Deep Learning is Robust to Massive Label Noise

David Rolnick*[1], Andreas Veit*[2], Serge Belongie[2], Nir Shavit[1]
[1]Massachusetts Institute of Technology, [2]Cornell University
drolnick@mit.edu, av443@cornell.edu, sjb344@cornell.edu, shanir@csail.mit.edu

170918
Junho Song

annotation can be expensive and, for tasks requiring expert knowledge, may simply be unattainable at scale.

ImageNet dataset [3] required more than a year of human labor on Amazon Mechanical Turk.

unsupervised learning [10], self-supervised learning [15, 22] and learning from noisy annotations [5, 14, 20]. Very large datasets (e.g., [6, 18]) The key takeaways from this paper…

Deep neural networks are able to learn from data that has been diluted by an arbitrary amount of noise.

A sufficiently large training set is more important than a lower level of noise.

Choosing good hyperparameters can allow conventional neural networks to operate in the regime of very high label noise.

Learning from noisy data

- learn directly from noisy labels and noise-robust algorithms e.g., [1, 5, 7, 12, 13, 19]

- label-cleansing methods e.g., [2]

Analyzing the robustness of neural networks

- network architectures with residual connections have a high redundancy [21]

- investigate the robustness of neural networks to adversarial examples. [17]

number of original training examples = n

adding noisy examples = α

total number of noisy labels in the training set = αn

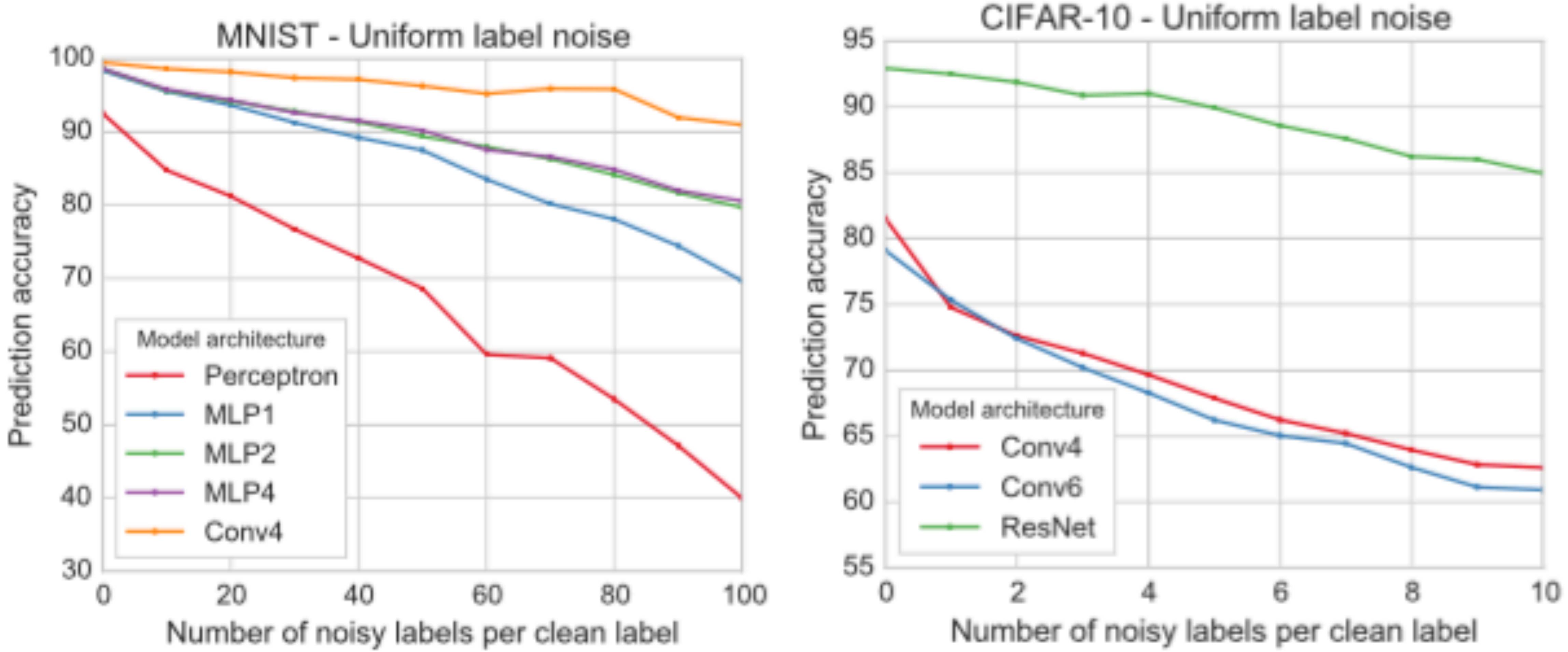
total number of training set = n + αn

Experiment 1: Training with uniform label noise

Experiment 2: Training with structured label noise

Experiment 3: Source of noisy labels

3.1 Experiment 1: Training with uniform label noise



3.2 Experiment 2: Training with structured label noise

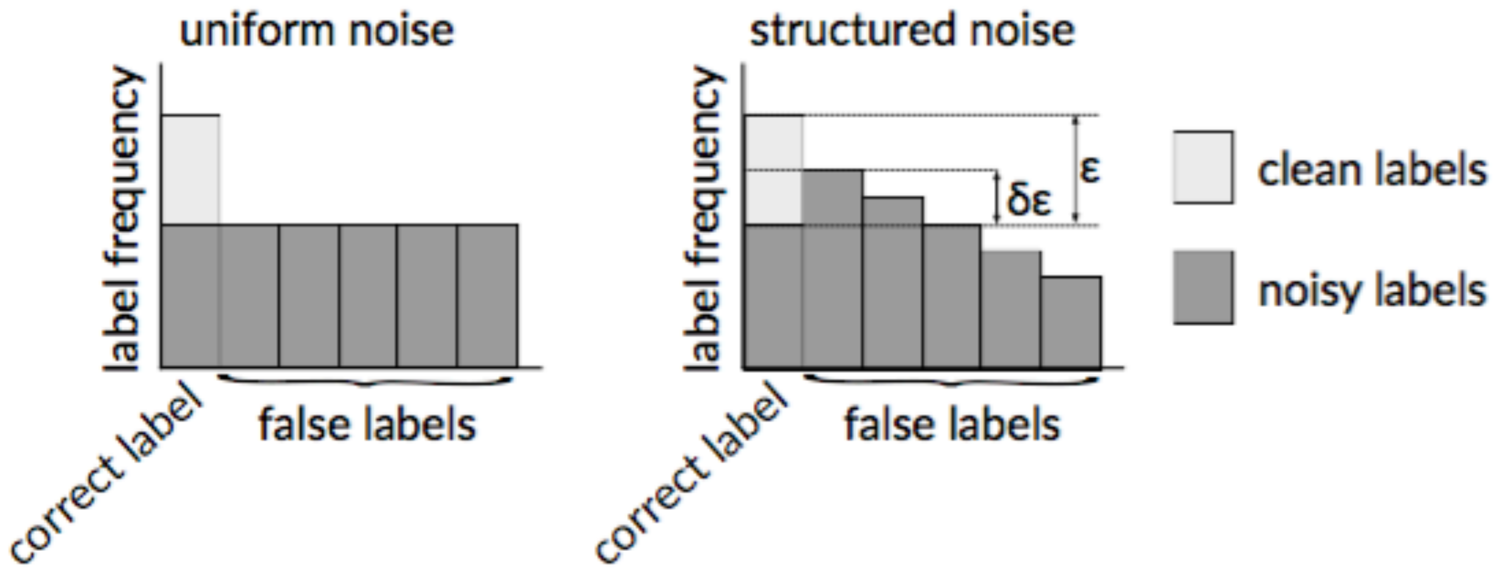


Figure 3: Illustration of uniform and structured noise models. In the case of structured noise, the order of false labels is important; we tested decreasing order of confusion, increasing order of confusion, and random order. The parameter δ parameterizes the degree of structure in the noise. It defines how much more likely the second most likely class is over chance.

