A Simple yet Effective Baseline for Robust Deep Learning with Noisy Labels

Yucen Luo*
Tsinghua University
luoyc15@mails.tsinghua.edu.cn

Jun Zhu
Tsinghua University
dcszj@mail.tsinghua.edu.cn

Tomas Pfister
Google Cloud AI
tpfister@google.com

CoRR 2019

Contents

- Problem Scenario
- Variance-based Regularization
- Experiments

Problem Scenario

- DNNs need a large-scale training dataset, which can be collected by crowd-sourcing
- DNNs are prone to overfit noisy training data
- The target is to learn a robust K-class classifier $\boldsymbol{f}$ from a training dataset of images with noisy supervision

Related Works

- Reweigh training examples
  - Assign important weights to examples with a high chance of being correct.
  - An insufficient usage of training data.
  - Need some prior knowledge on the noise ratio or the availability of an additional clean unbiased validation dataset, which is usually impractical
- Correction-based
  - Estimating the noisy corruption matrix and correcting the labels
  - It is often difficult to estimate the underlying noise corruption matrix when the number of classes is large
  - There may not be an underlying ground truth corruption process but an open set of noisy labels in the real world

Variance-based Regularization

- Minimize the predictive variance

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \ell(\mathbf{f}(\mathbf{x}_i;\boldsymbol{\theta}), \tilde{y}_i) + \lambda \hat{R}_V(\mathbf{X}, \boldsymbol{\theta}),$$

- The variance is estimated by the difference of predictions under perturbations $\boldsymbol{\xi}$ and $\boldsymbol{\xi}'$

$$R_V(\mathbf{X}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}', \boldsymbol{\xi}} \, \|\mathbf{f}(\mathbf{x}_i;\boldsymbol{\theta}, \boldsymbol{\xi}') - \mathbf{f}(\mathbf{x}_i;\boldsymbol{\theta}, \boldsymbol{\xi})\|^2,$$

- The regularizer is an unbiased estimator of the Jacobian norm with bounded variance

The Estimator of $\boldsymbol{R_V}$

i.i.d. r.v. $\boldsymbol{\xi}, \boldsymbol{\xi}'$,

$$R_V(\mathbf{X}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}', \boldsymbol{\xi}} \|\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}') - \mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi})\|^2$$

$$= \frac{1}{N} \sum_{i=1}^{N} \Big[\mathbb{E}_{\boldsymbol{\xi}'} \|\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}')\|^2 + \mathbb{E}_{\boldsymbol{\xi}} \|\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi})\|^2$$

$$- 2\mathbb{E}_{\boldsymbol{\xi}'} \mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}')^\top \mathbb{E}_{\boldsymbol{\xi}} \mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}) \Big] \quad (6)$$

$$= \frac{1}{N} \sum_{i=1}^{N} \Big[2\mathbb{E}_{\boldsymbol{\xi}} \|\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi})\|^2 - 2\|\mathbb{E}_{\boldsymbol{\xi}} \mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi})\|^2 \Big] \quad (7)$$

$$= \frac{2}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} \mathrm{Var}_{\boldsymbol{\xi}} [\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi})]_k \quad (8)$$

$$\hat{R}_V(\mathbf{X}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}_i') - \mathbf{f}(\mathbf{x}_i; \boldsymbol{\theta}, \boldsymbol{\xi}_i)\|^2,$$

$$\mathbb{E}_{\boldsymbol{\xi}, \boldsymbol{\xi}'} \hat{R}_V(\mathbf{X}, \boldsymbol{\theta}) = R_V(\mathbf{X}, \boldsymbol{\theta}).$$

Unbiased Estimator of Jacobian

- A simplified version to analyze is to assume ***ξ, ξ'*** are i.i.d. sampled from a Gaussian distribution

$$\boldsymbol{\xi}, \boldsymbol{\xi}' \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_D)$$

- The variance term implicitly estimates the Jacobian norm $\|\mathbf{J}(\mathbf{x})\|_F$. of the neural network

$$\frac{1}{\sigma^2} R_V(\mathbf{X}, \boldsymbol{\theta}) = \frac{1}{\sigma^2} \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}', \boldsymbol{\xi}} \|\mathbf{f}(\mathbf{x}_i + \boldsymbol{\xi}; \boldsymbol{\theta}) - \mathbf{f}(\mathbf{x}_i + \boldsymbol{\xi}'; \boldsymbol{\theta})\|^2$$

