Robust Inference via Generative Classifiers for Handling Noisy Labels

Kimin Lee[1] Sukmin Yun[1] Kibok Lee[2] Honglak Lee[4,2] Bo Li[3] Jinwoo Shin[1,5]

[1] Korea Advanced Institute of Science and Technology (KAIST)

[2] University of Michigan

[3]University of Illinois at Urbana Champaign

[4]Google Brain

[5]AItrics

ICML 2019

Introduction: Noisy Labels

- Large-scale datasets collect class labels from
  - Data mining on social media and web data
- Large-scale datasets may contain noisy (incorrect) labels
- DNNs do not generalize well from such noisy datasets
- **Several training strategies have also been investigated**

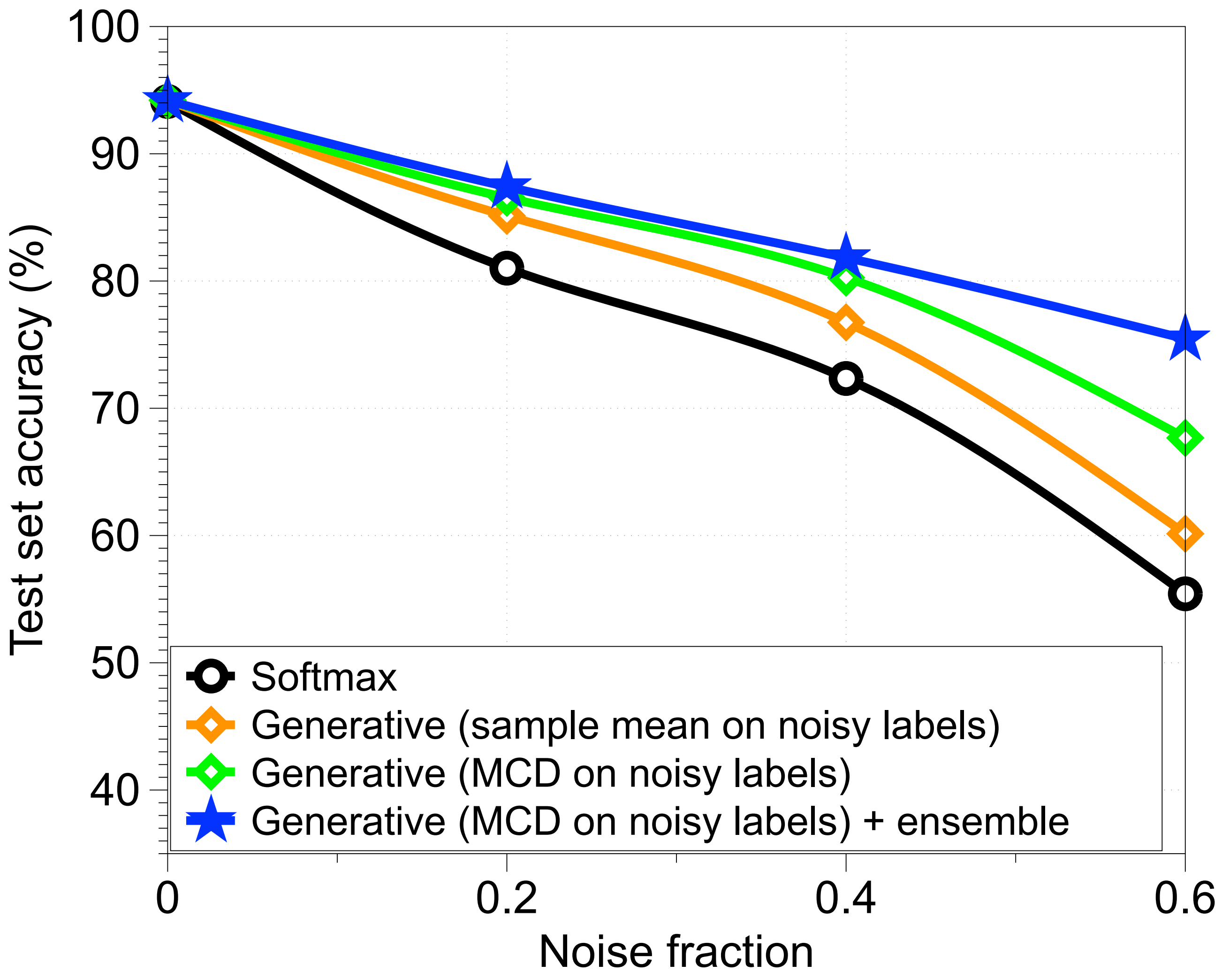
- Utilizing an estimated/corrected label
  - Bootstrapping [Reed' 14; Ma' 18]
  - Loss correction [Patrini' 17; Hendrycks' 18]

- Training on selected (cleaner) samples
  - Ensemble [Malach' 17; Han' 18]
  - Meta-learning [Jiang' 18]

[Reed' 14] Training deep neural networks on noisy labels with bootstrapping. arXiv preprint 2014.
[Hendrycks' 18] Using trusted data to train deep networks on labels corrupted by severe noise. In NeurIPS, 2018
[Ma' 18] Dimensionality-driven learning with noisy labels. In ICML, 2018
[Partrini' 17] Making deep neural networks robust to label noise: A loss correction approach. In CVPR, 2017
[Han' 18] Co-teaching: robust training deep neural networks with extremely noisy labels. In NeurIPS, 2018.
[Jiang' 18] Mentornet: Regularizing very deep neural networks on corrupted labels. In ICML, 2018.
[Malach ' 17] Decoupling" when to update" from" how to update". In NeurIPS, 2017.

