Feature Directions Matter: Long-Tailed Learning via Rotated Balanced Representation

Peifeng Gao, Qianqian Xu, Peisong Wen,
Zhiyong Yang, Huiyang Shao, Qingming Huang

中国科学院大学
University of Chinese Academy of Sciences

中科院计算所
INSTITUTE OF COMPUTING TECHNOLOGY, CAS

0

Topic

- **Motivation:**

  Do We Really Need a Learnable Classifier at the End of Deep Neural Network?

- **Proposed Method:**

  Representation-Balanced Learning

- **Experiment:**

  Generalization Analysis

  Performance Comparison

1 Background

Neural Collapse

Logit

The linear classifier

$$logit = Mf(x;w)$$

A data point: x

The parameter of deep model $f(\cdot;w)$

1 Background

Neural Collapse

1 Background

Neural Collapse

Three manifestations in the classifier and last-layer feature:

NC1 Variability Collapse All samples belonging to the same class converge to the class mean

NC2 Convergence to Self Duality The samples and classifier belonging to the same class converge to the same

NC3 Convergence to Simplex ETF The classifier weight converges to the vertices of Simplex Equiangular Tight Frame (ETF).

1

Background

Neural Collapse

Three manifestations in the classifier and last-layer feature:

NC1 Variability Collapse All samples belonging to the same class converge to the class mean:

NC2 Convergence to Self Duality The samples and classifier belonging to the same class converge to the same:

NC3 Convergence to Simplex ETF The classifier weight converges to the vertices of Simplex Equiangular Tight Frame (ETF).

Definition. Simplex Equiangular Tight Frame

A Simplex ETF is a collection of points in $\mathbb{R}^C$:

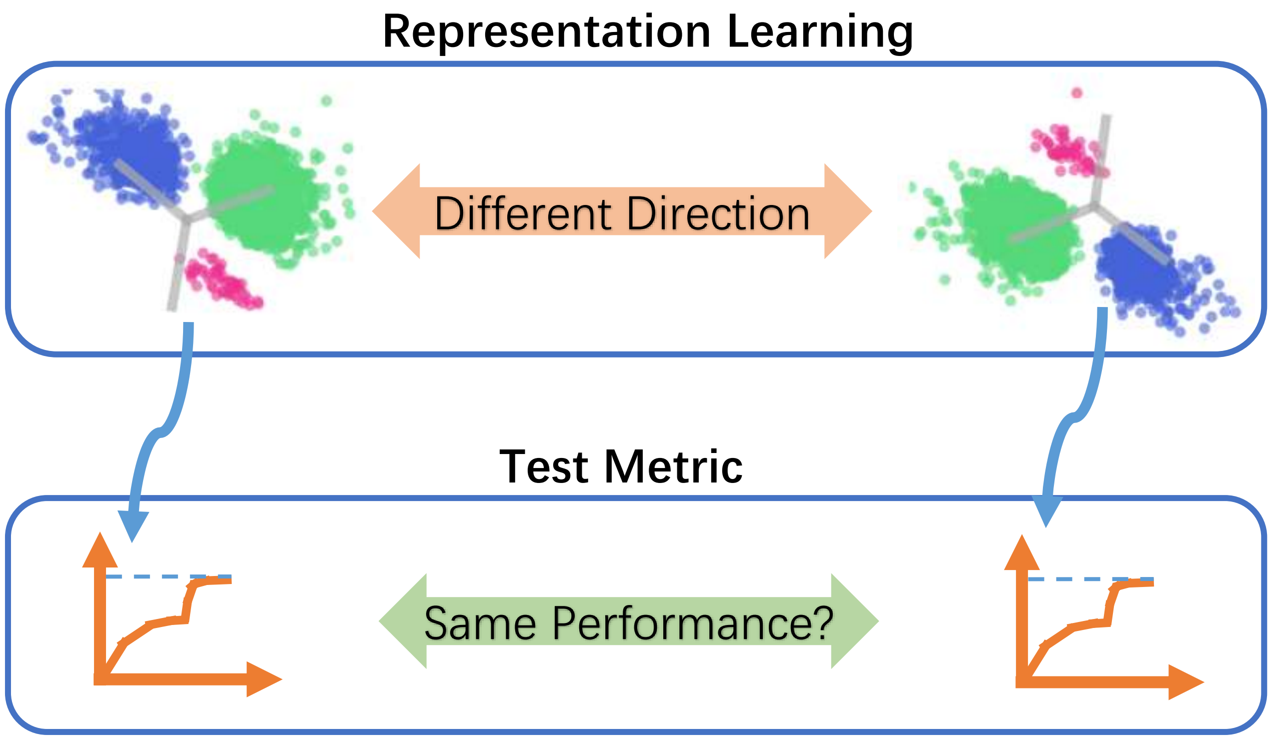
$$M^{\star} = \alpha R \sqrt{\frac{C}{C-1}}\left(I - \frac{1}{C}\mathbb{I}\mathbb{I}^T\right)$$

where α is a scale factor and R is the orthogonal matrix in $\mathbb{R}^{C\times d}(\mathrm{d} \geq C)$.