3.2 Experiment 2: Training with structured label noise

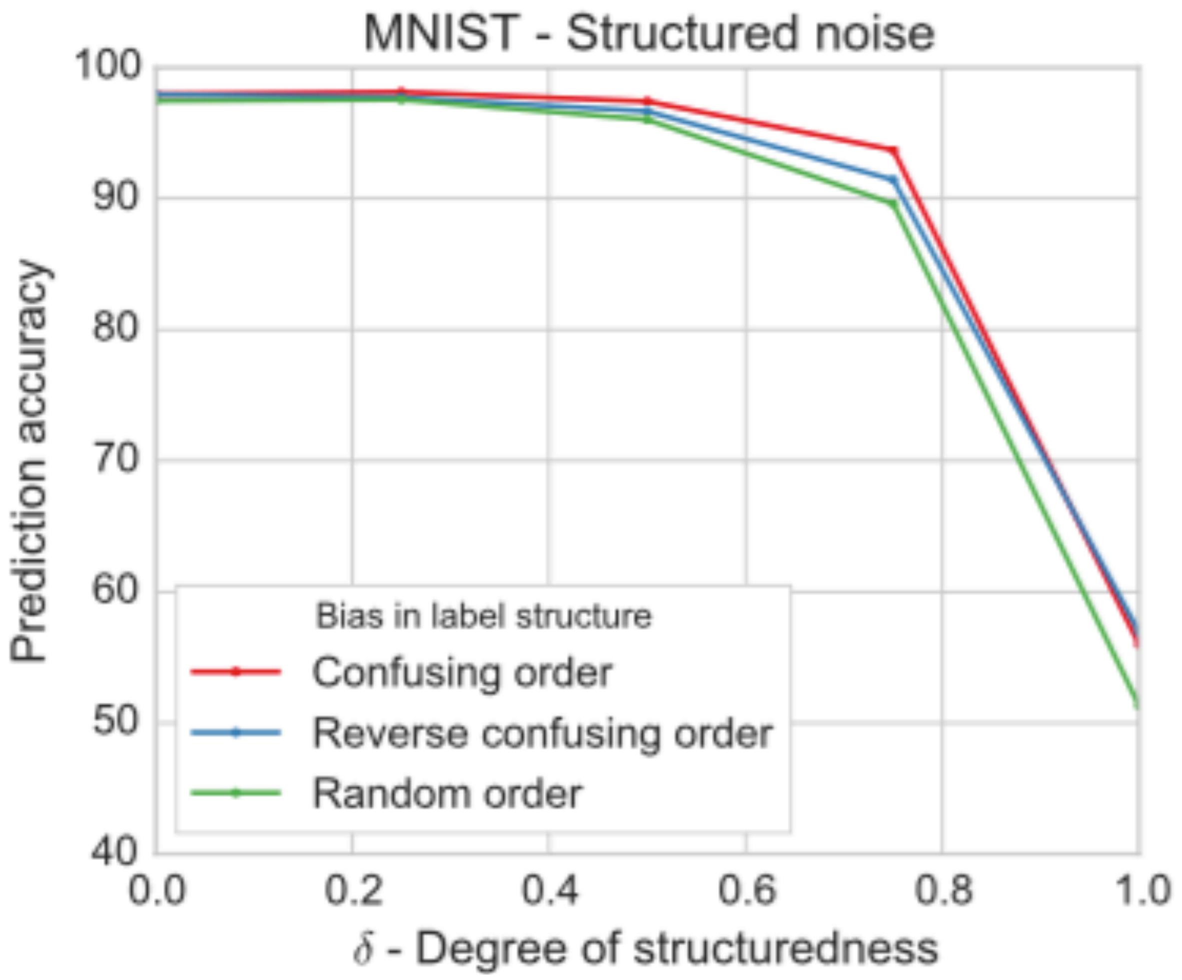
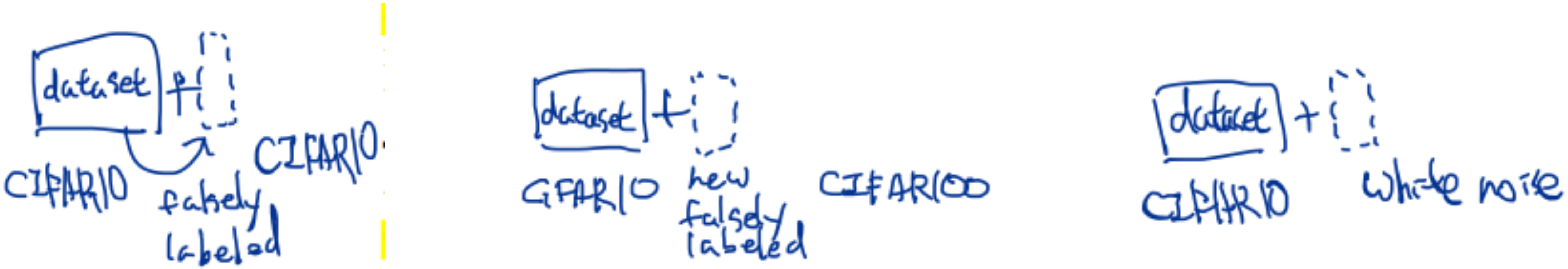


