Learning under Noisy Supervision

Part II: Statistical Learning under Noisy Supervision

Tongliang Liu

Trustworthy Machine Learning Lab
School of Computer Science
University of Sydney

Structure

Basics

Consistent algorithms

Transition matrix

Structure

Basics

Consistent algorithms

Transition matrix

Learning without noisy labels

Problem setup:

Data: $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\} \sim D^n$.

Aim: Learn a classifier $f \in F$, such that $\forall(\boldsymbol{x}, y) \sim D$, $f(\boldsymbol{x})$ is a good prediction for y.

What is the best classifier we can obtain?

w.r.t. accuracy

To measure the accuracy, we define loss function

$$\ell(\boldsymbol{x}, y), f \mapsto \ell\big(y, f(\boldsymbol{x})\big) \in \mathbb{R}.$$

For example, 0-1 loss: $\mathbf{1}(y \neq \operatorname{sign}(f(\boldsymbol{x})))$.

The best classifier should be the one that has the minimum loss on all the possible data from the domain.

Expected risk, Bayes classifier

Theoretically,

The expected risk:

$$R_{D,0-1}(f) = \mathbb{E}_{(X,Y)\sim D}[\mathbf{1}(Y \neq \operatorname{sign}(f(X)))].$$

Bayes classifier:

$$f_\rho(\boldsymbol{x}) = \operatorname{argmin}_f R_{D,0-1}(f) = \arg\max_y P(Y = y | X = \boldsymbol{x}).$$

Restricted Bayes classifier:

$$f^* = \operatorname*{argmin}_{f\in\mathcal{F}} R_{D,0-1}(f).$$

Empirically,

In reality, we can only observe a sample of data $S = \{(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n)\} \sim D^n$.

We approximate the expected risk $R(f)$ via the empirical risk: $\widehat{R}_{D,\ell}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(y_i, f(\boldsymbol{x}_i)\big)$.

We minimize the empirical risk to find a predictor:

$$f_n = \arg\min_{f \in \boldsymbol{\mathcal{F}}} \widehat{R}_{D,\ell}(f).$$

Statistically consistent classifier [1,2]:

With high probability, as $n \longrightarrow \infty$,
we have: $R_{D,\ell}(f_n) \longrightarrow R_{D,\ell}(f^*)$.

[1] Mohri et al. *Foundations of machine learning*. MIT press, 2018.
[2] Devroye, et al. *A probabilistic theory of pattern recognition*. Vol. 31. Springer Science & Business Media, 2013.

Aim:

Designing algorithms whose outputs will approach
$f_\rho(\boldsymbol{x}) = \arg\max_y P(Y = y|X = \boldsymbol{x})$.

Structure

Basics

Consistent algorithms

Transition matrix

Learning with noisy labels

Noisy sample: $\tilde{S} = \{(\boldsymbol{x}_1, \tilde{y}_1), \ldots, (\boldsymbol{x}_n, \tilde{y}_n)\} \sim \widetilde{D}^n$,
where $\tilde{y}$ stands for noisy labels and $\widetilde{D}$ the noisy distribution.

What is the best classifier we can learn?

Can we approach $f_\rho(\boldsymbol{x}) = \arg\max_y P(Y = y | X = \boldsymbol{x})$?

Learning with noisy labels

One category: extracting confident examples.

SOTA, e.g., Co-teaching [3]; Joint Optim [4].

Another category: label-noise learning [5].

Methodology, i.e., statistically consistent algorithms.

[3] Han, Bo, et al. "Co-teaching: Robust training of deep neural networks with extremely noisy labels." *NeurIPS* 2018.
[4] Tanaka, Daiki, et al. "Joint optimization framework for learning with noisy labels." *CVPR* 2018.
[5] Xia, Xiaobo, et al. “Are anchor points really indispensable in label-noise learning?.” *NeurIPS* 2019.

Learning with noisy labels

One category: extracting confident examples.

SOTA, e.g., Co-teaching [3]; Joint Optim [4].

Another category: label-noise learning [5].

Methodology, i.e., statistically consistent algorithms.

[3] Han, Bo, et al. "Co-teaching: Robust training of deep neural networks with extremely noisy labels." *NeurIPS* 2018.
[4] Tanaka, Daiki, et al. "Joint optimization framework for learning with noisy labels." *CVPR* 2018.
[5] Xia, Xiaobo, et al. “Are anchor points really indispensable in label-noise learning?.” *NeurIPS* 2019.