2 Motivation

Do We Really Need a Learnable Classifier?

Do we really need to learn a linear classifier at the end of deep model? [1]



[1] Yang Y, Xie L, Chen S, et al. Do we really need a learnable classifier at the end of deep neural network?[J]. arXiv preprint arXiv:2203.09081, 2022.

2

Motivation

Do We Really Need a Learnable Classifier?

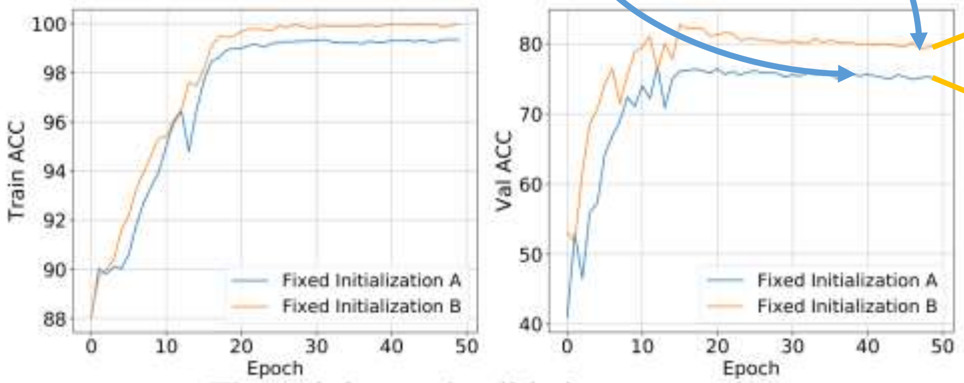
Features visualization on training set at epoch 30. Left: initialization A. Right: initialization B.

2

Motivation

Do We Really Need a Learnable Classifier?

Features visualization on training set at epoch 30. Left: initialization A. Right: initialization B.



The training and validation accuracies.

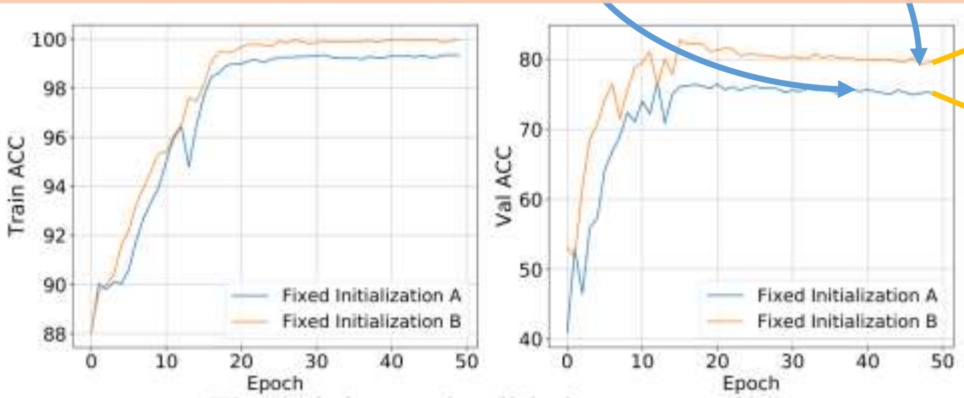
A large gap

Motivation

Do We Really Need a Learnable Classifier?

Features visualization on training set at epoch 30. Left: initialization A. Right: initialization B.

Different directions of ETF lead to different generalization!

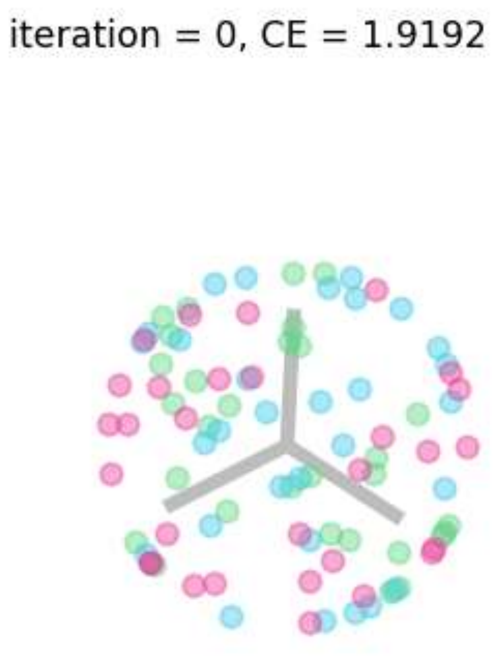


The training and validation accuracies.