Figure 4: Performance on MNIST with fixed $\alpha = 20$ noisy labels per clean label. Noise is drawn from three types of structured distribution: (1) "confusing order" (highest probability for the most confusing label), (2) "reverse confusing order", and (3) random order. We interpolate between uniform noise, $\delta = 0$, and noise so highly skewed that the most common false label is as likely as the correct label, $\delta = 1$. Except for $\delta \approx 1$, performance is similar to uniform noise.

3.3 Experiment 3: Source of noisy labels



3.3 Experiment 3: Source of noisy labels

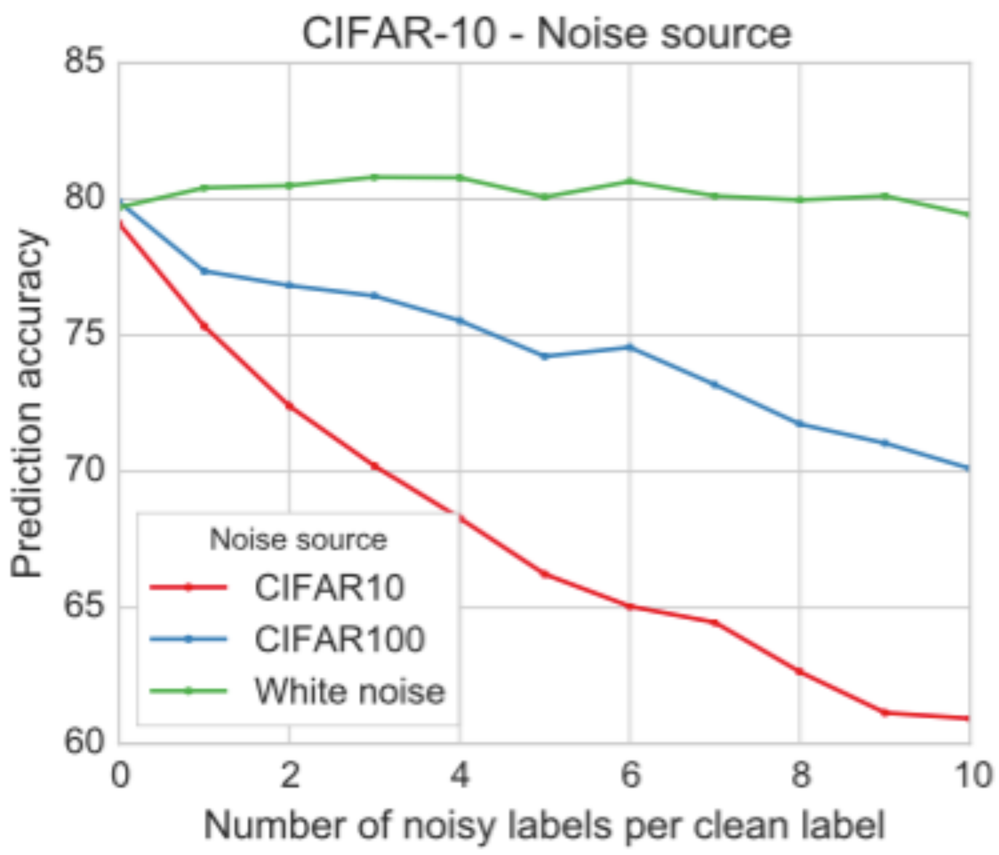


Figure 5: Performance on CIFAR-10 for varying amounts of noisy labels. Noisy training examples are drawn from (1) CIFAR-10 itself, but mislabeled uniformly at random, (2) CIFAR-100, with uniformly random labels, and (3) white noise. Noise from CIFAR-100 resulted in only half the drop in performance observed with noise from CIFAR-10 itself.

