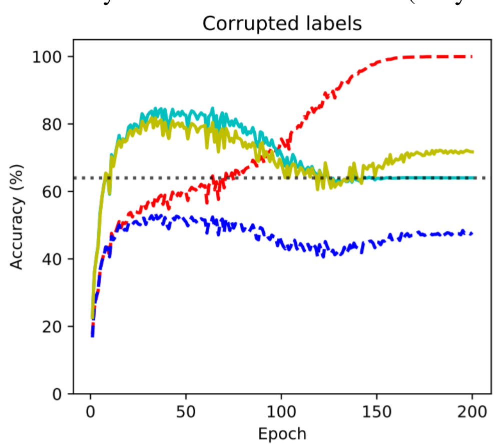
Appendix

A Supplementary Figures and Tables

Table 3: CIFAR-10 with 100% random labels. Note that test performance is always 10%.

	Models that fit train 100%	Mixup [23]	SAT [5] & Ours ($\alpha = 0.2$)
Train Acc.	100.0%	12.1%	**10.2%**
Gen. Gap	90.0%	2.1%	**0.2%**

Figure 2: The performance on the clean training set (the green curve) rises above the total number of correct examples in the training set (the dotted line), before the model fits the entire noisy training set (the red curve) and drops in accuracy on the clean validation set (the yellow curve) [5].



B The Self-Adaptive Training Algorithm of [5]

The algorithm form of self-adaptive training is reproduced below. In particular, label correction appears on lines 6 and 10, and re-weighting appears on lines 8 and 10. In the algorithm, the $\mathbf{t}_i$ represent "soft labels" on the examples in the training set, which start out as the (possibly noisy) "one-hot" labels. The model trains regularly until epoch E_s, when the model begins updating the soft labels based on current predictions.

Algorithm 1 Self-Adaptive Training [5]

1: $\{\mathbf{t}_i\}_n = \{\mathbf{y}_i\}_n$
2: **for** $e = 1$ to E_s **do**
3:   **for** $i = 1$ to m **do**
4:     $\mathbf{p}_i = f(\mathbf{x}_i)$
5:     **if** $E > E_s$ **then**
6:       $\mathbf{t}_i = \alpha \times \mathbf{t}_i + (1 - \alpha) \times \mathbf{p}_i$ {*LABEL CORRECTION*}
7:     **end if**
8:     $w_i = \max_j \mathbf{t}_{i,j}$ {*RE-WEIGHTING*}
9:   **end for**
10:   $\mathcal{L}(f) = -\frac{1}{\sum w_i} \sum_i w_i \sum_j \mathbf{t}_{i,j} \log \mathbf{p}_{i,j}$
11:   Update f by SGD on $\mathcal{L}(f)$.
12: **end for**

C Mathematical Derivation of the fact that the Self-Adaptive Weights should be In-Distribution Probabilities

Formally, the goal of the model θ is to minimize the true loss

$$\mathcal{L}_{\mathcal{D}}(\theta) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(f_\theta(\mathbf{x}), y)].$$

Now, suppose that there are many samples $\{(\mathbf{x}_i, y_i)\}_{1 \le i \le n}$, where some come from a distribution $\mathcal{D}$ and some come from a distribution $\mathcal{D}'$. Let the datapoint $(\mathbf{x}_i, y_i)$ be drawn from the distribution $\mathcal{D}$ with probability p_i and $\mathcal{D}'$ with probability $1 - p_i$.

For many kinds of noise (including uniform label noise), $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}'}[\ell(f_\theta(\mathbf{x}), y)]$ is constant for all θ. Since to compare models it suffices to compute loss up to translation, we assume that $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}'}[\ell(f_\theta(\mathbf{x}), y)]$ is always 0.

For any weights q_i $(1 \le i \le n)$, we can write an unbiased Monte Carlo estimator of the true loss $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(f_\theta(\mathbf{x}), y)]$ as (modulo some algebra)

$$\frac{\sum_{i=1}^{n} q_i \ell(f_\theta(\mathbf{x}_i), y_i)}{p_1 q_1 + p_2 q_2 + \ldots + p_n q_n}. \tag{2}$$

We seek to choose the q_i to minimize the variance of the estimator (2). The scaling of the q_i is irrelevant, so treat $\|q\|_2$ as constant. Under the assumption that the variance of $\ell(f_\theta(\mathbf{x}_i), y_i)$ is v for all i, and since the pairs $(\mathbf{x}_i, y_i)$ are independent, the variance of the numerator of (2) is just $v \cdot \|q\|_2^2$—a constant. Thus, minimal variance is achieved in (2) when the denominator is maximized. By the Cauchy-Schwarz inequality, for fixed $\|q\|_2$, the denominator is maximized with $q_i \propto p_i$.