- By fifirst-order Taylor expansion, and let $\mathbf{J}(\mathbf{x}) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}$

$$\mathbf{f}(\mathbf{x} + \boldsymbol{\xi}) = \mathbf{f}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\boldsymbol{\xi} + o(\boldsymbol{\xi}),$$

Unbiased Estimator of Jacobian

$$
\begin{aligned}
&\frac{1}{\sigma^2} R_V(\mathbf{X}, \boldsymbol{\theta}) \\
=&\frac{1}{\sigma^2}\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}} \|\mathbf{J}(\mathbf{x}_i)\boldsymbol{\xi}\|^2 + \mathbb{E}_{\boldsymbol{\xi}'} \|\mathbf{J}(\mathbf{x}_i)\boldsymbol{\xi}'\|^2 \\
&- 2\mathbb{E}_{\boldsymbol{\xi}} \boldsymbol{\xi}^\top \mathbf{J}(\mathbf{x}_i)^\top \mathbf{J}(\mathbf{x}_i) \mathbb{E}_{\boldsymbol{\xi}} \boldsymbol{\xi}' \\
=&2\frac{1}{\sigma^2}\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}} \|\mathbf{J}(\mathbf{x}_i)\boldsymbol{\xi}\|^2 - [\|\mathbb{E}_{\boldsymbol{\xi}} \mathbf{J}(\mathbf{x}_i)\boldsymbol{\xi}\|]^2 \\
=&2\frac{1}{\sigma^2}\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\xi}} [\boldsymbol{\xi}^\top \mathbf{J}(\mathbf{x}_i)^\top \mathbf{J}(\mathbf{x}_i)\boldsymbol{\xi}] - 0 \\
=&2\frac{1}{N}\sum_{i=1}^{N} \mathrm{Tr}\left[\mathbf{J}(\mathbf{x}_i)^\top \mathbf{J}(\mathbf{x}_i)\frac{1}{\sigma^2}\mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{\xi}\boldsymbol{\xi}^\top]\right] \\
=&2\frac{1}{N}\sum_{i=1}^{N} \mathrm{Tr}\left[\mathbf{J}(\mathbf{x}_i)^\top \mathbf{J}(\mathbf{x}_i)\frac{1}{\sigma^2}\sigma^2\mathbf{I}_D\right] \\
=&2\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{J}(\mathbf{x}_i)\|_F^2.
\end{aligned}
$$

If we further take expectation over N samples of $\boldsymbol{x}_i$

$$\mathbb{E}_{\mathbf{X}} \frac{1}{\sigma^2} R_V(\mathbf{X}, \boldsymbol{\theta}) = 2\frac{1}{N} N \mathbb{E}_{\mathbf{x}} \|\mathbf{J}(\mathbf{x})\|_F^2 = 2\mathbb{E}_{\mathbf{x}} \|\mathbf{J}(\mathbf{x})\|_F^2.$$

$$\mathbb{E}_{\boldsymbol{\xi},\boldsymbol{\xi}'} \hat{R}_V(\mathbf{X}, \boldsymbol{\theta}) \;=\; R_V(\mathbf{X}, \boldsymbol{\theta}).$$

Then we can prove that $\frac{1}{2\sigma^2}\hat{R}_V(\mathbf{X}, \boldsymbol{\theta})$ is an unbiased estimator of $\mathbb{E}_{\mathbf{x}} \|\mathbf{J}(\mathbf{x})\|_F^2$

$$\mathbb{E}_{\mathbf{X},\boldsymbol{\xi}} \frac{1}{2\sigma^2} \hat{R}_V(\mathbf{X}, \boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x}} \|\mathbf{J}(\mathbf{x})\|_F^2.$$

Analyze the Reliability of this Estimator

- An estimator with low variance will give more confidence in its usage, which means more reliable.