Model label noise

Transition matrix:

$$\begin{bmatrix} P(\tilde{Y} = 1 | Y = 1, \boldsymbol{x}) & \cdots & P(\tilde{Y} = 1 | Y = C, \boldsymbol{x}) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y} = C | Y = 1, \boldsymbol{x}) & \cdots & P(\tilde{Y} = C | Y = C, \boldsymbol{x}) \end{bmatrix}.$$

Transition matrix

$$\begin{bmatrix} P(\tilde{Y} = 1|\boldsymbol{x}) \\ \vdots \\ P(\tilde{Y} = C|\boldsymbol{x}) \end{bmatrix} = \underbrace{\boxed{\begin{bmatrix} P(\tilde{Y} = 1|Y = 1, \boldsymbol{x}) & \cdots & P(\tilde{Y} = 1|Y = C, \boldsymbol{x}) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y} = C|Y = 1, \boldsymbol{x}) & \cdots & P(\tilde{Y} = C|Y = C, \boldsymbol{x}) \end{bmatrix}}}_{T^\top(\boldsymbol{x})} \begin{bmatrix} P(Y = 1|\boldsymbol{x}) \\ \vdots \\ P(Y = C|\boldsymbol{x}) \end{bmatrix}$$

Why called "label-noise learning"?

- Label-noise learning [5]
- Learning with noisy labels [6]

[5] Xia, Xiaobo, et al. "Are anchor points really indispensable in label-noise learning?." *NeurIPS* 2019.
[6] Natarajan, Nagarajan, et al. "Learning with noisy labels." *NeurIPS* 2013.

Model Label Noise

(1) Random Classification Noise (RCN) [7]:

$$\rho_{\tilde{Y},Y}(X) = P\left(\tilde{Y}\middle|Y,X\right)=P\left(\tilde{Y}\middle|Y\right) = \rho, \forall\, Y \neq \tilde{Y}.$$

(2) Class-conditional Noise (CCN) [6]:

$$\rho_{\tilde{Y},Y}(X) = P\left(\tilde{Y}\middle|Y,X\right)=P\left(\tilde{Y}\middle|Y\right).$$

(3) Instance-dependent Noise (IDN) [8,9]:

$$\rho_{\tilde{Y},Y}(X) = P\left(\tilde{Y}\middle|Y,X\right).$$

[7] Angluin, Dana, and Philip Laird. “Learning from noisy examples.” *Machine Learning* 2.4: 343-370, 1988.
[8] Cheng, Jiacheng, et al. “Learning with bounded instance and label-dependent label noise.” *ICML* 2020.
[9] Berthon, Antonin, et al. "Confidence scores make instance-dependent label-noise learning possible." *ICML*, 2021

Model Label Noise

(1) Random Classification Noise (RCN) [7]:

$$\rho_{\tilde{Y},Y}(X) = P(\tilde{Y}|Y,X) = P(\tilde{Y}|Y) = \rho, \forall\, Y \neq \tilde{Y}.$$

(2) Class-conditional Noise (CCN) [6]:

$$\rho_{\tilde{Y},Y}(X) = P(\tilde{Y}|Y,X) = P(\tilde{Y}|Y).$$

(3) Instance-dependent Noise (IDN) [8,9]:

$$\rho_{\tilde{Y},Y}(X) = P(\tilde{Y}|Y,X).$$

[7] Angluin, Dana, and Philip Laird. “Learning from noisy examples.” *Machine Learning* 2.4: 343-370, 1988.
[8] Cheng, Jiacheng, et al. “Learning with bounded instance and label-dependent label noise.” *ICML* 2020.
[9] Berthon, Antonin, et al. "Confidence scores make instance-dependent label-noise learning possible." *ICML*, 2021

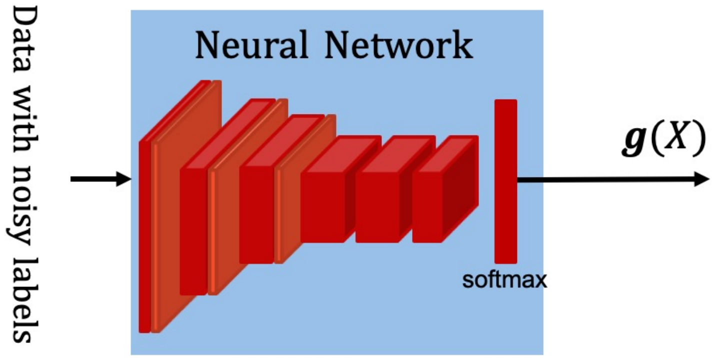
Class-conditional Noise (CCN)