Our Contributions

- We propose a new **inference method** which can be applied to any pre-trained DNNs



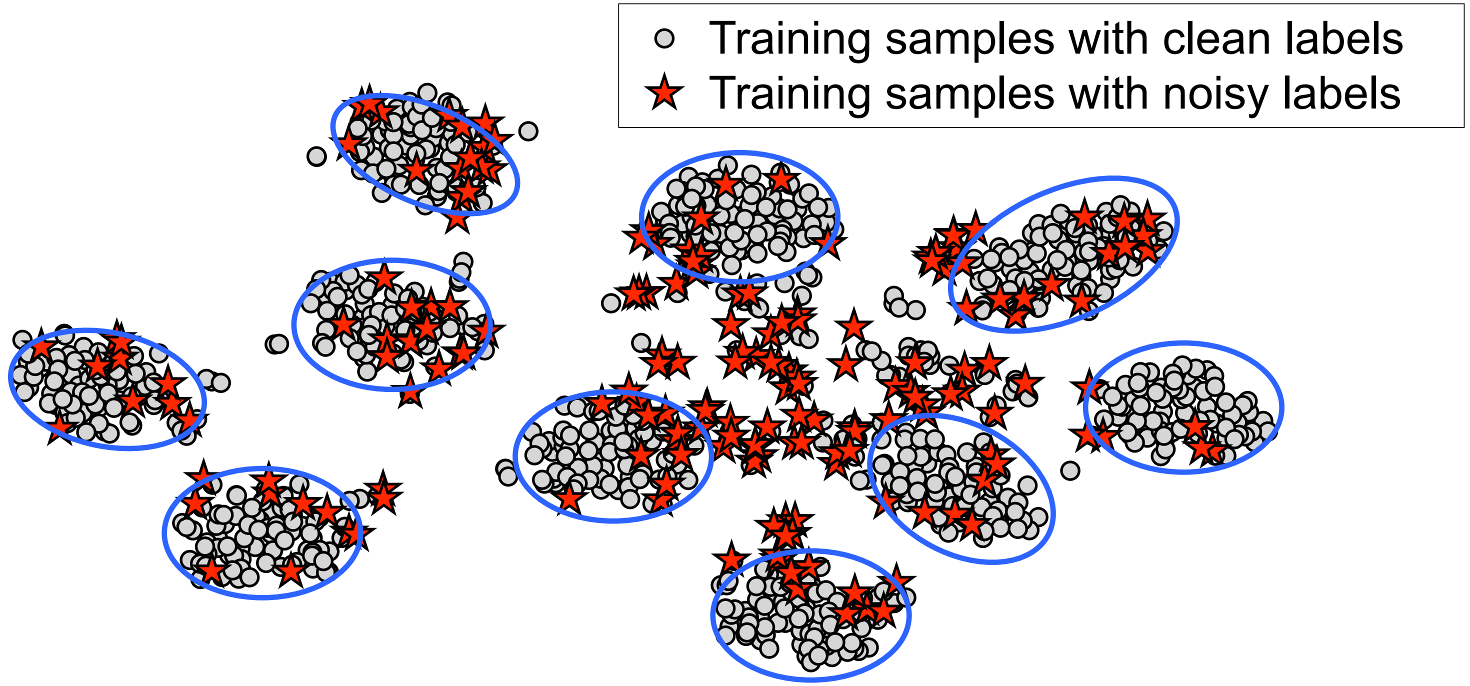
- Inducing a **"generative classifier"**
- Applying a **"robust inference"** to estimate parameters of generative classifier
  - **Breakdown points**
  - **Generalization bounds**
- Introducing **"ensemble of generative classifiers"**

Outline

- Our method: Robust Inference via Generative Classifiers
  - Generative classifier
  - Minimum covariance determinant estimator
  - Ensemble of generative classifiers
- Experiments
  - Experimental results on synthetic noisy labels
  - Experimental results on semantic and open-set noisy labels
- Conclusion

Motivation: Why Generative Classifier?

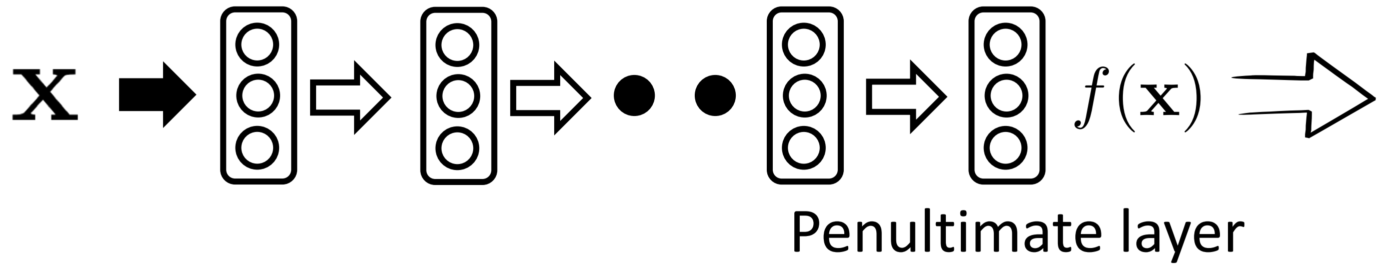
- **t-SNE embedding** of DenseNet-100 trained on CIFAR-10 with uniform noisy labels



