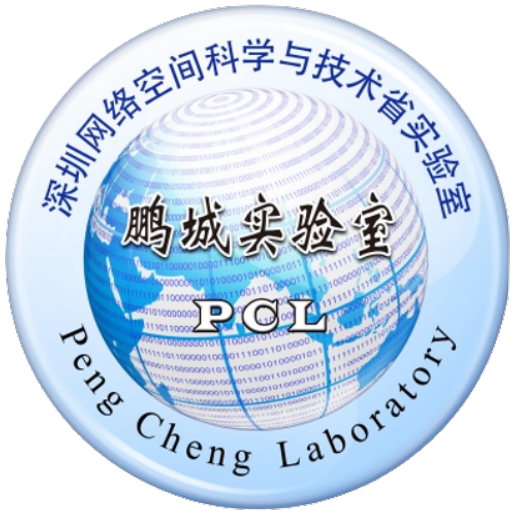
Learning with Noisy Labels via Sparse Regularization

Xiong Zhou[1 2], Xianming Liu[1 2], Chenyang Wang[1], Deming Zhai[1], Junjun Jiang[1 2], Xiangyang Ji[3]

[1] *Harbin Institute of Technology*

[2] *Peng Cheng Laboratory*

[3] *Tsinghua University*



Learning with Noisy Labels

- Given a sample $(\boldsymbol{x}_n, y_n)$, the label noise model can be formulated as

$$\tilde{y}_n = \begin{cases} y_n & \text{with probability } (1 - \eta_{\boldsymbol{x}_n}) \\ i, i \in [k], i \neq y_n & \text{with probability } \eta_{\boldsymbol{x}_n, i} \end{cases}.$$

Learning with Noisy Labels

- Given a sample $(\boldsymbol{x}_n, y_n)$, the label noise model can be formulated as

$$\tilde{y}_n = \begin{cases} y_n & \text{with probability } (1 - \eta_{\boldsymbol{x}_n}) \\ i, i \in [k], i \neq y_n & \text{with probability } \eta_{\boldsymbol{x}_n, i} \end{cases}.$$

- For a k-classification problem, a loss $L \geq 0$ will be noise-tolerant, if it satisfies the symmetric condition

$$\sum_{i=1}^{k} L(f(\boldsymbol{x}), i) = C, \forall x \in \mathcal{X}, \forall f \in \mathcal{H}$$

- where $\mathcal{H}$ is the hypothesis class, C is a fixed constant.

Learning with Noisy Labels

- Given a sample $(\boldsymbol{x}_n, y_n)$, the label noise model can be formulated as

$$\tilde{y}_n = \begin{cases} y_n & \text{with probability } (1 - \eta_{\boldsymbol{x}_n}) \\ i, i \in [k], i \neq y_n & \text{with probability } \eta_{\boldsymbol{x}_n, i} \end{cases}.$$

- For a k-classification problem, a loss $L \geq 0$ will be noise-tolerant, if it satisfies the symmetric condition

$$\sum_{i=1}^{k} L(f(\boldsymbol{x}), i) = C, \forall x \in \mathcal{X}, \forall f \in \mathcal{H}$$

- where $\mathcal{H}$ is the hypothesis class, C is a fixed constant.
- The existing methods only focus on designing new losses satisfies the symmetric condition, but they have ignored another way where we can constrain the hypothesis class $\mathcal{H}$.

Output permutations

Definition 1. *Given a vector* $\boldsymbol{v} \in \mathbb{R}^k$*, the permutation on it is defined as*

$$\boldsymbol{v}_\pi = P_\pi \boldsymbol{v}$$

where $P_{\boldsymbol{\pi}} = \left[\boldsymbol{e}_{\pi_1}, \boldsymbol{e}_{\pi_2}, \dots, \boldsymbol{e}_{\pi_k}\right]^T$ *is the permutation matrix, and* $\{\pi_1, \dots, \pi_k\} = [k]$.

- For example, when $k = 3$, $\boldsymbol{v} = (v_1, v_2, v_3)^T$, and $\boldsymbol{\pi} = [3,1,2]$, then $P_{\boldsymbol{\pi}} = [\boldsymbol{e}_3, \boldsymbol{e}_1, \boldsymbol{e}_2]^T$, and the vector after permutation operation is $\boldsymbol{v}_\pi = (v_3, v_1, v_2)^T$.

Output permutations

Definition 1. *Given a vector* $\boldsymbol{v} \in \mathbb{R}^k$*, the permutation on it is defined as*

$$\boldsymbol{v}_\pi = P_\pi \boldsymbol{v}$$

where $P_{\boldsymbol{\pi}} = \left[\boldsymbol{e}_{\pi_1}, \boldsymbol{e}_{\pi_2}, \dots, \boldsymbol{e}_{\pi_k}\right]^T$ *is the permutation matrix, and* $\{\pi_1, \dots, \pi_k\} = [k]$.

