	69.5 (1.0x)	43.3	64.0	44.8	18.1
M2m (Kim et al., 2020)	-	43.5	-	-	-
LFME (Xiang et al., 2020)	-	43.8	-	-	-
RIDE (2 experts)	**64.8 (0.9x)**	47.0 (+3.2)	67.9	48.4	21.8
RIDE (3 experts)	77.8 (1.1x)	48.0 (+4.2)	68.1	49.2	23.9
RIDE (4 experts)	91.9 (1.3x)	**49.1** (+5.3)	**69.3**	**49.3**	**26.0**

ImageNet-LT (1000 Classes)

Consistent improvements to various backbones by 6.9~7.7%

Methods	ResNet-50		ResNeXt-50	
	GFlops	Acc. (%)	GFlops	Acc. (%)
Cross Entropy (CE) †	4.11 (1.0x)	41.6	4.26 (1.0x)	44.4
OLTR † (Liu et al., 2019)	-	-	-	46.3
NCM (Kang et al., 2020)	4.11 (1.0x)	44.3	4.26 (1.0x)	47.3
τ-norm (Kang et al., 2020)	4.11 (1.0x)	46.7	4.26 (1.0x)	49.4
cRT (Kang et al., 2020)	4.11 (1.0x)	47.3	4.26 (1.0x)	49.6
LWS (Kang et al., 2020)	4.11 (1.0x)	47.7	4.26 (1.0x)	49.9
RIDE (2 experts)	**3.71 (0.9x)**	54.4 (+6.7)	**3.92 (0.9x)**	55.9 (+6.0)
RIDE (3 experts)	4.36 (1.1x)	54.9 (+7.2)	4.69 (1.1x)	56.4 (+6.5)
RIDE (4 experts)	5.15 (1.3x)	**55.4** (+7.7)	5.19 (1.2x)	**56.8** (+6.9)

iNaturalist (8000 Classes)

Significantly better performance on many-shot than current SOTA BBN.

Methods	GFlops	All	Many	Medium	Few
CE †	4.14 (1.0x)	61.7	72.2	63.0	57.2
CB-Focal †	4.14 (1.0x)	61.1	-	-	-
OLTR	4.14 (1.0x)	63.9	59.0	64.1	64.9
LDAM + DRW †	4.14 (1.0x)	64.6	-	-	-
cRT	4.14 (1.0x)	65.2	69.0	66.0	63.2
τ-norm	4.14 (1.0x)	65.6	65.6	65.3	65.9
LWS	4.14 (1.0x)	65.9	65.0	66.3	65.5
BBN	4.36 (1.1x)	66.3	49.4	70.8	65.3
RIDE (2 experts)	**3.67 (0.9x)**	71.4 (+5.1)	70.2 (+1.2)	71.3 (+0.5)	71.7 (+5.8)
RIDE (3 experts)	4.17 (1.0x)	72.2 (+5.9)	70.2 (+1.2)	72.2 (+1.4)	72.7 (+6.8)
RIDE (4 experts)	4.51 (1.1x)	**72.6** (+6.3)	**70.9** (+1.9)	**72.4** (+1.6)	**73.1** (+7.2)