$$\mathbf{A} = \mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x}) \quad \mathbf{z} = \frac{1}{\sigma}\boldsymbol{\xi}$$

$$\mathrm{Tr}(\mathbf{A}) = \mathbb{E}_{\mathbf{z}}[\mathbf{z}^\top \mathbf{A}\mathbf{z}]$$

- We first derive the variance of $\mathbf{z^T A z}$

$$\text{Let } \mathbf{a} = diag(\mathbf{A}), \mathbf{m} = \mathbb{E}\mathbf{z} = \mathbf{0},$$

$$\mathrm{Var}[\mathbf{z}^\top \mathbf{A}\mathbf{z}] = 2\mu_2^2 \,\mathrm{Tr}(\mathbf{A}\mathbf{A}^\top) + 4\mu_2 \mathbf{m}^\top \mathbf{A}\mathbf{m} + 4\mu_3 \mathbf{m}^\top \mathbf{A}\mathbf{a} + (\mu_4 - 3\mu_2^2)\mathbf{a}^\top \mathbf{a}$$

$$\mathrm{Var}[\mathbf{z}^\top \mathbf{A}\mathbf{z}] = 2\|\mathbf{A}\|_F^2 = 2\|\mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x})\|_F^2.$$

Analyze the Reliability of this Estimator

- And then we derive the variance of $\frac{1}{2\sigma^2}\hat{R}_V(\mathbf{X}, \boldsymbol{\theta})$ by

$$\frac{1}{2\sigma^2}\hat{R}_V(\mathbf{X}, \boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{z}}[\mathbf{z}^\top \mathbf{A}\mathbf{z}]$$

- And the law of total variance.

$$\mathrm{Var}(\mathbf{y}) = \mathbb{E}[\mathrm{Var}(\mathbf{y}|\mathbf{x})] + \mathrm{Var}[\mathbb{E}(\mathbf{y}|\mathbf{x})]$$

- Treat $\frac{1}{2\sigma^2}\hat{R}_V(\mathbf{X}, \boldsymbol{\theta})$ as $\boldsymbol{y}$ then we get

$$\mathrm{Var}_{\mathbf{x},\boldsymbol{\xi}}[\frac{1}{2\sigma^2}\hat{R}_V(\mathbf{X}, \boldsymbol{\theta})] = 2\mathbb{E}_{\mathbf{x}}\|\mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x})\|_F^2 + \mathrm{Var}_{\mathbf{x}}[\|\mathbf{J}(\mathbf{x})\|_F^2]].$$

Experiments

- Input-agnostic uniform label noise

- a certain percentage η (0%, 20%, 40%,60%, 80%) of true labels on the training dataset are replaced by random labels through uniform sampling

Experiments

– Input-agnostic uniform label noise

Table 1: Averaged test error rates (%) and the standard deviations over 3 runs on CIFAR-10 under different uniform noise fraction. Methods marked with † are trained using additional clean validation images. Best results are highlighted in bold.

Methods	Noise Ratio η					Network
	0	0.2	0.4	0.6	0.8	
Bootstrap-hard [23]	10.94 ± 0.9	20.81 ± 0.4	23.33 ± 0.8	29.43 ± 0.3	–	12-layer CNN
Forward-correction [22]	9.73 ± 0.0	15.39 ± 0.3	18.16 ± 0.1	27.59 ± 0.7	–	12-layer CNN
D2L [17]	10.59 ± 0.2	14.87 ± 0.6	16.64 ± 0.5	27.16 ± 0.6	–	12-layer CNN
Generalized Cross Entropy [35]	6.5	10.13 ± 0.2	12.87 ± 0.22	17.46 ± 0.23	32.08 ± 0.6	ResNet-34
Co-teaching [7]	6.05	17.68	–	–	–	13-layer CNN
MentorNet [12]†	4	8	11	–	51	WRN-101-10
Learning to reweight [24]†	3.87	–	13.08 ± 0.19	–	–	WRN-28-10
Ours	3.79 ± 0.13	$\mathbf{3.87 \pm 0.15}$	$\mathbf{5.05 \pm 0.24}$	$\mathbf{6.42 \pm 0.28}$	$\mathbf{13.31 \pm 0.45}$	WRN-28-10

Experiments

– Input-agnostic uniform label noise

Table 2: Test error rates (%) on CIFAR-100 under different uniform noise fraction. Methods marked with † are trained using additional clean validation images. Best results are highlighted in bold.