- For example, when $k = 3$, $\boldsymbol{v} = (v_1, v_2, v_3)^T$, and $\boldsymbol{\pi} = [3,1,2]$, then $P_{\boldsymbol{\pi}} = [\boldsymbol{e}_3, \boldsymbol{e}_1, \boldsymbol{e}_2]^T$, and the vector after permutation operation is $\boldsymbol{v}_\pi = (v_3, v_1, v_2)^T$.
- More generally, let $\mathcal{P}_{\boldsymbol{v}}$ denote the permutation set over $\boldsymbol{v}$, we have

$$\sum_{i=1}^{k} \ell(u_i) = \sum_{i=1}^{k} \ell(v_i), \qquad \forall\, \boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}$$

- As can be seen, if $\boldsymbol{v}$ is fixed, then any loss ℓ satisfies the symmetric condition, $\forall\, f\colon \mathcal{X} \to \mathcal{P}_{\boldsymbol{v}}$.

Output permutations

Lemma 1. *Given a vector* $\boldsymbol{v} \in \mathbb{R}^k$, $L(\boldsymbol{u}, i) = \ell_1(u_i) + \sum_{j \neq i} \ell_2(u_j)$, *we have*

$$\sum_{i=1}^{k} L(\boldsymbol{u}, i) = C, \qquad \forall\, \boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}$$

where $C = \sum_{i=1}^{k} L(\boldsymbol{v}, i)$ *is a constant when* $\boldsymbol{v}$ *is fixed.*

Lemma 2. *Given a vector* $\boldsymbol{v} \in \mathbb{R}^k$ *and weights* $w_1, \ldots, w_k \geq 0$ $(\exists t \in [k]$, *s.t.* $w_t > \max_{i \neq t} w_i)$, $L(\boldsymbol{u}, i) = \ell_1(u_i) + \sum_{j \neq i} \ell_2(u_j)$, *we have*

$$\arg\min_{\boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}} \sum_{i=1}^{k} w_i L(\boldsymbol{u}, i) = \arg\min_{\boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}} L(\boldsymbol{u}, t)$$

where $\boldsymbol{v}$ *satisfies some condition.*

These two lemmas indicate that any loss can be robust if the output is in a fixed permutation space.

Noise Tolerance

Theorem 1 (Noise tolerance under symmetric noise). *In a multi-class classification problem, for* $L(\boldsymbol{u}, i) = \ell_1(u_i) + \sum_{j \neq i} \ell_2(u_j)$, *$L$ is noise-tolerant under symmetric label noise if* $\eta < 1 - \frac{1}{k}$ *and* $f\colon \mathcal{X} \to \mathcal{P}_{\boldsymbol{v}}$, *i.e.,*

$$\underset{f:\mathcal{X}\to\mathcal{P}_{\boldsymbol{v}}}{arg\,min}\, R_L(f) = \underset{f:\mathcal{X}\to\mathcal{P}_{\boldsymbol{v}}}{arg\,min}\, R_L^{\eta}(f)\,,$$

where $\boldsymbol{v}$ *is a fixed vector.*

Theorem 2 (Noise tolerance under asymmetric noise). *In a multi-class classification problem, let* $f\colon \mathcal{X} \to \mathcal{P}_{\boldsymbol{v}}$, *where* $\boldsymbol{v}$ *is a fixed vector, and suppose that* $L(\boldsymbol{u}, i) = \ell_1(u_i) + \sum_{j \neq i} \ell_2(u_j)$ *satisfies* $0 \leq L(f(\boldsymbol{x}), i) \leq \frac{C}{k-1}$, $\forall\, i \in [k]$. *If* $R_L(f^*) = 0$, *then L is noise-tolerant under asymmetric or class-conditional noise when* $\eta_{\boldsymbol{x}_n,i} < 1 - \eta_y$ *with* $\sum_{k \neq y} \eta_{\boldsymbol{x}_n,i} = \eta_y$, $\forall\, \boldsymbol{x}$.