  - Features from training samples with noisy labels (red stars) are distributed like outliers
  - Features from training samples with clean labels (black dots) are still clustered!!
- If we remove the outliers and induce decision boundaries, they can be more robust
- **Generative classifier:** model of a data distribution $P(x|y)$ instead of $P(y|x)$

Robust Inference via Generative Classifier

- Given pre-trained softmax classifier with DNNs
  - Inducing a generative classifier on the hidden feature space



$$P\left(f(\mathbf{x})|y=c\right) = \mathcal{N}\left(f(\mathbf{x})|\mu_c, \mathbf{\Sigma}\right)$$

Bayes' rule

$$P\left(y=c|f(\mathbf{x})\right) = \frac{P\left(y=c\right)P\left(f(\mathbf{x})|y=c\right)}{\sum_{c'} P\left(y=c'\right)P\left(f(\mathbf{x})|y=c'\right)} = \frac{\exp\left(\mu_c^\top \mathbf{\Sigma}^{-1} f(\mathbf{x}) - \frac{1}{2}\mu_c^\top \mathbf{\Sigma}^{-1}\mu_c + \log\beta_c\right)}{\sum_{c'} \exp\left(\mu_{c'}^\top \mathbf{\Sigma}^{-1} f(\mathbf{x}) - \frac{1}{2}\mu_{c'}^\top \mathbf{\Sigma}^{-1}\mu_{c'} + \log\beta_{c'}\right)}.$$

- How to estimate the parameters of the generative classifier?

$$\bar{\mu}_c = \sum_{i:y_i=c} \frac{f(\mathbf{x}_i)}{N_c}, \quad \bar{\mathbf{\Sigma}} = \sum_{c}\sum_{i:y_i=c} \frac{\left(f(\mathbf{x}_i)-\bar{\mu}_c\right)\left(f(\mathbf{x}_i)-\bar{\mu}_c\right)^\top}{N}, \quad \bar{\beta}_c = \frac{N_c}{N}$$

  - With training data $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$

Minimum Covariance Determinant (MCD)

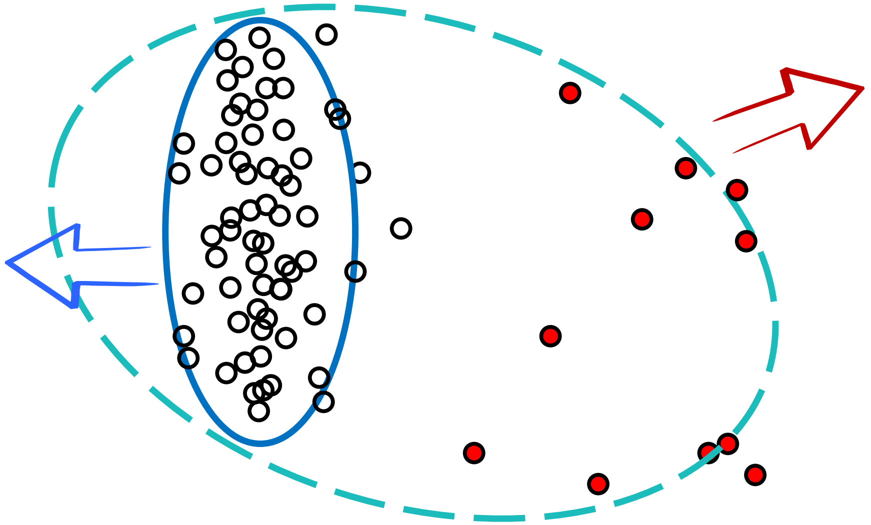
- Naïve sample estimator (green circle) can be affected by outliers (i.e., noisy labels)
- **Minimum Covariance Determinant (MCD)** estimator (blue circle)
  - For each class *c*, find a subset for which the determinant of the sample covariance matrix is minimum

$$\boxed{\min_{\mathcal{X}_{K_c} \subset \mathcal{X}_{N_c}} \det\left(\widehat{\boldsymbol{\Sigma}}_c\right) \quad \text{subject to } \left|\mathcal{X}_{K_c}\right| = K_c,}$$

  - Compute the mean and covariance matrix only using selected samples

$$\bar{\mu}_c = \sum_{i:y_i=c} \frac{f(\mathbf{x}_i)}{N_c},$$

$$\bar{\boldsymbol{\Sigma}} = \sum_c \sum_{i:y_i=c} \frac{\left(f(\mathbf{x}_i) - \bar{\mu}_c\right)\left(f(\mathbf{x}_i) - \bar{\mu}_c\right)^\top}{N}$$