4. Importance of larger datasets

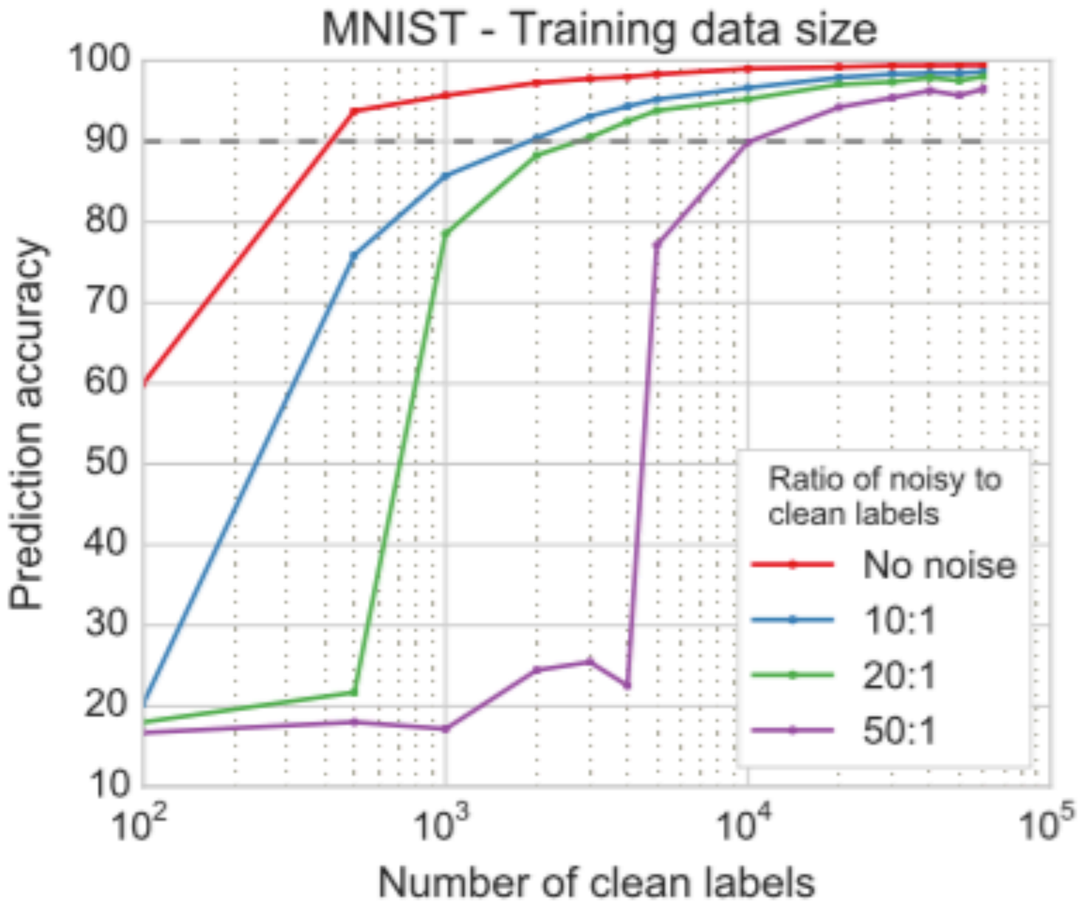
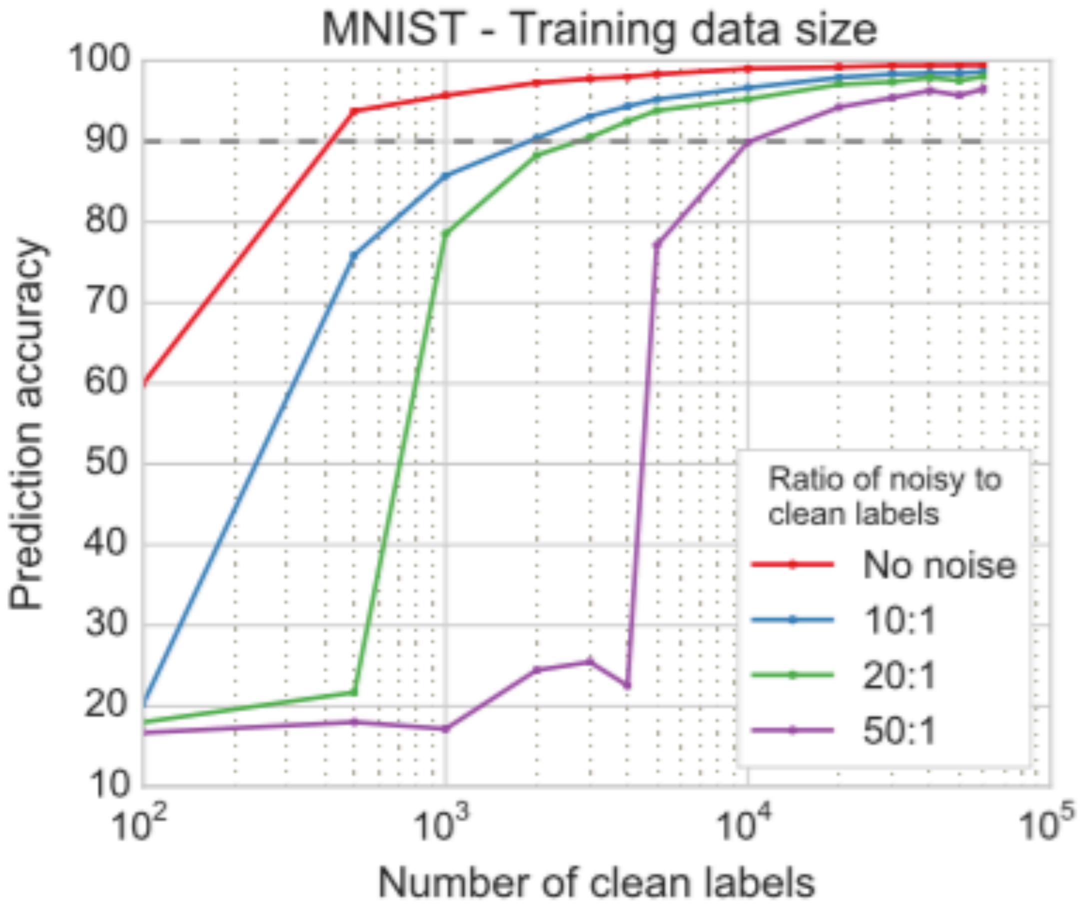