ϵ-relaxation

- Theorem 1 and Theorem 2 shows that label noise can be mitigated by restricting the network output to a permutation set.
- However, the optimization is non-trivial because $\mathcal{P}_{\boldsymbol{v}}$ leads to a discrete mapping. Instead, we turn to approximate the constraint by **ϵ-relaxation** :

$$\mathcal{H}_{\boldsymbol{v},\boldsymbol{\epsilon}} = \{f : \min_{\boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}} ||f(\boldsymbol{x}) - \boldsymbol{u}||_2 \leq \epsilon, \forall \boldsymbol{x}\}$$

ϵ-relaxation

- Theorem 1 and Theorem 2 shows that label noise can be mitigated by restricting the network output to a permutation set.
- However, the optimization is non-trivial because $\mathcal{P}_{\boldsymbol{v}}$ leads to a discrete mapping. Instead, we turn to approximate the constraint by **ϵ-relaxation** :

$$\mathcal{H}_{\boldsymbol{v},\boldsymbol{\epsilon}} = \{f : \min_{\boldsymbol{u} \in \mathcal{P}_{\boldsymbol{v}}} ||f(\boldsymbol{x}) - \boldsymbol{u}||_2 \leq \epsilon, \forall \boldsymbol{x}\}$$

Theorem 3. *In a multi-class classification problem, for* $L(\boldsymbol{u}, i) = \ell_1(u_i) + \sum_{j \neq i} \ell_2(u_j)$ *satisfies that* $\left|\sum_i^k (L(\boldsymbol{u}_1, i) - L(\boldsymbol{u}_2, i)\right| \leq \delta$ *when* $||\boldsymbol{u}_1 - \boldsymbol{u}_2||_2 \leq \epsilon$, *and* $\delta \to 0$ *as* $\epsilon \to 0$, *then for symmetric label noise satisfying* $\eta < 1 - \frac{1}{k}$, *the risk bound for* $f \in \mathcal{H}_{\boldsymbol{v},\boldsymbol{\epsilon}}$ *can be expressed as*

$$R_L(f_\eta^*) - R_L(f^*) \leq 2c\delta,$$

where $c = \frac{\eta}{(1-\eta)k-1}$, f_η^* *and* f^* *denote the global minimum of* $R_L^\eta(f)$ *and* $R_L(f)$, *respectively.*

Sparse Regularization

- A simple yet efficient method to implement the permutation set is considering v as a one-hot vector, then we can just require the output of network to be sparse.
- ***Network Output Sharpening***:

$$\sigma_\tau(z)_i = \frac{\exp(z_i/\tau)}{\sum_{j=1}^{k}\exp(z_j/\tau)}$$

- ℓ_p***-norm regularization***:

$$\min_{f\in\mathcal{H}} R_L(f) \quad s.t. \quad \left|\left|f\right|\right|_p \le \gamma \quad \Rightarrow \quad \sum_{i=1}^{N} L(f(x_i), y_i) + \lambda\left|\left|f(x_i)\right|\right|_p^p$$

On the robustness and learning sufficiency

- For output sharpening, the gradient of $\sigma_\tau(z)_j$ with respect to z_i can be derived as

$$\frac{\partial \sigma_\tau(z)_j}{\partial z_i} = \frac{1}{\tau}\sigma_\tau(z)_i(\delta_{ij} - \sigma_\tau(z)_j)$$

- where $\delta_{ij} = \mathbb{I}(i = j)$, and $\mathbb{I}(\cdot)$ is the identity function.
- An appropriate step size would speed up the convergence to one-hot vectors, so we can change the value of τ.
- On the other hand, we have $\lim_{\tau \to 0^+} \frac{\partial \sigma_\tau(z)_j}{\partial z_i} = 0$. This indicates that the gradient will disappear when τ is small.

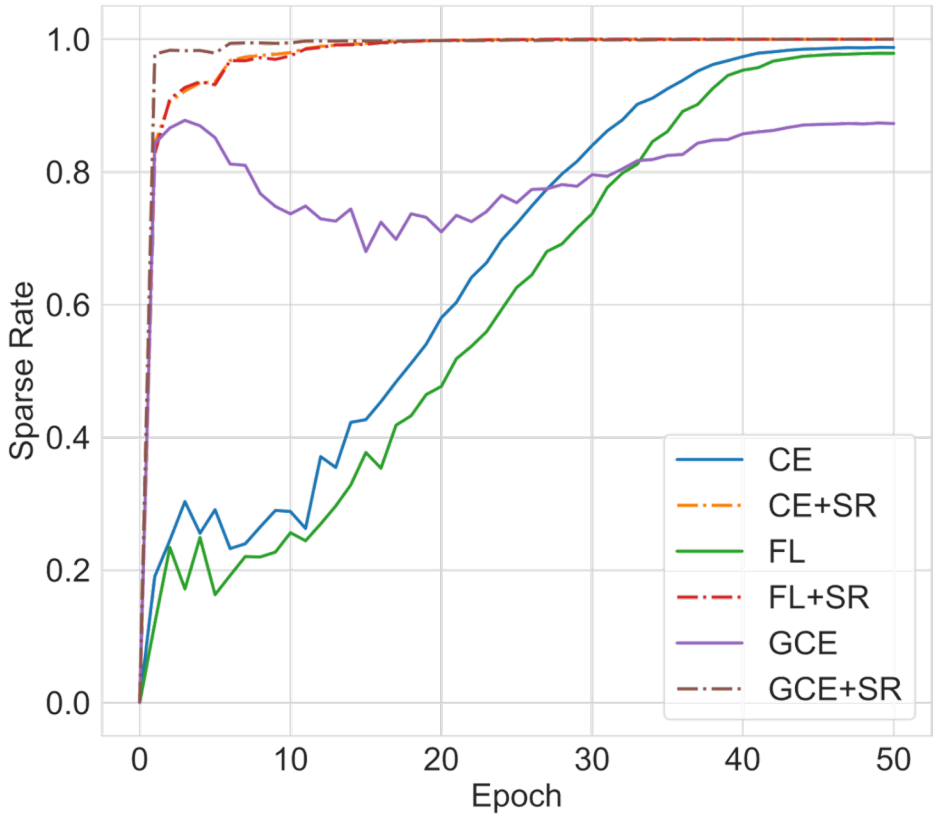
On the robustness and learning sufficiency