Motivation of MCD

- Outliers (e.g., sample with noisy labels) are scattered in the sample spaces

Advantages of MCD estimators

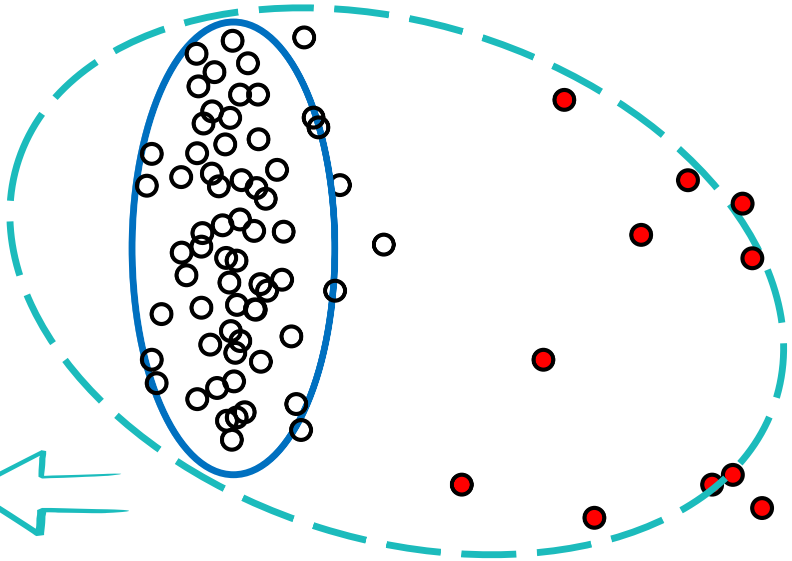
- **1. Breakdown points**
  - The smallest fraction of outliers to carry the estimate beyond all bounds.

$$|| \mu_{\texttt{true}} - \mu_{\texttt{estimate}} || = \infty$$

  - High breakdown points = robust to outliers

- **Theorem 1** (*Lopuhaa et al., 1991*)

  Under some mild assumptions, MCD estimator has near-optimal breakdown points, i.e., almost 50 %



Note: Naïve sample estimator has 0% breakdown points

[Lopuhaa et al., 1991] Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. The Annals of Statistics, 1991.

Advantages of MCD estimators

- **2. Tighter generalization errors under noisy labels**

- **Theorem 2** (Lee et al., 2019)

  Under some mild assumptions, parameters from MCD estimator are more closer to true parameters than parameters from sample estimator and has larger inter-class distance

$$\|\mu^{\texttt{true}} - \mu^{\texttt{MCD}}\| \leq \|\mu^{\texttt{true}} - \mu^{\texttt{sample}}\|$$

$$\phi(\mathbf{\Sigma}^{\texttt{MCD}})\|\mu_c^{\texttt{MCD}} - \mu_{c'}^{\texttt{MCD}}\| \geq \phi(\mathbf{\Sigma}^{\texttt{sample}})\|\mu_c^{\texttt{sample}} - \mu_{c'}^{\texttt{sample}}\|$$

Theorem 3 (Durrant et al., 2010)

Generalization error of generative classifier is **bounded by** negative of inter-class distance and distance between true and estimated parameters

[Durrant et al., 2010] A. Compressed fisher linear discriminant analysis: Classification of randomly projected data. In ACM SIGKDD, 2010.

How to Solve MCD?

Two-step approach [Hubert' 04]

- Step 1. For each class, find a subset as follows:
  - A. Uniformly sample an initial subset
    & compute sample mean and covariance matrix
  - B. Compute the Mahalanobis distance

$$\left(f(\mathbf{x}) - \widehat{\mu}_{\mathbf{c}}\right)^{\top} \widehat{\mathbf{\Sigma}}_c^{-1} \left(f(\mathbf{x}) - \widehat{\mu}_{\mathbf{c}}\right)$$

  - C. Construct a new subset which contains samples with smaller distances
  - D. Update the sample mean and covariance matrix
  - Repeat Step B ~ D until the determinant of covariance is not decreasing
- Step 2. Compute the mean and covariance only using selected samples

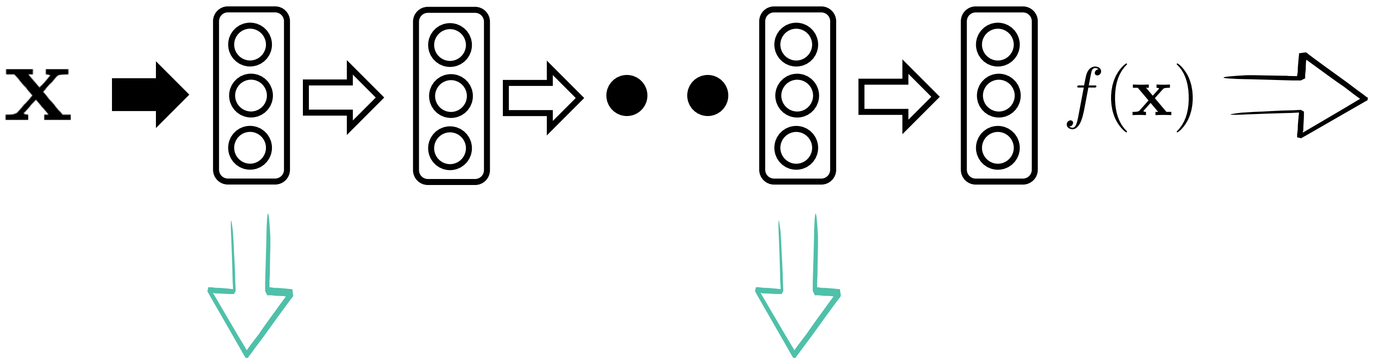
Monotonically decreasing a objective of MCD estimator [Hubert' 04] !

$$\min_{\mathcal{X}_{K_c} \subset \mathcal{X}_{N_c}} \det\left(\widehat{\mathbf{\Sigma}}_c\right)$$

[Hubert' 04] Fast and robust discriminant analysis. Computational Statistics & Data Analysis, 2004.