A large gap

3 Proposed Method

Learning Objective



$$logit = Mf(x; w)$$

$$logit = MRf(x; w)$$

R: A learnable orthogonal matrix registering the directions of features and classifiers

A learnable orthogonal matrix is introduced to learn directions of feature

3 Proposed Method

Learning Objective

Optimization of Rotation Matrix

- Lie Algebra: $\mathfrak{so}(d) = \{A \in \mathbb{R}^{d\times d} | A + A^T = 0\}$
- Lie Group: $SO(d) = \{A \in \mathbb{R}^{d\times d} | A^T A = I\}$

Step1 Optimization over $SO(d) \rightarrow$ Optimization over $\mathfrak{so}(d)$:

the exponential of matrices gives a parametrization of $\mathrm{SO}(d)$

$$\exp(A) = I + A + \frac{A^2}{2} + \cdots$$

Step2 Optimization over $\mathfrak{so}(d)$ $\rightarrow$ Optimization over $\mathbb{R}^{\frac{d(d-1)}{2}}$:

for $\mathfrak{so}(d)$, the isomorphism is given by following mapping

$$\mathbb{R}^{\frac{d(d-1)}{2}} \rightarrow , A \mapsto A - A^T$$

$$\min_{A \in SO(d)} loss(A)$$

$$A = \exp(\mathrm{B})$$

$$\min_{B \in \mathfrak{so}(d)} loss(\exp(B))$$

$$\mathrm{B} = C - C^T$$

$$\min_{\mathrm{C} \in \mathbb{R}^{d\times d}} loss(\exp(C - C^T))$$

3

Proposed Method

Post-Hoc Logit Adjustment

When testing, a set of margins is subtracted:

$$\arg\max_{i\in[1,\ldots,C]}[M^{\star}Rf(x;w)-\log(N_i/N)]_i$$

[1] Ren, J., Yu, C., Ma, X., Zhao, H., Yi, S., et al. Balanced meta-softmax for long-tailed visual recognition. Advances in neural information processing systems, 33: 4175–4186, 2020.
[2] Menon, A. K., Jayasumana, S., Rawat, A. S., Jain, H., Veit, A., and Kumar, S. Long-tail learning via logit adjustment. arXiv preprint arXiv:2007.07314, 2020.

4 Experiment

Generalization Analysis

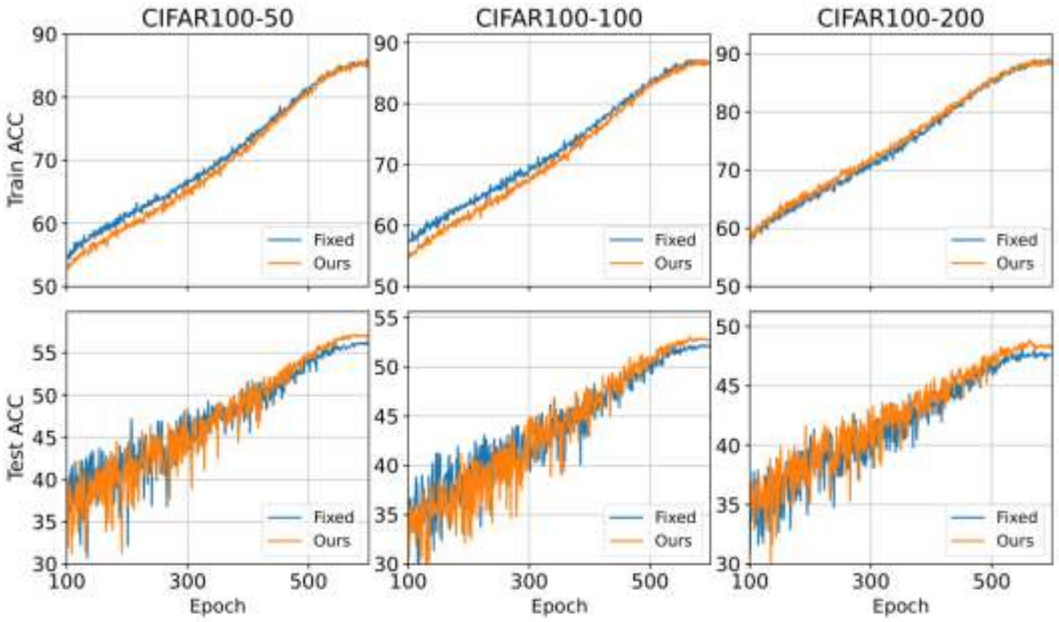


Figure 3: Generalization analysis on CIFAR100. The two rows show the accuracies of *Fixed* and our method on training set and test set in every epoch respectively.