- Consider $L(\sigma_\tau(\boldsymbol{z}), y) = -\log \sigma_\tau(\boldsymbol{z})_y$, the gradient of the loss with ℓ_p-norm can be derived as

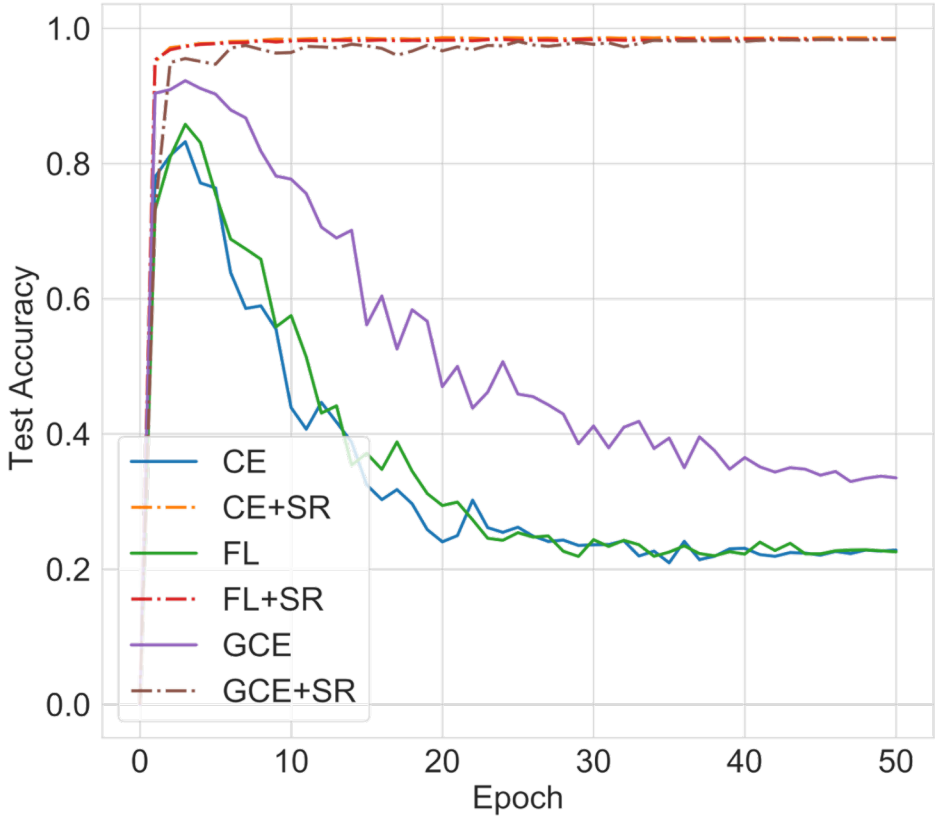
$$\frac{\partial -\log \sigma_\tau(\boldsymbol{z})_y + \lambda \|\sigma_\tau(\boldsymbol{z})\|_p^p}{\partial \boldsymbol{z}} = \underbrace{-\left(\frac{1}{\sigma_\tau(\boldsymbol{z})_y} - \frac{\lambda p}{[\sigma_\tau(\boldsymbol{z})_y]^{1-p}}\right) \cdot \frac{\partial \sigma_\tau(\boldsymbol{z})_y}{\partial \boldsymbol{z}}}_{\text{fitting term}} + \underbrace{\lambda p \sum_{i \neq y} \frac{1}{[\sigma_\tau(\boldsymbol{z})_i]^{1-p}} \cdot \frac{\partial \sigma_\tau(\boldsymbol{z})_i}{\partial \boldsymbol{z}}}_{\text{complementary term}}$$

- The fitting term denotes the gradient of learning towards the target y, while the complementary term limits the increase of $\sigma_\tau(\boldsymbol{z})_i, \forall\, i \neq y$.
- In the early phase of training, we should guarantee enough fitting power by setting $\lambda p < 1$. As λ increases, the fitting term becomes weaker to mitigate label noise, but the complementary term still maintains a certain amount of fitting power through minimizing $\sigma_\tau(\boldsymbol{z})_i, \forall\, i \neq y$ *to passively maximize* $\sigma_\tau(\boldsymbol{z})_y$.