The loss correction method:
Modify ℓ to be $\tilde{\ell}$ such that

$$\mathbb{E}_{(X,\tilde{Y})\sim\tilde{D}}\left[\tilde{\ell}\left(f(X),\tilde{Y}\right)\right] = \mathbb{E}_{(X,Y)\sim Y}\left[\ell\left(f(X),Y\right)\right]$$

By exploiting the model of label noise:

$$\begin{bmatrix} P(\tilde{Y}=1|\boldsymbol{x}) \\ \vdots \\ P(\tilde{Y}=C|\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C) \end{bmatrix} \begin{bmatrix} P(Y=1|\boldsymbol{x}) \\ \vdots \\ P(Y=C|\boldsymbol{x}) \end{bmatrix}$$



$$\begin{bmatrix} P(\tilde{Y}=1|\boldsymbol{x}) \\ \vdots \\ P(\tilde{Y}=C|\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C) \end{bmatrix} \begin{bmatrix} P(Y=1|\boldsymbol{x}) \\ \vdots \\ P(Y=C|\boldsymbol{x}) \end{bmatrix}$$

Importance reweighting [11]:

$$\tilde{\ell}_{ir}(f(\boldsymbol{x}), y) = \frac{P(\boldsymbol{x}, y)}{\tilde{P}(\boldsymbol{x}, y)} \ell(f(\boldsymbol{x}), y) = \frac{\boldsymbol{g}_y(\boldsymbol{x})}{(T^\top \boldsymbol{g})_y(\boldsymbol{x})} \ell(f(\boldsymbol{x}), y),$$

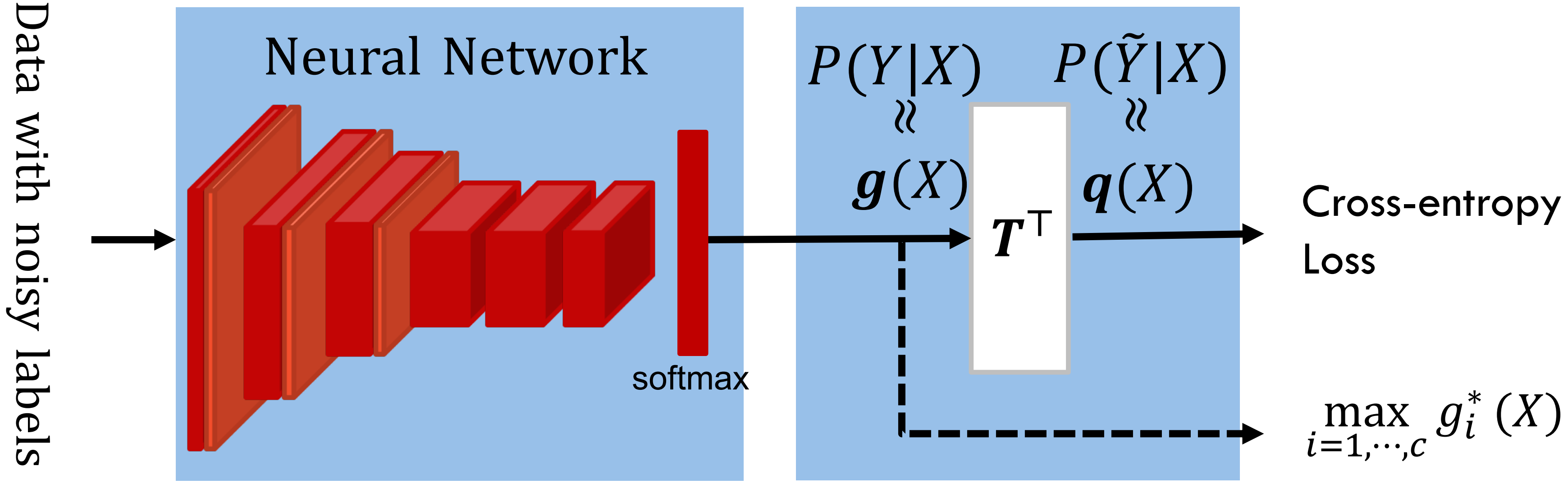
where $f(\boldsymbol{x}) = \arg \max_{j \in \{1,\ldots,C\}} \boldsymbol{g}_j(\boldsymbol{x})$.