Methods	Noise Ratio η					Network
	0	0.2	0.4	0.6	0.8	
Bootstrap-hard [23]	31.69 ± 0.2	41.51 ± 0.4	53.56 ± 0.7	57.35 ± 0.9	–	ResNet-44
Forward-correction [22]	31.46 ± 0.1	39.75 ± 0.2	48.73 ± 0.3	55.78 ± 0.7	–	ResNet-44
D2L [17]	31.40 ± 0.3	37.80 ± 0.5	46.99 ± 0.7	54.79 ± 0.4	–	ResNet-44
Generalized Cross Entropy [35]	28.6	33.19 ± 0.42	38.23 ± 0.24	45.96 ± 0.56	52.34 ± 0.69	ResNet-34
Co-teaching [7]	29.15	45.77	–	–	–	13-layer CNN
MentorNet [12]†	21	27	32	–	65	WRN-101-10
Learning to reweight [24]†	21.8	–	38.66 ± 2.06	–	–	WRN-28-10
Ours	18.6±0.15	**19.45 ±0.22**	**25.73 ± 0.47**	**38.23 ±0.52**	**44.68 ± 0.75**	WRN-28-10

Experiments

- Input-agnostic uniform label noise

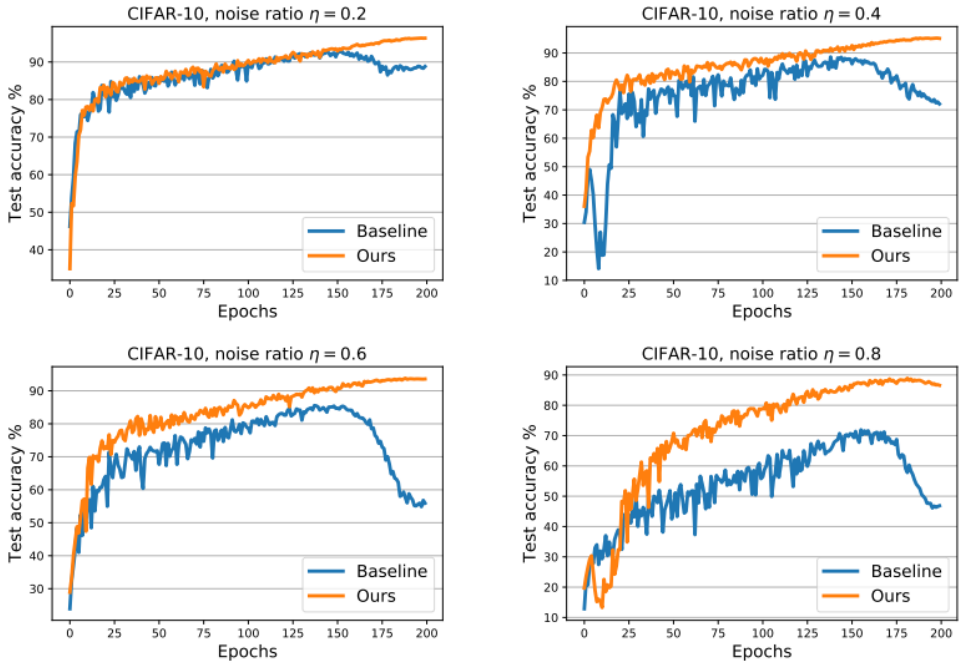


Figure 1: Test accuracy against the number of epochs on CIFAR-10 under different uniform noise ratio trained with WRN-28-10. Our method is less prone to the label noise over-fitting.