On the robustness and learning sufficiency



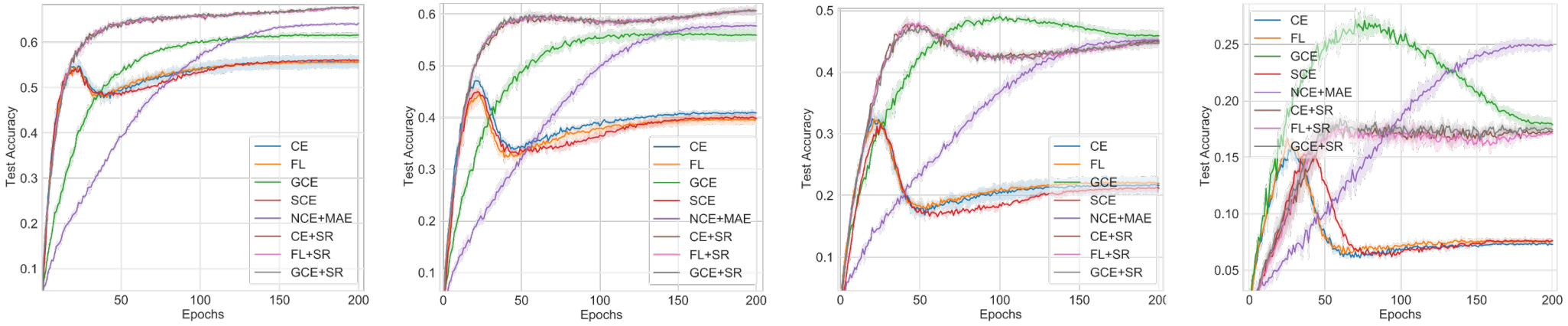
(a) Sparse Rate



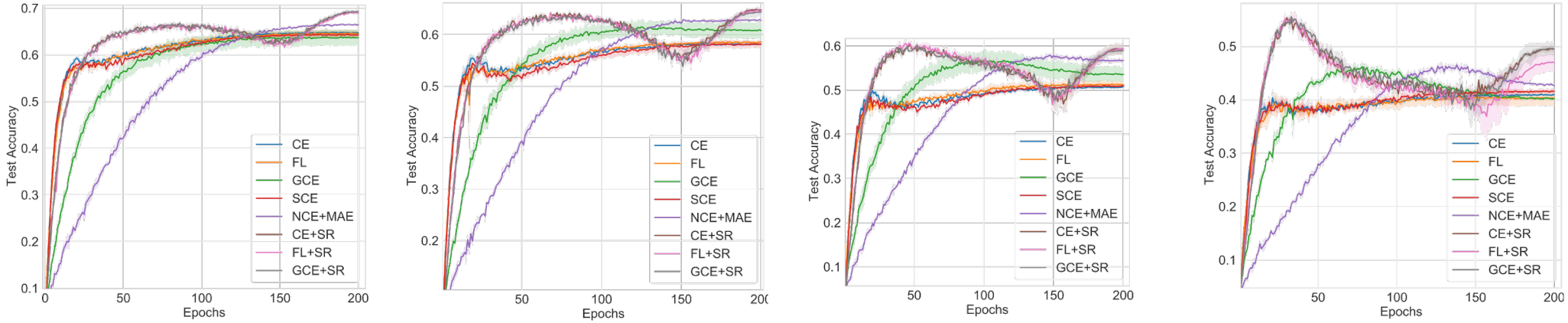
(b) Test Accuracy

Figure 1. Sparse rate and test accuracy of different methods on MNIST with 0.8 symmetric label noise.