Thus, $\mathbb{E}_{(X,\tilde{Y})\sim\tilde{D}}\left[\tilde{\ell}_{ir}\left(f(X), \tilde{Y}\right)\right] = \mathbb{E}_{(X,Y)\sim D}[\ell(f(X), Y)]$

[11] Liu, Tongliang, and Dacheng Tao. "Classification with noisy labels by importance reweighting." *IEEE Transactions on pattern analysis and machine intelligence* 38.3 (2015): 447-461..

$$\begin{bmatrix} P(\tilde{Y}=1|\boldsymbol{x}) \\ \vdots \\ P(\tilde{Y}=C|\boldsymbol{x})\end{bmatrix} = \begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C)\end{bmatrix}\begin{bmatrix}P(Y=1|\boldsymbol{x})\\ \vdots \\ P(Y=C|\boldsymbol{x})\end{bmatrix}$$

Forward correction [12]:



[12] Patrini, Giorgio, et al. "Making deep neural networks robust to label noise: A loss correction approach." *CVPR* 2017.

A summary of consistent algorithms

- Many methods for dealing with noisy labels
  Loss correction, Sample selection, label correction, …

- Model label noise
  Random Classification Noise (RCN)
  Class-conditional Noise (CCN)
  Instance-dependent Noise (IDN)

- Three loss correction methods
  Importance reweighting (risk-consistent), forward correction (classifier-consistent)

Structure

Basics

Consistent algorithms

Transition matrix

How to estimate the transition matrix

Given the noisy data
$\tilde{S} = \{(\boldsymbol{x}_1, \tilde{y}_1), \ldots, (\boldsymbol{x}_n, \tilde{y}_n)\} \sim \widetilde{D}$.

How to estimate the transition matrix T?

Definition

If $P\left(Y=i\middle|\boldsymbol{x}^i\right)=1$, then $\boldsymbol{x}^i$ is called the anchor point for the i-th class.

[11] Liu, Tongliang, and Dacheng Tao. "Classification with noisy labels by importance reweighting." *IEEE Transactions on pattern analysis and machine intelligence* 38.3 (2015): 447-461..

Anchor point assumption

$$\begin{bmatrix} P(\tilde{Y}=1|X) \\ \vdots \\ P(\tilde{Y}=C|X) \end{bmatrix} = \underset{T}{\boxed{\begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C) \end{bmatrix}}} \begin{bmatrix} P(Y=1|X) \\ \vdots \\ P(Y=C|X) \end{bmatrix}$$

$$\begin{bmatrix} P(\tilde{Y}=1|X=\boldsymbol{x}^1) \\ \vdots \\ P(\tilde{Y}=C|X=\boldsymbol{x}^1) \end{bmatrix} = \begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} P(\tilde{Y}=1|X=\boldsymbol{x}^i) \\ \vdots \\ P(\tilde{Y}=C|X=\boldsymbol{x}^i) \end{bmatrix} = \begin{bmatrix} P(\tilde{Y}=1|Y=i) \\ \vdots \\ P(\tilde{Y}=C|Y=i) \end{bmatrix}$$

[11] Liu, Tongliang, and Dacheng Tao. "Classification with noisy labels by importance reweighting." *IEEE Transactions on pattern analysis and machine intelligence* 38.3 (2015): 447-461..

How to find anchor points

Binary classification, find the anchor points:

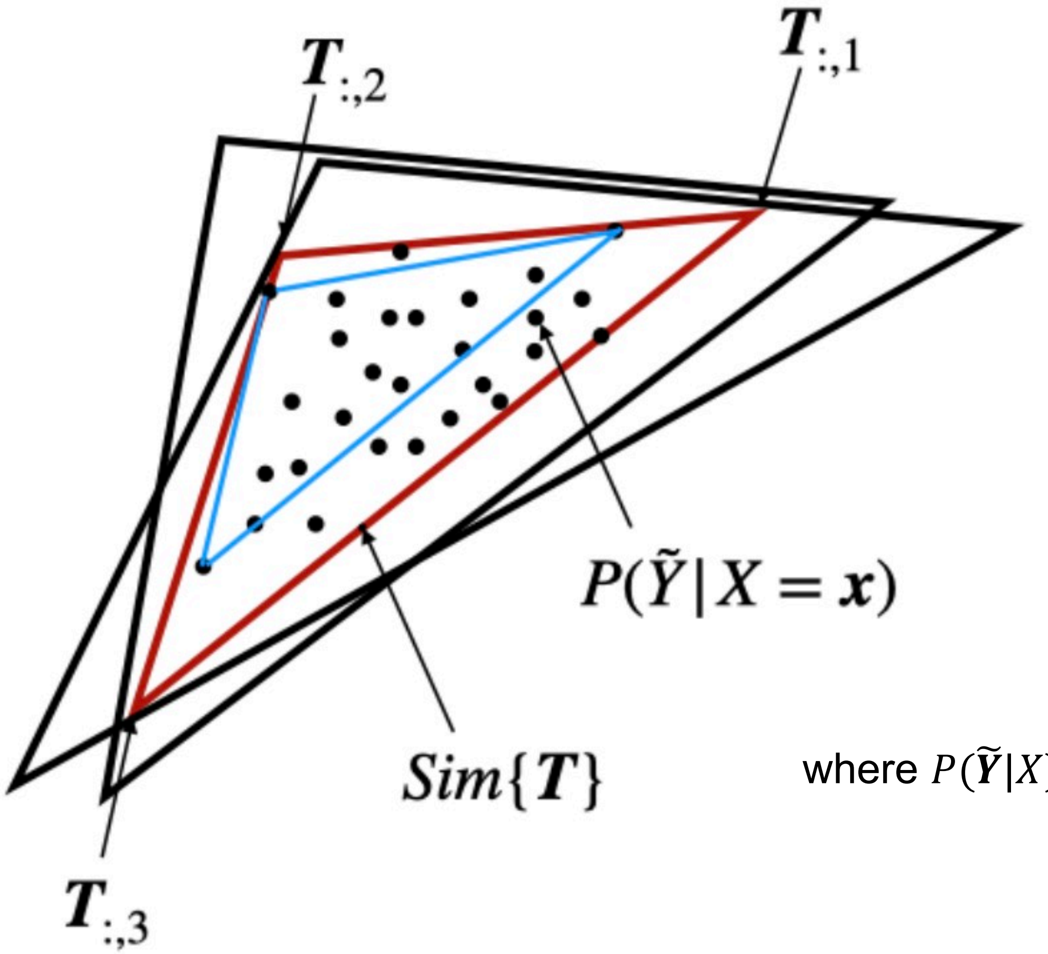
$$\boldsymbol{x}^y = \underset{\boldsymbol{x} \in X}{\operatorname{argm}ax}\, P(\tilde{Y} = y | \boldsymbol{x}).$$

Multi-classification, approximate the anchor points for multi-class learning:

$$\boldsymbol{x}^y \approx \underset{\boldsymbol{x} \in X}{\operatorname{argm}ax}\, P(\tilde{Y} = y | \boldsymbol{x}).$$

[11] Liu, Tongliang, and Dacheng Tao. "Classification with noisy labels by importance reweighting." *IEEE Transactions on pattern analysis and machine intelligence* 38.3 (2015): 447-461..

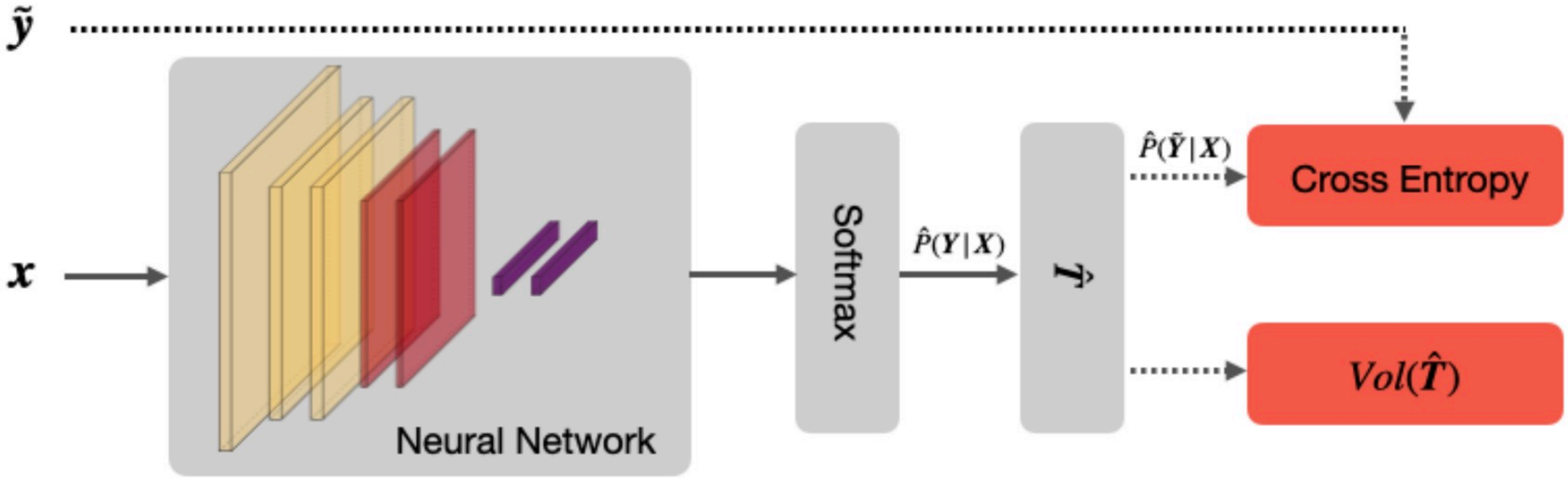
Sufficiently scattered assumption vs anchor point assumption



