Supplementary material for the paper “Unsupervised Label Noise Modeling and Loss Correction”

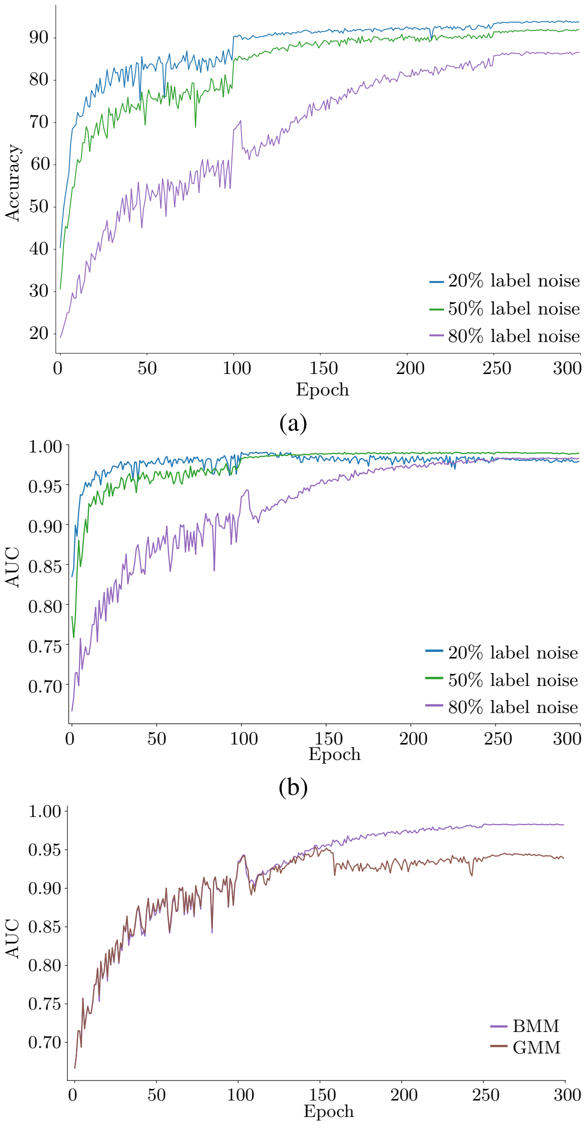
A. Beta Mixture Model (BMM)

This section extends the discussion of the proposed unsupervised BMM in the main paper providing detail on several more aspects.

BMM performance under low levels of label noise We seek robust representation learning in the presence of label noise, which may occur when images are automatically labeled. Performance will likely drop in carefully annotated datasets with near 0% noise because the loss distribution is not a two-component mixture. In this situation the BMM classifies almost all samples as clean, but some estimation errors may occur, which lead to a reliance on the sometimes incorrect network prediction instead of the true clean label. Nevertheless, for 20% noise, we outperform the compared state-of-the-art at the end of the training, demonstrating improved robustness for low noise levels.

BMM parameter estimation frequency The BMM parameters are re-estimated after every epoch once the loss correction begins (i.e. there is an initial warm-up as noted in Subsection 4.1 with no loss correction) by computing the cross-entropy loss from a forward pass with the original (potentially noisy) labels. We also tested our approach M-DYR-H (CIFAR-10, 80% of label noise) changing the estimation period to 5 and 0.5 epochs, observing no decrease in accuracy. While the original configuration presented in Figure 4(a) reaches 86.8 (86.6) for best (last), every 5 epochs leads to (86.9) 86.8 and every 0.5 to 88.0 (87.5).

BMM classification accuracy and robustness Figure 4(b) shows the clean/noisy classification capabilities of the BMM in terms of Area Under the Curve (AUC) evolution during training, demonstrating that performance and robustness are consistent across noise levels. In particular, the experiment on CIFAR-10 with M-DYR-H exceeds 0.98 AUC for 20, 50 and 80% label noise. AUC increases during training and increases faster for lower noise levels, showing increasingly better clean/noisy discrimination related to consistent BMM predictions over time.



(a)

(b)

(c)

Figure 4. M-DYR-H results on CIFAR-10 for (a) image classification and (b) clean/noisy classification of the BMM. (c) comparison of GMM and BMM for clean/noisy classification with 80% label noise.

Effect of BMM classification accuracy on image classification accuracy BMM prediction accuracy is essential for high image classification accuracy, as demonstrated by the tendency for both image classification and BMM accuracy to increase together in Figure 4(a) and (b), especially for higher noise levels. Figure 4(c) further verifies this relationship by comparing the BMM with a GMM (Gaussian Mixture Model) on CIFAR-10 with M-DYR-H and 80% label noise. The GMM gives both less accurate clean/noisy discrimination and worse image classification results (clean/noisy AUC drops from 0.98 to 0.94, while image classification accuracy drops from 86.6 to 83.5).

Performance attributable to the BMM Incorporating the BMM results in a loss that goes beyond mere regularization. This can be verified by removing the BMM and assigning fixed weights in the bootstrapping loss (0.8 to GT and 0.2 to network prediction, keeping mixup for robustness). This leads to a drop from 86.6 for M-DYR-H to 74.6 in the last epoch (80% of label noise on CIFAR10).

B. Hyperparameters

We stress that experiments across all datasets share the same hyperparameter configuration and lead to consistent improvements over the state-of-the-art, demonstrating that the general approach does not require carefully tuned hyperparams. Indeed, we are likely reporting suboptimal results that could be improved with a label noise free validation set, though availability of this set is not assumed in this paper.

Starting training with high learning rates is important: training more epochs leads to better performance, as mixup together with a high learning rate helps prevent fitting label noise. This warm-up learns the structured data (mainly associated to clean samples) and helps separate the losses between clean/noisy samples for a better BMM fit.

Experiment details All experiments used the following setup and hyperparameter configuration:

Preprocessing Images are normalized and augmented by random horizontal flipping. We use 32×32 random crops after zero padding with 4 pixels on each side.

Network A PreAct ResNet-18 is trained from scratch using PyTorch 0.4.1. Default PyTorch initialization is used on all layers.

Optimizer SGD with momentum (0.9), weight decay of 10^{-4}, and batch size 128.

Training schedule without mixup Training for 120 epochs in total. We reduce the initial learning rate (0.1) by a factor of 10 after 30, 80, and 110 epochs. Warm-up for 30 epochs, i.e. bootstrapping (when used) starts in epoch 31. This configuration is used in all experiments in Table 1.

Training schedule with mixup Training for 300 epochs in total. We reduce the initial learning rate (0.1) by a factor of 10 after 100 and 250 epochs. Warm-up for 105 epochs, i.e. bootstrapping starts in epoch 106 when used (note: the warmup period can be much longer when using mixup because it mitigates fitting label noise. Mixup $\alpha = 32$. This configuration is used for all experiments *excluding* those in Table 1.

Regarding BMM parameter estimation: parameters are fit automatically using 10 EM iterations as noted in the paper. We also ran M-DYR-H (80% of label noise, CIFAR-10) using 5 and 20 EM iterations, obtaining 87.4 (87.2) and 86.9 (86.3) for best (last) epoch, suggesting that the method is relatively robust to this hyperparameter.