On the robustness and learning sufficiency



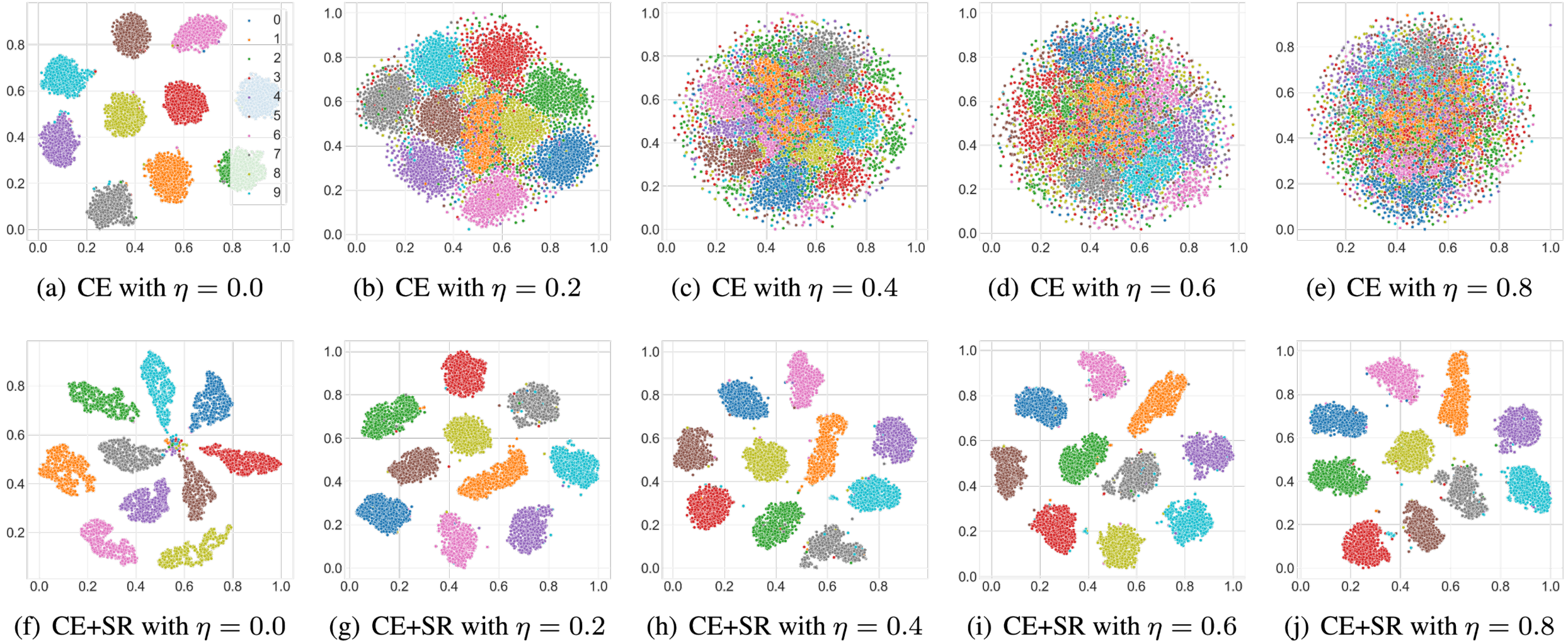
(b) $\eta = 0.2$ (c) $\eta = 0.4$ (d) $\eta = 0.6$ (e) $\eta = 0.8$



(f) $\eta = 0.1$ (g) $\eta = 0.2$ (h) $\eta = 0.3$ (i) $\eta = 0.4$

Figure 2. Curves of validation accuracy on CIFAR-100 with symmetric and asymmetric label noise.

On the robustness and learning sufficiency



(a) CE with $\eta = 0.0$ (b) CE with $\eta = 0.2$ (c) CE with $\eta = 0.4$ (d) CE with $\eta = 0.6$ (e) CE with $\eta = 0.8$

(f) CE+SR with $\eta = 0.0$ (g) CE+SR with $\eta = 0.2$ (h) CE+SR with $\eta = 0.4$ (i) CE+SR with $\eta = 0.6$ (j) CE+SR with $\eta = 0.8$

Figure 3. Visualization of learned representations on MNSIT with different symmetric label noise.

Experimental Results

Table 1. Test accuracies (%) of different methods on benchmark datasets with clean or symmetric label noise ($\eta \in [0.2, 0.4, 0.6, 0.8]$). The results (mean±std) are reported over 3 random runs and the top 3 best results are **boldfaced**.