where $P(\tilde{\boldsymbol{Y}}|X) = \begin{bmatrix} P(\tilde{Y}=1|X) \\ \vdots \\ P(\tilde{Y}=C|X) \end{bmatrix} = \underbrace{\boxed{\begin{bmatrix} P(\tilde{Y}=1|Y=1) & \cdots & P(\tilde{Y}=1|Y=C) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y}=C|Y=1) & \cdots & P(\tilde{Y}=C|Y=C) \end{bmatrix}}}_{T} \begin{bmatrix} P(Y=1|X) \\ \vdots \\ P(Y=C|X) \end{bmatrix}$

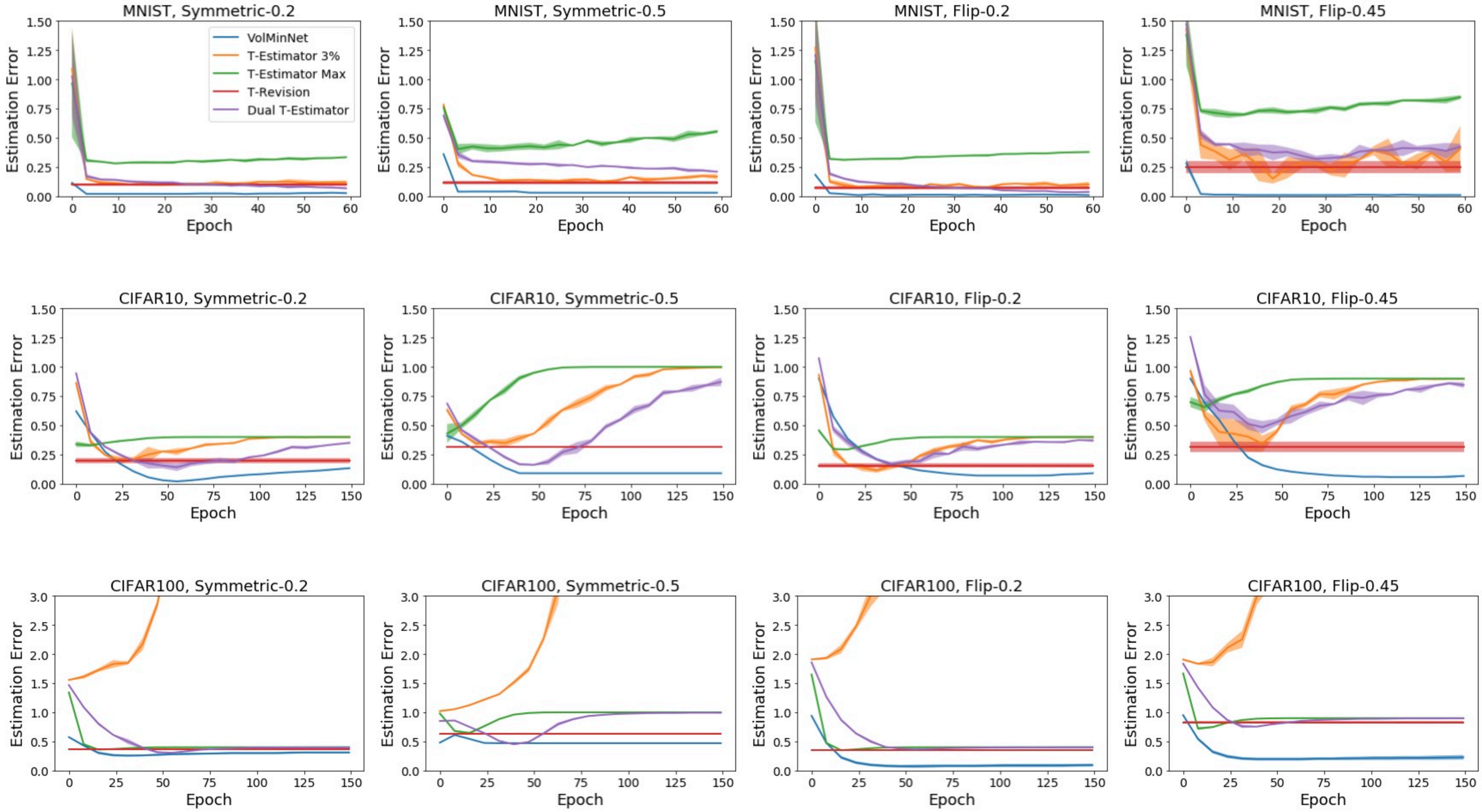
[14] Li, Xuefeng, et al. "Provably end-to-end label-noise learning without anchor points." *ICML* 2021.