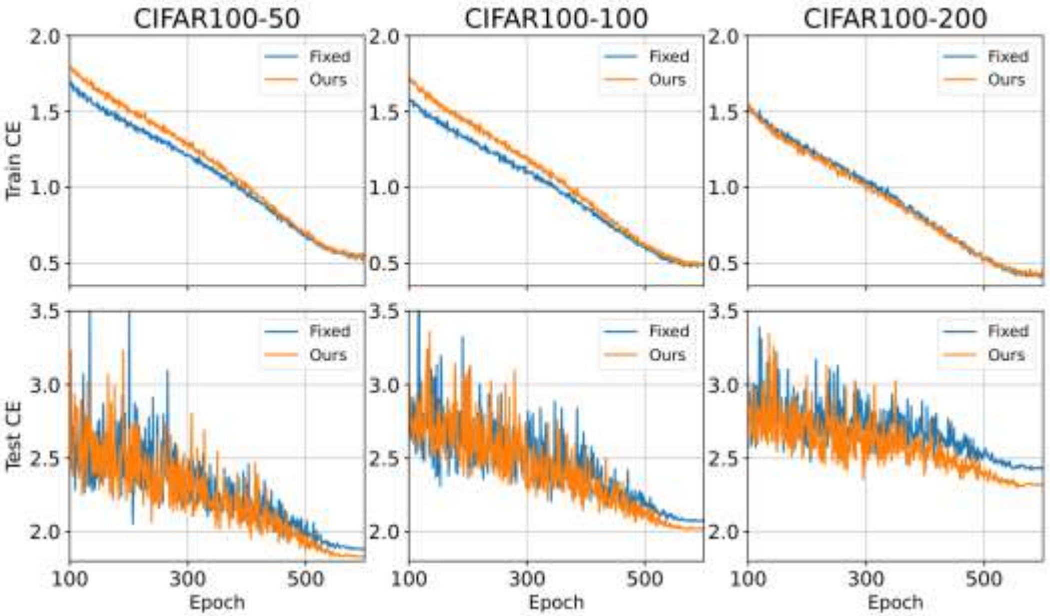


Figure 4: Generalization analysis on CIFAR100. The two rows show the cross entropy loss of *Fixed* and our method on training set and test set in every epoch respectively.

4

Experiment

Performance Comparison

Table 1: Test accuracies on CIFAR10/100-LT. The best and second best results are marked as **bold** and underline. Rows with † denote results borrowed from (Wang et al., 2021c). Results of other competitors are taken from original papers.

Method	CIFAR-10			CIFAR-100		
	50	100	200	50	100	200
CB	79.3	74.6	68.9	45.3	39.6	36.2
LADE	-	-	-	50.5	45.4	-
Calibrated	84.3	82.8	78.5	51.1	45.5	42.1
cRT†	-	82.0	76.6	-	50.0	44.5
LWS†	-	83.7	78.1	-	50.5	45.3
BS†	-	83.1	79.0	-	50.3	45.9
MARC	-	**85.3**	81.1	-	50.8	47.4
HCL	85.4	81.4	-	51.9	46.7	-
TSC	82.9	79.7	-	47.4	43.8	-
Fixed	87.1	84.0	80.2	56.2	52.3	47.2
RBL	**87.6**	84.7	**81.2**	**57.2**	**53.1**	**48.9**

Table 2: Test accuracies on ImageNet-LT. The best and second best results are marked as **bold** and underline. Rows with † denote results borrowed from (Wang et al., 2021c). Results of other competitors are taken from original papers.

Method	Many	Medium	Few	All
Calibrated	-	-	-	48.4
cRT	61.8	46.2	27.4	49.6
LWS	60.2	47.2	30.3	49.9
Seesaw	**67.1**	45.2	21.4	50.4
BS†	62.2	48.8	29.8	51.4
MARC	60.4	50.3	**36.6**	52.3
LADE	65.1	48.9	33.4	53.0
KCL	61.8	49.4	30.9	51.5
TSC	63.5	**49.7**	30.4	52.4
Fixed	64.3	47.6	27.2	51.2
RBL	64.8	49.6	34.2	**53.3**

Thanks for your attention!