Datasets	Methods	Clean ($\eta = 0.0$)	Symmetric Noise Rate (η) 0.2	0.4	0.6	0.8
MNIST	CE	99.15 ± 0.05	91.62 ± 0.39	73.98 ± 0.27	49.36 ± 0.43	22.66 ± 0.61
	FL	99.13 ± 0.09	91.68 ± 0.14	74.54 ± 0.06	50.39 ± 0.28	22.65 ± 0.26
	GCE	99.27 ± 0.05	98.86 ± 0.07	97.16 ± 0.03	81.53 ± 0.58	33.95 ± 0.82
	SCE	99.23 ± 0.10	98.92 ± 0.12	97.38 ± 0.15	88.83 ± 0.55	48.75 ± 1.54
	NLNL	98.85 ± 0.05	98.33 ± 0.03	97.80 ± 0.07	96.18 ± 0.11	86.34 ± 1.43
	APL	99.34 ± 0.02	**99.14 ± 0.05**	98.42 ± 0.09	95.65 ± 0.13	72.97 ± 0.34
	CE+SR	99.33 ± 0.02	**99.22 ± 0.06**	**99.16 ± 0.04**	**98.85 ± 0.02**	**98.06 ± 0.86**
	FL+SR	99.35 ± 0.05	**99.25 ± 0.01**	**99.10 ± 0.10**	**98.81 ± 0.06**	**97.00 ± 1.28**
	GCE+SR	99.27 ± 0.06	99.13 ± 0.07	**99.06 ± 0.02**	**98.84 ± 0.09**	**98.37 ± 0.26**
CIFAR-10	CE	90.48 ± 0.11	74.68 ± 0.25	58.26 ± 0.21	38.70 ± 0.53	19.55 ± 0.49
	FL	89.82 ± 0.20	73.72 ± 0.08	57.90 ± 0.45	38.86 ± 0.07	19.13 ± 0.28
	GCE	89.59 ± 0.26	87.03 ± 0.35	82.66 ± 0.17	67.70 ± 0.45	26.67 ± 0.59
	SCE	91.61 ± 0.19	87.10 ± 0.25	79.67 ± 0.37	61.35 ± 0.56	28.66 ± 0.27
	NLNL	90.73 ± 0.20	73.70 ± 0.05	63.90 ± 0.44	50.68 ± 0.47	29.53 ± 1.55
	APL	89.17 ± 0.09	86.98 ± 0.07	83.74 ± 0.10	76.02 ± 0.16	46.69 ± 0.31
	CE+SR	90.06 ± 0.02	**87.93 ± 0.07**	**84.86 ± 0.18**	**78.18 ± 0.36**	**51.13 ± 0.51**
	FL+SR	89.86 ± 0.11	**87.94 ± 0.19**	**84.65 ± 0.05**	**77.85 ± 0.74**	**52.42 ± 0.76**
	GCE+SR	90.02 ± 0.40	**87.93 ± 0.27**	**84.82 ± 0.06**	**77.65 ± 0.05**	**51.97 ± 1.13**
CIFAR-100	CE	71.33 ± 0.43	56.51 ± 0.39	39.92 ± 0.10	21.39 ± 1.17	7.59 ± 0.20
	FL	70.06 ± 0.70	55.78 ± 1.55	39.83 ± 0.43	21.91 ± 0.89	7.51 ± 0.09
	GCE	63.09 ± 1.39	61.57 ± 1.06	56.11 ± 1.35	**45.28 ± 0.61**	**17.42 ± 0.06**
	SCE	70.64 ± 0.05	56.07 ± 0.26	39.88 ± 0.67	21.16 ± 0.65	7.63 ± 0.15
	NLNL	68.72 ± 0.60	46.99 ± 0.91	30.29 ± 1.64	16.60 ± 0.90	11.01 ± 2.48
	APL	67.95 ± 0.21	64.21 ± 0.24	57.70 ± 0.64	**45.20 ± 0.75**	**24.91 ± 0.42**
	CE+SR	72.19 ± 0.06	**67.51 ± 0.29**	**60.70 ± 0.25**	**44.95 ± 0.65**	**17.35 ± 0.13**
	FL+SR	72.08 ± 0.31	**67.64 ± 0.10**	**60.67 ± 0.48**	44.76 ± 0.08	17.16 ± 0.24
	GCE+SR	72.11 ± 0.26	**67.03 ± 0.46**	**60.68 ± 0.90**	44.66 ± 0.84	17.35 ± 0.42

Table 2. Test accuracies (%) of different methods on benchmark datasets with clean or asymmetric label noise ($\eta \in [0.1, 0.2, 0.3, 0.4]$) The results (mean±std) are reported over 3 random runs and the top 3 best results are **boldfaced**.