Ensemble of Generative Classifiers

- Boosting the performance: utilizing low-level features
  - Post-processing the generative classifiers with respect to low-level features



$$P\left(f(\mathbf{x})|y=c\right) = \mathcal{N}\left(f(\mathbf{x})|\mu_c, \mathbf{\Sigma}\right)$$

$$P\left(f(\mathbf{x})|y=c\right) = \mathcal{N}\left(f(\mathbf{x})|\mu_c, \mathbf{\Sigma}\right)$$

$$P\left(f(\mathbf{x})|y=c\right) = \mathcal{N}\left(f(\mathbf{x})|\mu_c, \mathbf{\Sigma}\right)$$

- Ensemble of generative classifiers

$$P(y=c|\mathbf{x}) = \sum_{\ell} \alpha_\ell \underline{P\left(y=c|f_\ell(\mathbf{x})\right)}$$

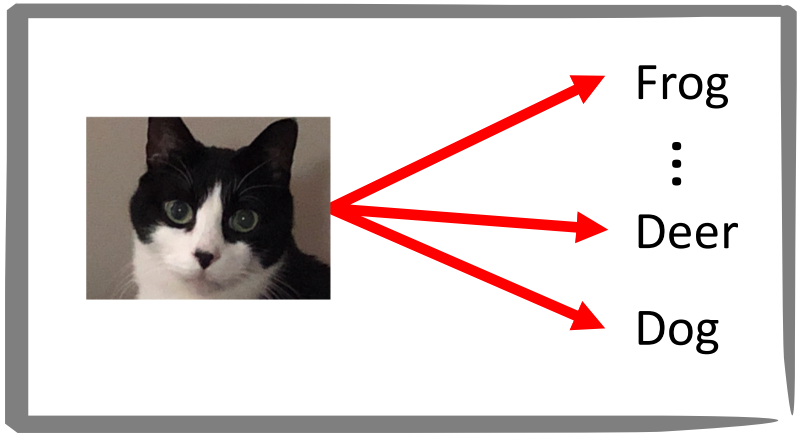
Posterior distribution from ℓ-th layer

Outline

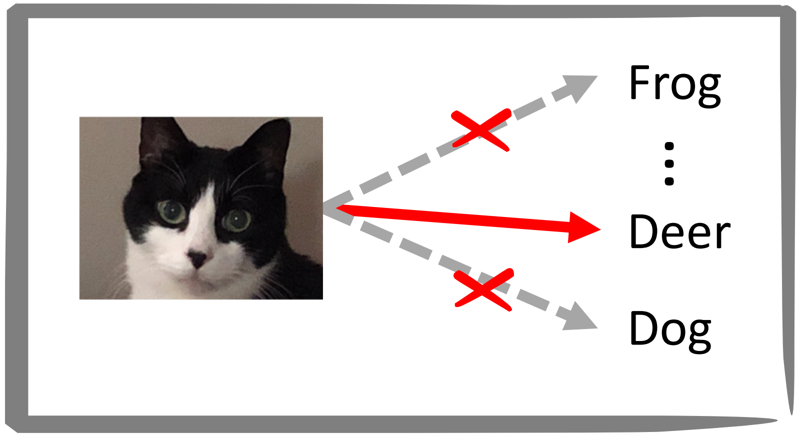
- Our method: Robust Inference via Generative Classifiers
  - Generative classifier
  - Minimum covariance determinant estimator
  - Ensemble of generative classifiers
- Experiments
  - Experimental results on synthetic noisy labels
  - Experimental results on semantic and open-set noisy labels
- Conclusion

Experiments: Setup

- Model: DenseNet-100 [Huang' 17] and ResNet-34 [He' 16]
- Image classification on CIFAR-10, CIFAR-100 [Krizhevsky' 09] and SVHN [Netzer' 11]
- NLP tasks on Tweeter [Gimpel' 11] and Reuters [Lewis' 04]
- Noise type
  - **Uniform**: corrupting a label to other class uniformly at random
  - **Flip**: corrupting a label only to a specific class



[Uniform noise]



[Flip noise]