VolMinNet [14]



$$\min_{\hat{T}\in\mathbb{T}} \mathrm{vol}\,(\hat{T})$$
$$\text{s.t.}\, \hat{T}h_\theta = P(\widetilde{\boldsymbol{Y}}|X)$$

[14] Li, Xuefeng, et al. "Provably end-to-end label-noise learning without anchor points." *ICML* 2021.



[14] Li, Xuefeng, et al. "Provably end-to-end label-noise learning without anchor points." *ICML* 2021.

T revision [15]

$$T = \begin{bmatrix} 0.5 & 0.3 & 0.2 & & \\ & 0.5 & .25 & .25 & \\ & & 0.5 & 0.3 & 0.2 \\ 0.2 & & & 0.5 & 0.3 \\ 0.3 & 0.2 & & & 0.5 \end{bmatrix} \qquad \tilde{T} = \begin{bmatrix} 0.5 & 0.3 & 0.2 & & \\ & 0.5 & 0.3 & 0.2 & \\ & & 0.5 & 0.3 & 0.2 \\ 0.2 & & & 0.5 & 0.3 \\ 0.3 & 0.2 & & & 0.5 \end{bmatrix}$$

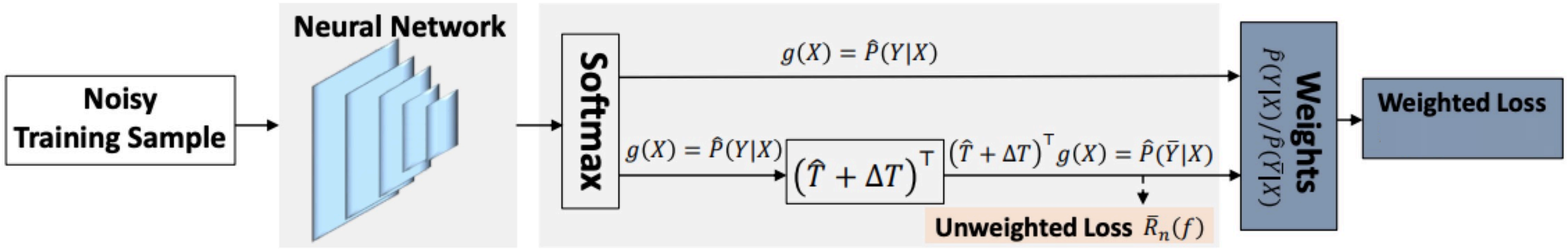
If $P(\widetilde{\boldsymbol{Y}}|X = \boldsymbol{x}) = [0.141; 0.189; 0.239; 0.281; 0.15]$,

then, $P(\boldsymbol{Y}|X = \boldsymbol{x}) = (T^\top)^{-1} P(\widetilde{\boldsymbol{Y}}|X = \boldsymbol{x}) =$
$[0.15; 0.28; 0.25; \boxed{0.3;}\, 0.02]$.

$$P(\boldsymbol{Y}|X = \boldsymbol{x}) = (\tilde{T}^\top)^{-1} P(\widetilde{\boldsymbol{Y}}|X = \boldsymbol{x})$$
$$= [0.1587; 0.2697; \boxed{0.2796;}\, 0.2593; 0.0325].$$

[15] Xia X, et al. Are Anchor Points Really Indispensable in Label-Noise Learning? NeurIPS. 2019