Datasets	Methods	Asymmetric Noise Rate (η) 0.1	0.2	0.3	0.4
MNIST	CE	97.57 ± 0.22	94.56 ± 0.22	88.81 ± 0.10	82.27 ± 0.40
	FL	97.58 ± 0.09	94.25 ± 0.15	89.09 ± 0.25	82.13 ± 0.49
	GCE	99.01 ± 0.04	96.69 ± 0.12	89.12 ± 0.24	81.51 ± 0.19
	SCE	99.14 ± 0.04	98.03 ± 0.05	93.68 ± 0.43	85.36 ± 0.17
	NLNL	98.63 ± 0.06	98.35 ± 0.01	97.51 ± 0.15	95.84 ± 0.26
	APL	**99.32 ± 0.09**	98.89 ± 0.04	96.93 ± 0.17	91.45 ± 0.40
	CE+SR	**99.42 ± 0.02**	**99.27 ± 0.06**	**99.24 ± 0.08**	**99.23 ± 0.07**
	FL+SR	**99.34 ± 0.05**	**99.31 ± 0.02**	**99.23 ± 0.02**	**99.36 ± 0.05**
	GCE+SR	99.28 ± 0.06	**99.22 ± 0.02**	**99.13 ± 0.05**	**99.09 ± 0.02**
CIFAR-10	CE	87.55 ± 0.14	83.32 ± 0.12	79.32 ± 0.59	74.67 ± 0.38
	FL	86.43 ± 0.30	83.37 ± 0.07	79.33 ± 0.08	74.28 ± 0.44
	GCE	88.33 ± 0.05	85.93 ± 0.23	80.88 ± 0.38	74.29 ± 0.43
	SCE	**89.77 ± 0.11**	86.20 ± 0.37	81.38 ± 0.35	75.16 ± 0.39
	NLNL	88.54 ± 0.25	84.74 ± 0.08	81.26± 0.43	76.97 ± 0.52
	APL	88.31 ± 0.20	86.50 ± 0.31	83.34 ± 0.39	77.14 ± 0.33
	CE+SR	**89.08 ± 0.08**	**87.70 ± 0.19**	**85.63 ± 0.07**	**79.29 ± 0.20**
	FL+SR	88.68 ± 0.23	**87.56 ± 0.29**	**85.10 ± 0.23**	**79.07 ± 0.50**
	GCE+SR	**89.20 ± 0.23**	**87.55 ± 0.08**	**84.69 ± 0.46**	**79.01 ± 0.18**
CIFAR-100	CE	64.85 ± 0.37	58.11 ± 0.32	50.68 ± 0.55	40.17 ± 1.31
	FL	64.78 ± 0.50	58.05 ± 0.42	51.15 ± 0.84	41.18 ± 0.68
	GCE	63.01 ± 1.01	59.35 ± 1.10	53.83 ± 0.64	40.91 ± 0.57
	NLNL	59.55 ± 1.22	50.19 ± 0.56	42.81 ± 1.13	35.10 ± 0.20
	SCE	64.26 ± 0.43	58.16 ± 0.73	50.98 ± 0.33	41.54 ± 0.52
	APL	66.48 ± 0.12	62.80 ± 0.05	56.74 ± 0.53	42.61 ± 0.24
	CE+SR	**68.96 ± 0.22**	**64.79 ± 0.01**	**59.09 ± 2.10**	**49.51 ± 0.59**
	FL+SR	**68.96 ± 0.17**	**64.61 ± 0.67**	**58.94 ± 0.33**	**46.94 ± 1.68**
	GCE+SR	**69.27 ± 0.31**	**64.35 ± 0.78**	**57.22 ± 0.80**	**49.51 ± 1.31**

Table 3. Top-1 validation accuracies (%) on WebVision validatoin set of ResNet-50 models trained on WebVision using different loss functions, under the Mini setting in [8].

Loss	CE	FL	SCE	APL	**CE+SR**	**FL+SR**
Acc	66.96	63.80	66.92	66.32	**69.12**	**70.28**

Additional Experimental Results

Table 3. A comparison with other methods

Dataset	Method	Label Noise Type			
		sy 0.6	sy 0.8	asy 0.3	asy 0.4
CIFAR-10	CE	38.70	19.55	79.32	74.67
	Co-teaching	65.74	38.01	64.01	51.26
	PHuber-CE	75.44	41.18	76.06	55.78
	CE+SR	**78.18**	**51.13**	**85.63**	**79.29**
CIFAR-100	CE	21.39	7.59	50.68	40.17
	Co-teaching	34.28	7.94	42.82	33.67
	PHuber-CE	21.54	9.33	26.91	23.43
	CE+SR	**44.95**	**17.35**	**59.09**	**49.51**

Table 5. Validation accuracy on CIFAR-10/-100 for imbalance classification using the same experimental settings as Section 3.2.

Dataset	Method	Imbalanced Type			
		lt-0.01	lt-0.1	step-0.01	step-0.1
CIFAR-10	CE	64.16	81.81	57.44	79.35
	CE+SR	**69.78**	**84.49**	**61.03**	**82.11**
CIFAR-100	CE	35.17	51.43	37.92	53.43
	CE+SR	**41.24**	**59.51**	**40.21**	**58.42**

Thanks for your attention!

Any question? Please contact us!

Xiong Zhou: cszx@hit.edu.cn
Xianming Liu: csxm@hit.edu.cn