Experiments: Empirical Analysis

- Test set accuracy of ResNet-34 trained on CIFAR-10 with 60% uniform noise

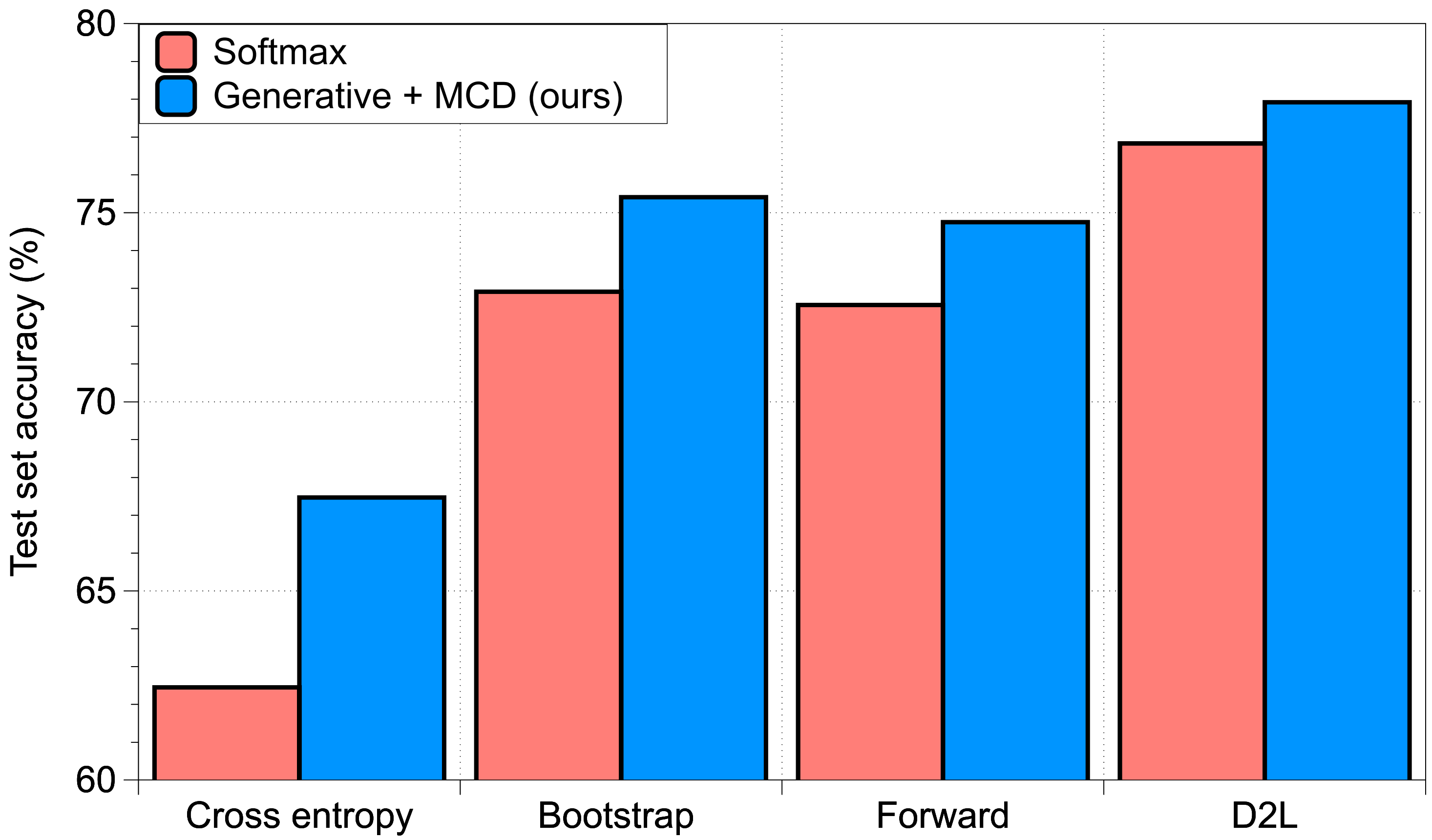
Inference	Ensemble	Clean	Uniform
Softmax	-	94.76	39.96
Generative	-	94.80	42.76
+ sample	✓	94.82	46.45
Generative	-	94.76	44.87
+ MCD (ours)	✓	94.68	**54.57**

- MCD estimator improves the performance by removing outliers

Comparison with Prior Training Methods

- Test set accuracy of ResNet-44 trained on CIFAR-10 with 60% uniform noises

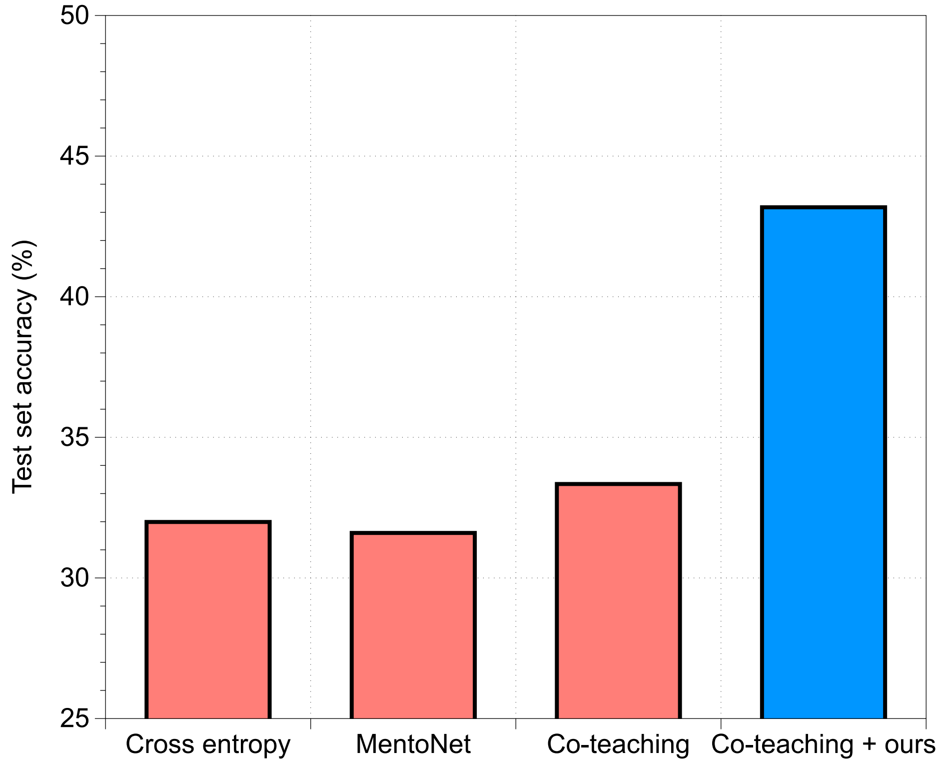
- Utilizing an estimated/corrected label
  - Bootstrap [Reed’ 14]
  - Forward[Patrini’ 17]
  - D2L [Ma’ 18]