Figure 7: Performance on MNIST at various noise levels, as a function of the number of clean labels. There seems to be a critical amount of clean training data required to successfully train the networks. This threshold increases as the noise level rises. For example, at $\alpha = 10$, 2,000 clean labels are needed to attain 90% performance, while at $\alpha = 50$, 10,000 clean labels are needed.

4. Importance of larger datasets



independent of the noise level

critical amount of clean training data that is required to successfully train the networks

Figure 7: Performance on MNIST at various noise levels, as a function of the number of clean labels. There seems to be a critical amount of clean training data required to successfully train the networks. This threshold increases as the noise level rises. For example, at $\alpha = 10$, 2,000 clean labels are needed to attain 90% performance, while at $\alpha = 50$, 10,000 clean labels are needed.

5. Training on noisy datasets - Batch size and Learning rate

$$H(X) = -\langle \log \hat{y}_{f(\mathbf{x})} \rangle_X,$$

$\boxed{-y \log \hat{y}} - (1-y) \log (1-\hat{y})$

$$H_\alpha(X) := -\frac{1}{1+\alpha} \langle \log \hat{y}_{f_0(\mathbf{x})} \rangle_X - \frac{\alpha}{m(1+\alpha)} \sum_{k=1}^{m} \langle \log \hat{y}_k \rangle_X$$

$$\propto -\boxed{\langle \log \hat{y}_{f_0(\mathbf{x})} \rangle}_X - \boxed{\alpha \left\langle \log \prod_{k=1}^{m} \hat{y}_k^{1/m} \right\rangle}_X$$