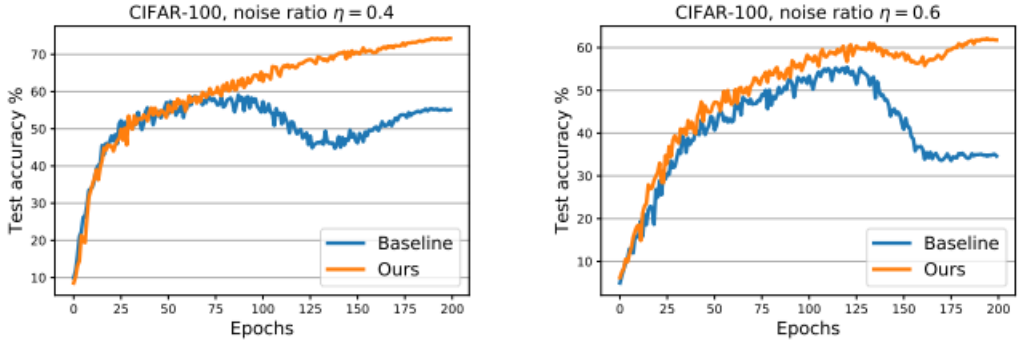


Figure 2: Test accuracy against the number of epochs on CIFAR-100 under different uniform noise ratios trained with WRN-28-10. Our method is less prone to label noise over-fitting.

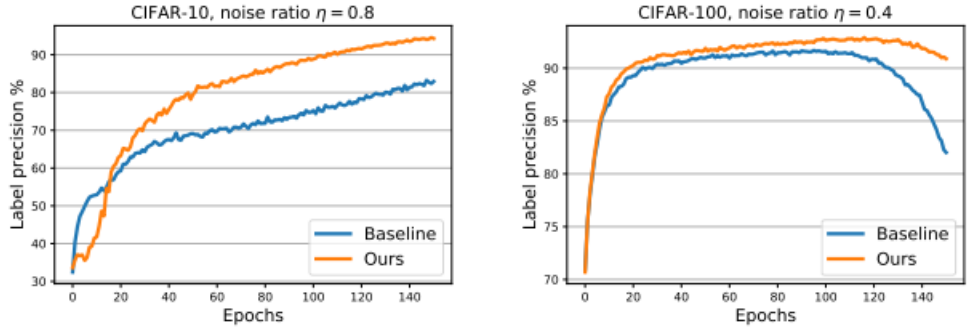


Figure 3: Label precision against the number of epochs on CIFAR-10 (left) and CIFAR-100 (right) with uniform noise, respectively. Here the label precision is computed by the percentage of clean training examples within those having $1 - \eta$ minimal training losses.

Experiments

– Class-dependent asymmetric label noise

- A more realistic and more challenging noise type than the uniform noise is to corrupt between the semantically similar classes.

Table 3: Results on CIFAR-10 and CIFAR-100 with class-dependent asymmetric noise. Averaged accuracy and standard deviation over 3 runs are reported. The results of competing methods are taken from [35]. CCE stands for commonly-used categorical cross-entropy loss function, MAE stands for mean absolute error. Forward $T^\dagger$ [22] uses the ground-truth noise transition matrix while Forward $\hat{T}$ [22] estimates T. Comparison to Forward $T^\dagger$ is not fair. Trunc $\mathcal{L}_q$ loss is a noise-robust loss function proposed in [35].

Datasets	Methods	Noise Ratio η			
		0.1	0.2	0.3	0.4
CIFAR-10	CCE	90.69 ± 0.17	88.59 ± 0.34	86.14 ± 0.40	80.11 ± 1.44
	MAE	82.61 ± 4.81	52.93 ± 3.60	50.36 ± 5.55	45.52 ± 0.13
	Forward $T^\dagger$ [22]	91.32 ± 0.21	90.35 ± 0.26	89.25 ± 0.43	$\mathbf{88.12 \pm 0.32}$
	Forward $\hat{T}$ [22]	90.52 ± 0.26	89.09 ± 0.47	86.79 ± 0.36	83.55 ± 0.58
	Trunc $\mathcal{L}_q$ [35]	90.43 ± 0.25	89.45 ± 0.29	87.10 ± 0.22	82.28 ± 0.67
	Baseline (CCE)	94.31 ± 0.19	90.29 ± 0.35	84.61 ± 0.41	78.24 ± 0.82
	Ours	$\mathbf{95.69 \pm 0.18}$	$\mathbf{94.01 \pm 0.22}$	$\mathbf{92.44 \pm 0.37}$	$\mathbf{85.62 \pm 0.77}$
CIFAR-100	CCE	66.54 ± 0.42	59.20 ± 0.18	51.40 ± 0.16	42.74 ± 0.61
	MAE	13.38 ± 1.84	11.50 ± 1.16	8.91 ± 0.89	8.20 ± 1.04
	Forward $T^\dagger$ [22]	71.05 ± 0.30	71.08 ± 0.22	70.76 ± 0.26	70.82 ± 0.45
	Forward $\hat{T}$ [22]	45.96 ± 1.21	42.46 ± 2.16	38.13 ± 2.97	34.44 ± 1.93
	Trunc $\mathcal{L}_q$ [35]	68.86 ± 0.14	66.59 ± 0.23	61.87 ± 0.39	47.66 ± 0.69
	Baseline (CCE)	79.40 ± 0.22	73.50 ± 0.21	63.02 ± 0.32	52.06 ± 0.71
	Ours	$\mathbf{82.55 \pm 0.24}$	$\mathbf{82.34 \pm 0.20}$	$\mathbf{80.55 \pm 0.26}$	$\mathbf{74.54 \pm 0.64}$