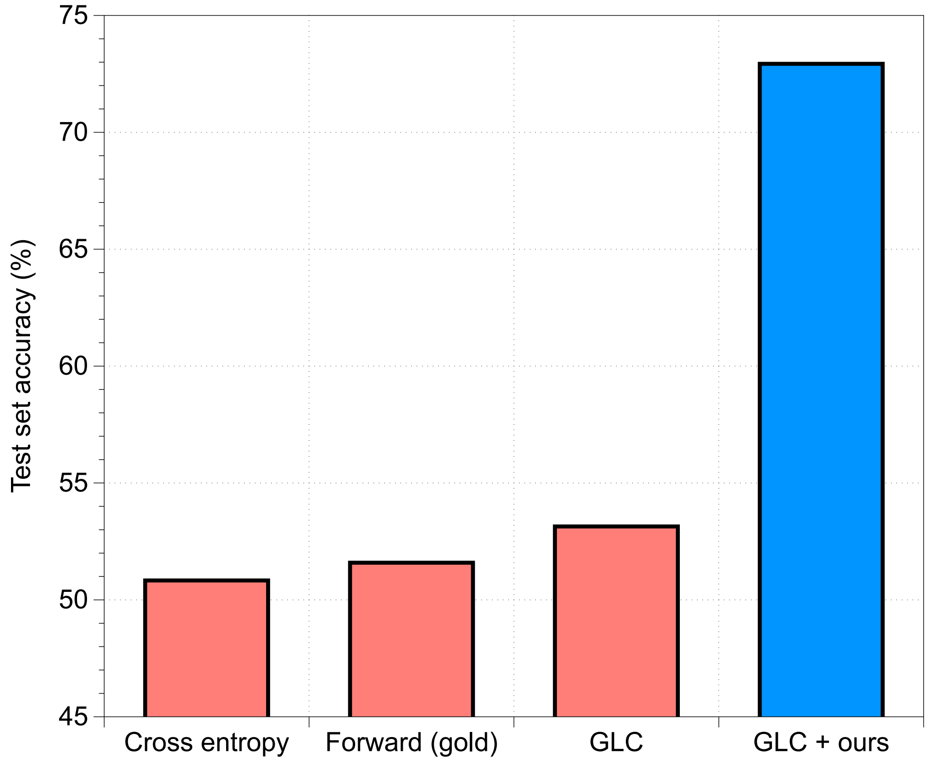
[Reed’ 14] Training deep neural networks on noisy labels with bootstrapping. arXiv preprint 2014.
[Ma’ 18] Dimensionality-driven learning with noisy labels. In ICML, 2018
[Partrini’ 17] Making deep neural networks robust to label noise: A loss correction approach. In CVPR, 2017

Comparison with Prior Training Methods

- Training methods utilizing an ensemble of classifiers or meta-learning model
  - Model: 9-layer CNNs
  - Dataset: CIFAR-100
  - Noise: 45% Flip noise



- Training methods utilizing clean labels on NLP datasets
  - Model: 2-layer FCNs
  - Dataset: Twitter
  - Noise: 60% uniform noise