5. Training on noisy datasets - Batch size and Learning rate

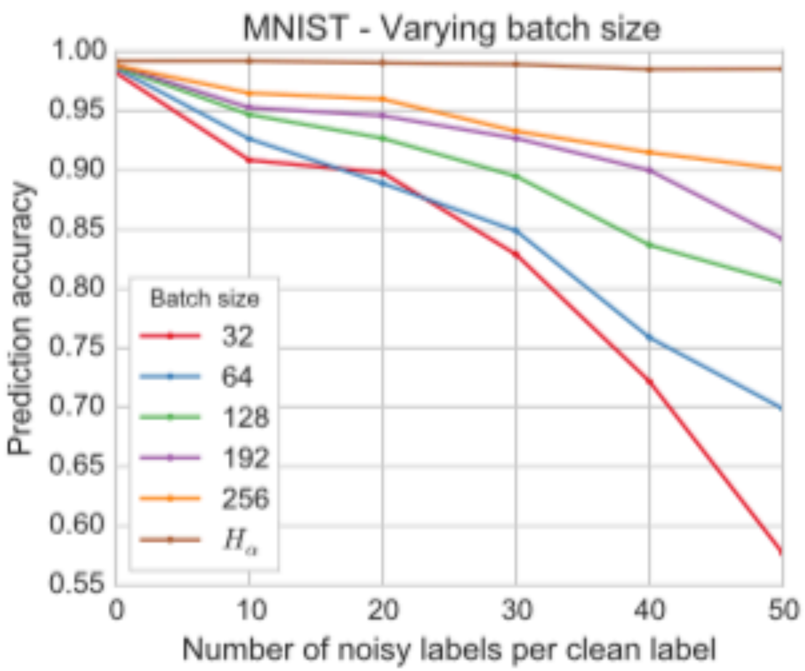


Figure 8: Performance on MNIST for varying batch size as a function of noise level. Higher batch size gives better performance. We approximate the limit of infinite batch size by training without noisy labels, but using the *noisy loss function* H_α.

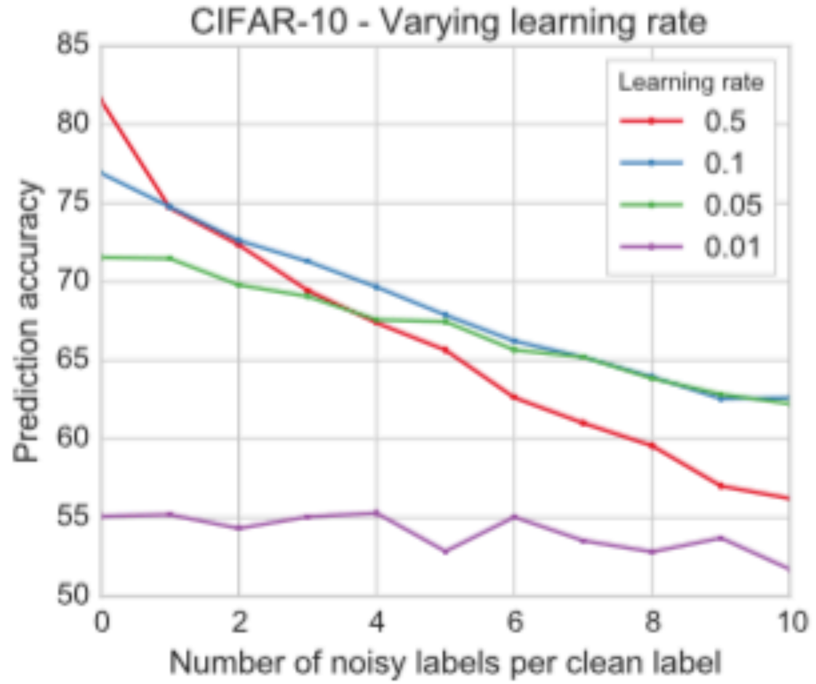


Figure 9: Performance on CIFAR-10 for varying learning rate as a function of noise level. Lower learning rates are generally optimal as the noise level increases.

better net: deeper (Conv) net

noisy ratio of skewed noisy label(second one) : 0.6 below

noisy label source : not mixed

independent of the noisy level and critical amount of clean data

larger batch size, lower learning rate