Experiments

- Class-dependent asymmetric label noise

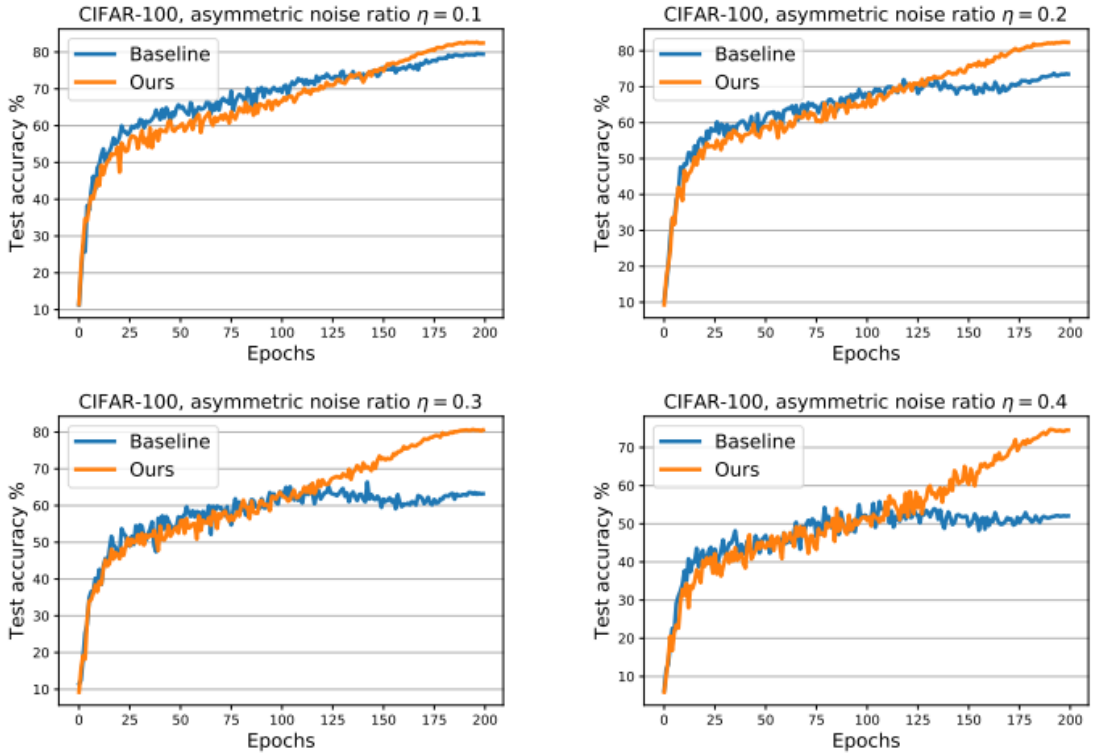


Figure 4: Test accuracy against the number of epochs on CIFAR-100 under different asymmetric noise ratios trained with WRN-28-10.

Experiments
-large-scale datasets

- **ImageNet-2012：**

Table 4: Results on the clean Imagenet validation set trained using ResNet-50. Top-1 (Top-5) error rates are listed.