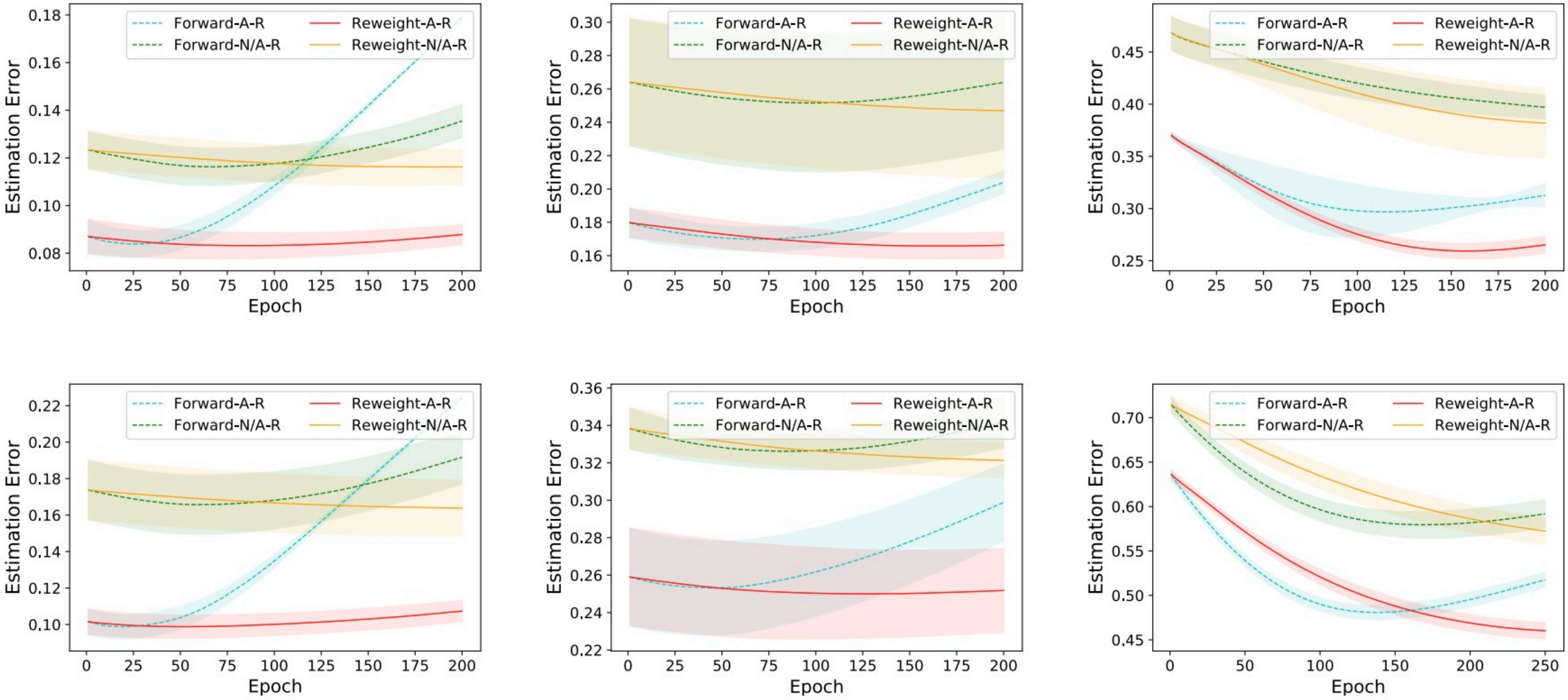
T revision [15]



Weighted loss $= \frac{1}{\mathrm{n}} \sum_{i=1}^{n} \frac{\boldsymbol{g}_{\tilde{y}_i}(\boldsymbol{x}_i)}{(\boldsymbol{T}^{\top}\boldsymbol{g})_{\tilde{y}_i}(\boldsymbol{x}_i)} L(f(\boldsymbol{x}_i), \tilde{y}_i)$,

where $f(\boldsymbol{x}) = \operatorname{argmax}_{i \in \{1,\ldots,C\}} \boldsymbol{g}_i(\boldsymbol{x})$.

[15] Xia X, et al. Are Anchor Points Really Indispensable in Label-Noise Learning? NeurIPS. 2019



(a) *MNIST* (b) *CIFAR-10* (c) *CIFAR-100*

[15] Xia X, et al. Are Anchor Points Really Indispensable in Label-Noise Learning? NeurIPS. 2019

A summary of estimating transition matrix

- How to estimate the transition matrix given only noisy data?
  Method: T estimator (by exploiting anchor points)

- How about if there is no anchor points?
  Method: VolMinNet (using the sufficiently scattered assumption)

- How to deal with poorly estimated transition matrix
  Method: T revision (revising the matrix by using a slack variable)

Conclusion and future directions

- Conclusion
  - Statistically consistent algorithms: the classifier learned by using noisy data will converge to the optimal one defined by using clean data
  - Statistically consistent algorithms are robust to the data distribution and label noise type
  - Modelling the label noise and estimating the transition matrix are cores in label-noise learning
- Future directions
  - Design effectively loss correction methods for deep learning
  - How to address the finite/small sample problem
  - How to use a small set of clean data to better estimate the transition matrix
  - How to model and estimate the instance-dependent label noise (IDN)