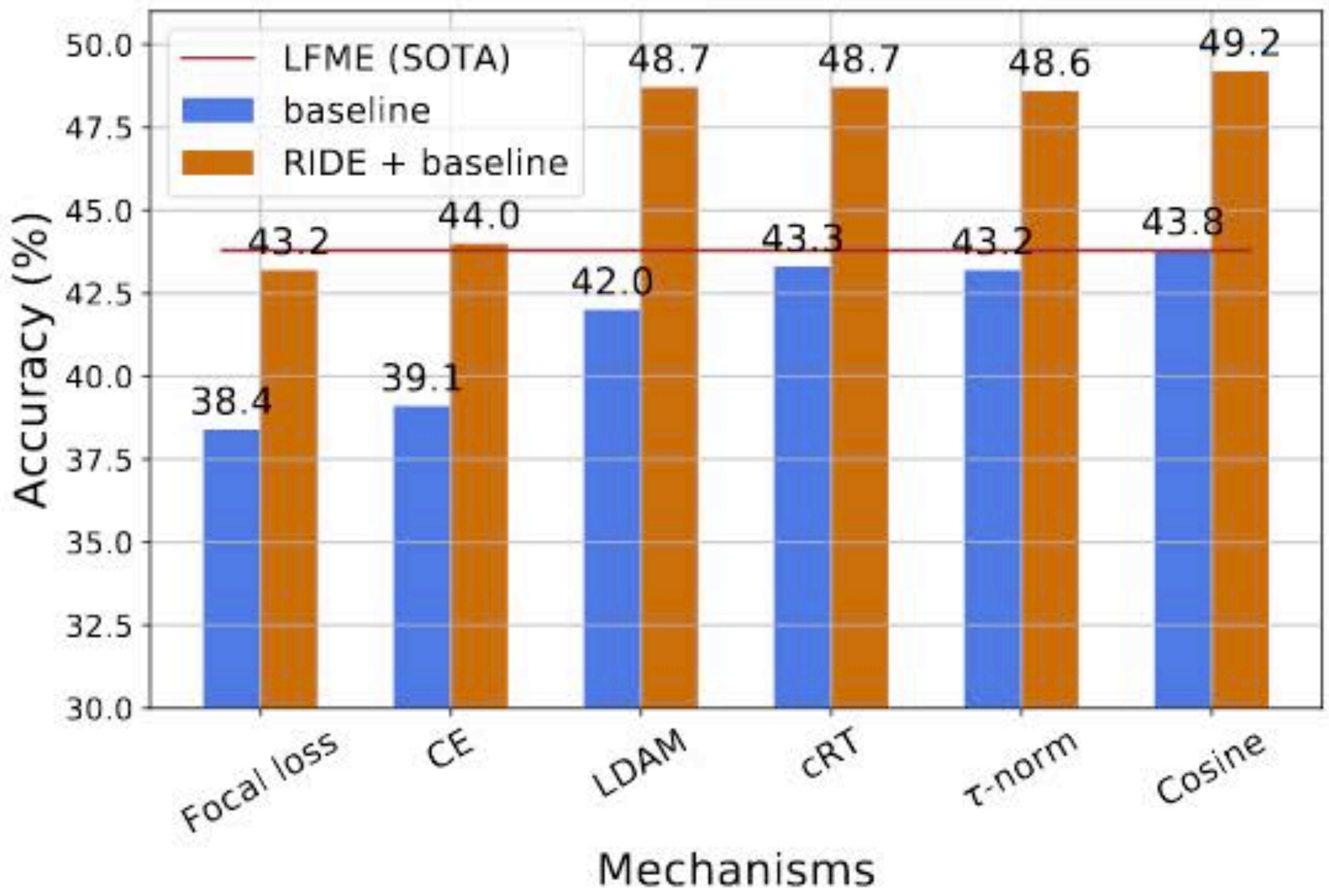
iNaturalist (8000 Classes)

SOTA performance on iNaturalist with the largest improvements from few-shot classes.

Methods	GFlops	All	Many	Medium	Few
CE †	4.14 (1.0x)	61.7	72.2	63.0	57.2
CB-Focal †	4.14 (1.0x)	61.1	-	-	-
OLTR	4.14 (1.0x)	63.9	59.0	64.1	64.9
LDAM + DRW †	4.14 (1.0x)	64.6	-	-	-
cRT	4.14 (1.0x)	65.2	69.0	66.0	63.2
τ-norm	4.14 (1.0x)	65.6	65.6	65.3	65.9
LWS	4.14 (1.0x)	65.9	65.0	66.3	65.5
BBN	4.36 (1.1x)	66.3	49.4	70.8	65.3
RIDE (2 experts)	**3.67 (0.9x)**	71.4 (+5.1)	70.2 (+1.2)	71.3 (+0.5)	71.7 (+5.8)
RIDE (3 experts)	4.17 (1.0x)	72.2 (+5.9)	70.2 (+1.2)	72.2 (+1.4)	72.7 (+6.8)
RIDE (4 experts)	4.51 (1.1x)	**72.6** (+6.3)	**70.9** (+1.9)	**72.4** (+1.6)	**73.1** (+7.2)