Methods	Noise ratio η		
	0	0.2	0.4
CCE	23.45 (6.78)	26.41 (8.62)	29.79 (10.58)
MentorNet [12]	–	–	34.9 (14.1)
Ours	**23.27 (6.69)**	**24.83 (7.74)**	**26.81 (8.99)**

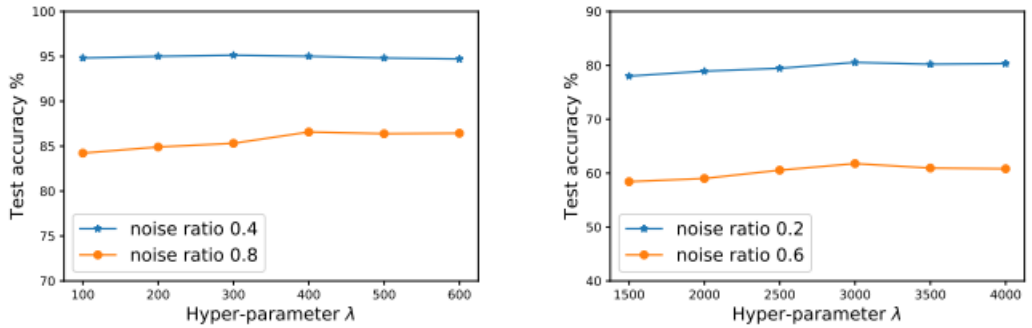
- **WebVision：**

Table 5: Results on the clean Webvision validation set and ImageNet validation set. The model is trained on noisy Webvision training data.

Method	Webvision		Imagenet	
	Top-1	Top-5	Top-1	Top-5
Li *et al.* [15]	43.0	22.1	52.4	29.6
Lee *et al.* [14]	31.5	13.5	39.8	18.9
MentorNet [12]	29.2	12.0	37.5	17.0
Ours	**27.3**	**10.5**	**34.1**	**14.25**

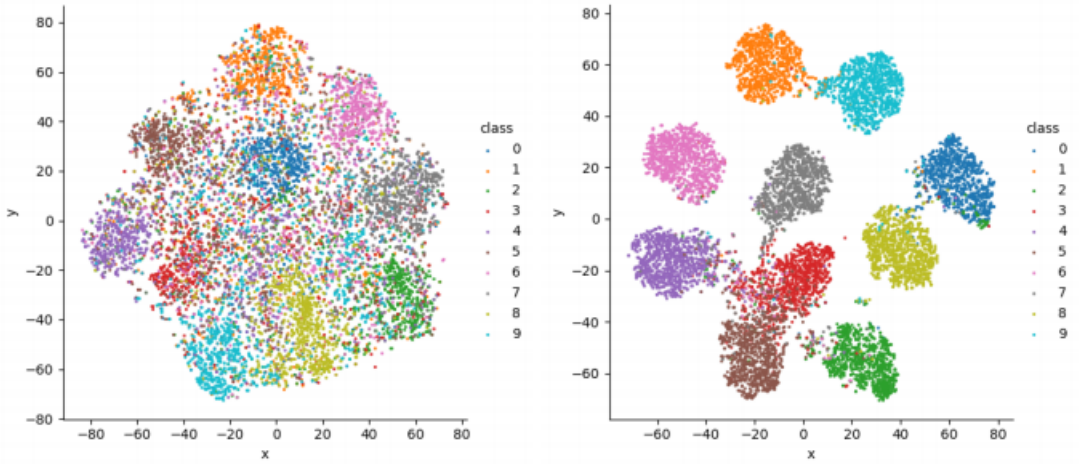
Experiments

-Hyper-parameter sensitivity analysis & Visualization



(a) CIFAR-10 (b) CIFAR-100

Figure 7: Hyper-parameter sensitivity analysis on CIFAR-10 and CIFAR-100 with various strengths of label noise . Our method is insensitive to a wide range of values for λ.



(a) CCE (b) Ours

Figure 8: t-SNE 2D embeddings of the test dataset on CIFAR-10 trained with 60% uniform label noise. Each color represents a class. Our method in (b) learns a more separable feature space than CCE.