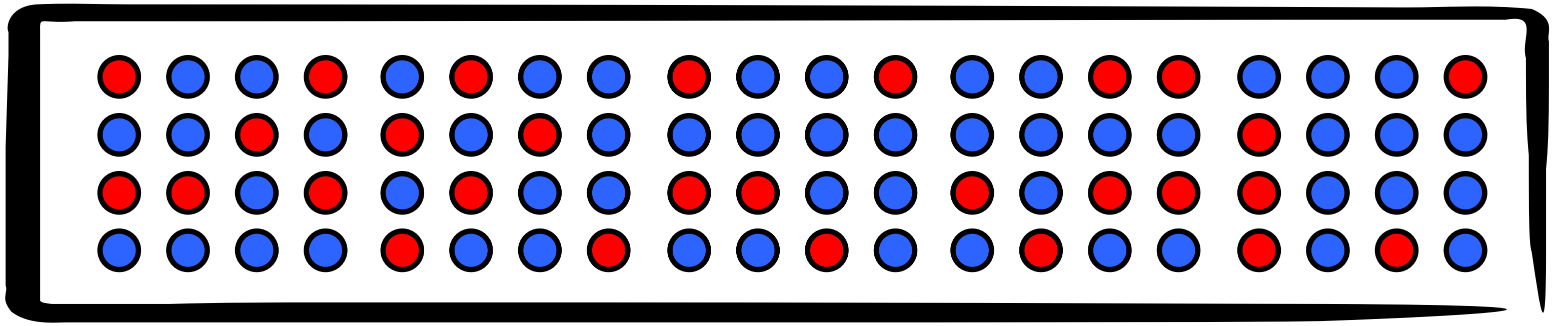


Experiments: Machine Noisy Labels

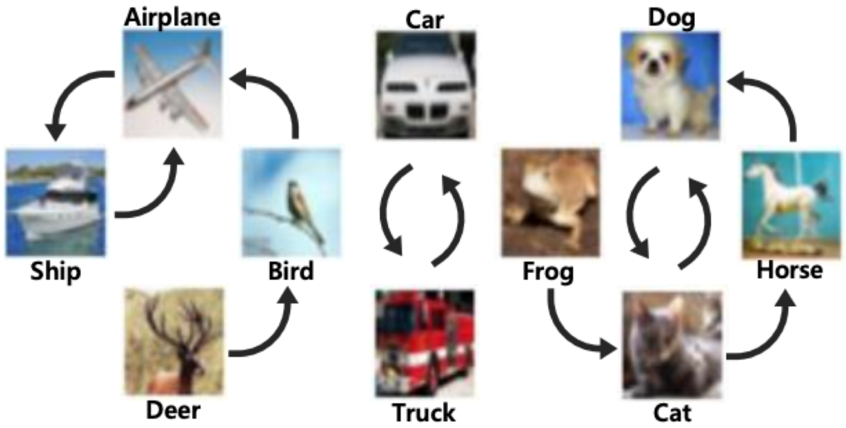
- Semantic noisy labels from a weak machine labeler

Pseudo-labeled data

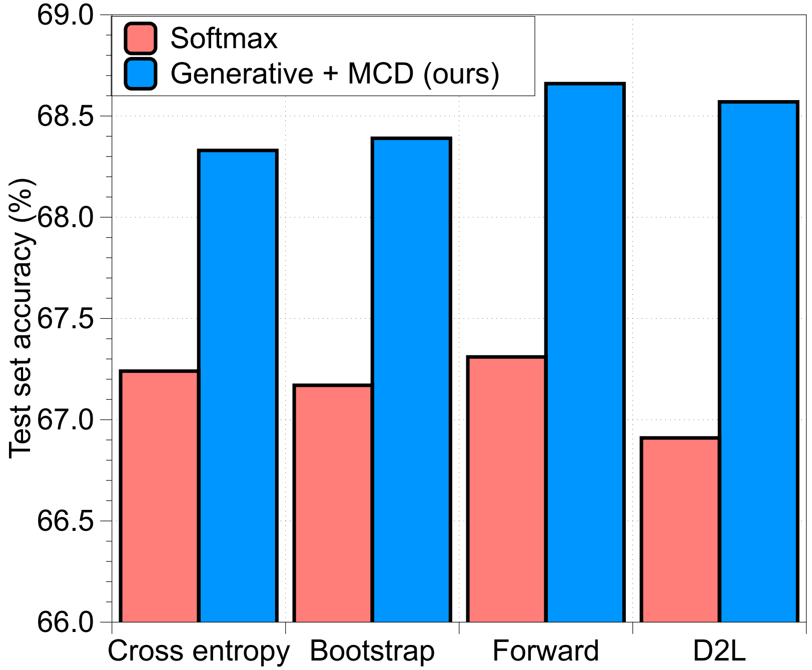
Pre-trained (weak) classifier



- Confusion graph from ResNet-34 trained on 5% of CIFAR-10 labels

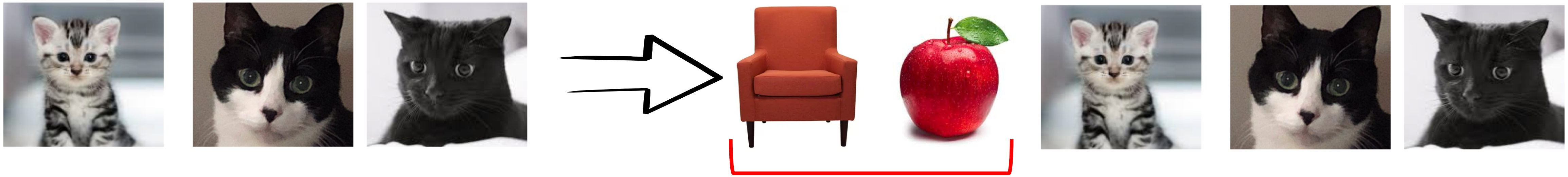


* Node: class, Edge: its most confusing class



Experiments: Open-set Noisy Labels

- What is Open-set noisy labels?
  - **Noisy samples from out-of-distribution [Wang' 18]**
  - E.g., "Cat" in CIFAR-10 (which does not contain "apple" and "chair")



Open-set noisy labels

- Experimental setup
  - In-distribution: CIFAR-10
  - 60% of noise samples from ImageNet and CIFAR-100
  - Model: DenseNet-100

Open-set data	Softmax	ours
CIFAR-100	79.01	**83.37**
ImageNet	86.88	87.05
CIFAR-100 + ImageNet	81.58	**84.35**

[Test accuracy (%) of DenseNet on the CIFAR-10]

[Wang' 18] Iterative learning with open-set noisy labels. In CVPR, 2018.

Conclusion

- To handle noisy labels,

Generative classifier
- New inference method
- LDA-based generative classifier

Robust inference
- MCD estimator
- Generalization error

Ensemble method
- Generative classifier from multiple layers

- We believe that our results can be useful for many machine learning problems:
  - Defense against adversarial attacks
  - Detecting out-of-distribution samples
- **Poster session: Pacific Ballroom #16**

Thank you for your attention ☺