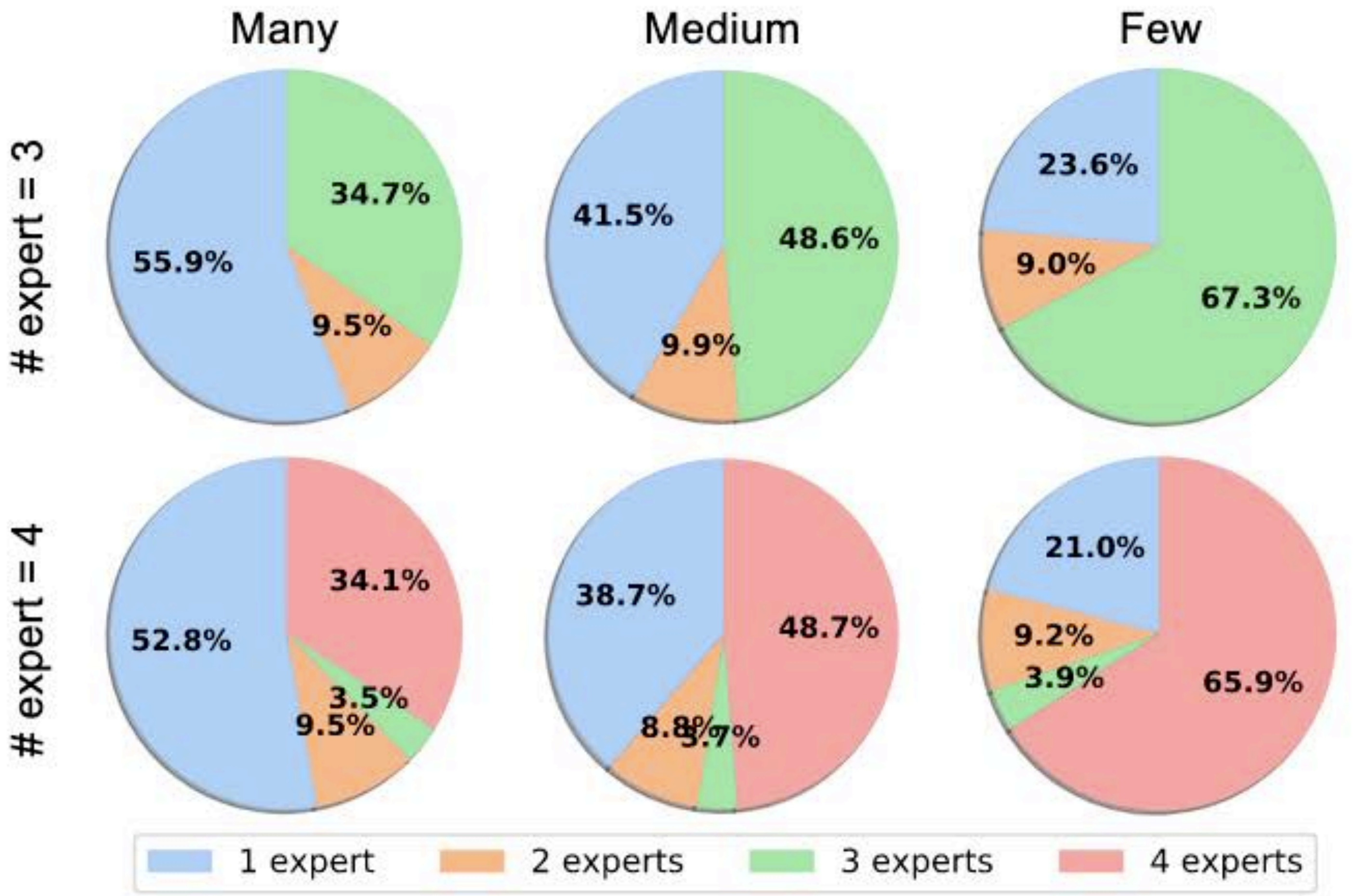
RIDE is a Universal Framework

Consistent improvements to various methods can be obtained



Expert Assignment: Tail Classes Require More Experts

More than half samples in few-shot require more than one expert
More than half samples in many-shot only require one expert



Summary

- ✓ RIDE is the first paper to theoretically analyze the long tail problem from the perspective of bias-variance decomposition.
- ✓ RIDE is the first paper that increases the performances on all three splits (many-/med-/few-shot).
- ✓ RIDE significantly outperforms current state-of-the-arts on all experimented benchmarks by 5%~8%, including CIFAR100-LT, ImageNet-LT and iNaturalist.
- ✓ RIDE is a universal framework that can be integrated with various existing methods, which provides a strong framework for future research in long-tailed recognition.

Long-tailed Recognition by Routing Diverse Distribution-Aware Experts

Xudong Wang[1], **Long Lian**[1], **Zhongqi Miao**[1], **Ziwei Liu**[2], **Stella X. Yu**[1]
[1]UC Berkeley / ICSI, [2]Nanyang Technological University
{xdwang,longlian,zhongqi.miao,stellayu}@berkeley.edu
ziwei.liu@ntu.edu.sg

Project Page

Code