Supplementary Material for "Does Label Smoothing Mitigate Label Noise?"

A. Proof of Theorem 1

Note that for linear models, $\Omega(\mathbf{f})$ is a convex function of $\mathbf{W}_{y'}$. Hence we can find the minimiser of $\Omega(\mathbf{f})$ by solving for when the gradient vanishes. We have,

$$\frac{\partial \Omega(\mathbf{f})}{\partial \mathbf{W}_i} = \mathop{\mathbb{E}}_{\mathbb{Q}} \left[L \cdot \frac{e^{\langle \mathbf{W}_i, x \rangle}}{\sum_{y'} e^{\langle \mathbf{W}_{y'}, x \rangle}} \cdot x - x \right].$$

We can swap differential and expectation in the above equation as $\Omega(\mathbf{f})$ is differentiable in both $\mathbf{W}_i$ and x. Now we show that the gradient evaluates to zero at $\mathbf{W}_i = 0, \forall i$:

$$\begin{aligned} \left.\frac{\partial \Omega(\mathbf{f})}{\partial \mathbf{W}_i}\right|_{\mathbf{W}_i = 0} &= \mathop{\mathbb{E}}_{\mathbb{Q}} \left[L \cdot \frac{1}{\sum_{y'} 1} x - x \right] \\ &= \mathop{\mathbb{E}}_{\mathbb{Q}} \left[L \cdot \frac{1}{L} x - x \right] \\ &= 0. \end{aligned}$$

To verify uniqueness, observe that for fixed x, the regulariser is equivalently $\mathrm{KL}(\mathrm{uniform} \| \mathbf{p}(x))$ for $\mathbf{p}(x) = \mathrm{softmax}(\mathbf{f}(x))$. This is minimised iff $\mathbf{p}(x)$ is uniform, which holds iff all elements of $\mathbf{f}(x) = \mathbf{W}x$ are the same. Since the regulariser takes expectation over x, if the feature matrix $\mathbf{X}$ has rank $> L$ (the # of labels), the only possibility is for $\mathbf{W} = 0$.

B. Experimental Setup

B.1. Architecture

We use ResNet with batch norm (He et al., 2016) for our experiments with the following configurations. For CIFAR-10 and CIFAR-100 we experiment with ResNet-32 and ResNet-56. We use ResNet-v2-50 for our experiments with ImageNet. We list the architecture configurations in terms of (n_{layer}, n_{filter}, stride) corresponding to each ResNet block in Table 7.

Architecture	**Configuration: [(n_{layer}, n_{filter}, stride)]**
ResNet-32	[(5, 16, 1), (5, 32, 2), (5, 64, 2)]
ResNet-56	[(9, 16, 1), (9, 32, 2), (9, 64, 2)]
ResNet-v2-50	[(3, 64, 1), (4, 128, 2), (6, 256, 2), (3, 512, 2)]

Table 7. ResNet Architecture configurations used in our experiments (He et al., 2016).

B.2. Training

We follow the experimental setup from Müller et al. (2019). For both CIFAR-10 and CIFAR-100 we use a mini-batch size of 128 and train for 64k steps. We use stochastic gradient descent with Nesterov momentum of 0.9. We use an initial learning rate of 0.1 with a schedule of dropping by a factor of 10 at 32k and 48k steps. We set weight decay to 1e-4. On ImageNet we train ResNet-v2-50 using the LARS optimizer (You et al., 2017) for large batch training, with a batch size of 3500, and training for 32768 steps. For data augmentation we used random crops and left-right flips [1].

For our distillation experiments we train only with the cross-entropy objective against the teacher's logits. We use a temperature of 2 unless specified otherwise when describing an experiment.

We ran training on CIFAR-100 and CIFAR-10 using 4 chips of TPU v2 and ImageNet using 128 chips of TPU v3. Training for CIFAR-100 and CIFAR-10 took under 15 minutes, and for ImageNet around 1.5h.

[1] `https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/data_generators/image_utils.py`

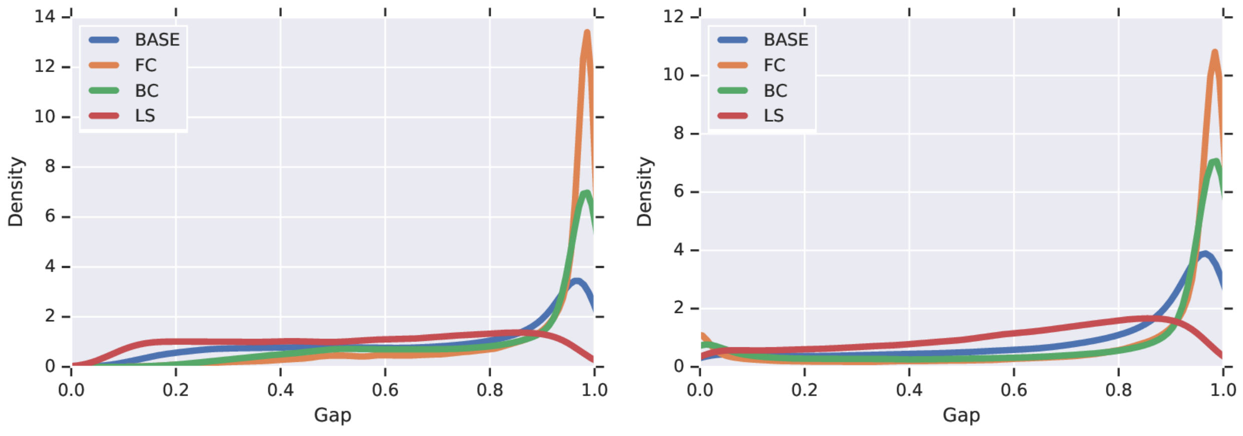
C. Experiments: Additional Results

C.1. Comparison of Smoothing against Label Noise Baselines

Dataset	Architecture	Baseline	LS	FC smoothing	BC smoothing	FC Patrini	BC Patrini
CIFAR100	RESNET-32	57.06±0.38	60.70±0.28	61.29±0.38	53.91±0.40	57.25±0.24	55.89±0.33
CIFAR100	RESNET-56	54.93±0.37	59.04±0.53	60.00±0.31	52.25±0.51	55.09±0.39	55.00±0.13
CIFAR10	RESNET-32	80.44±0.63	83.95±0.18	80.78±0.42	77.23±0.72	80.33±0.29	80.65±0.59
CIFAR10	RESNET-56	77.98±0.24	80.98±0.48	79.66±0.26	77.32±0.35	77.97±0.45	77.66±0.44

Table 8. Label smearing results under added label noise with probability of flipping each label $\rho = 20\%$. Label smoothing parameter $\alpha = 0.1$. For Patrini estimation of matrices for each label we use logits of an example falling into the 99.9th percentile according to activations for that label.

Figure 7 shows density plots of differences between maximum logit value (or corresponding to true/noisy label) and the average logit value across different portions of the training data. We notice that while label smoothing is reducing the confidence (by lowering the peak around 1.0), backward correction and forward correction methods increase the confidence by boosting the spike. This is the case for both the noisy and true labels, however the effect is much stronger on the correct label logit activation.



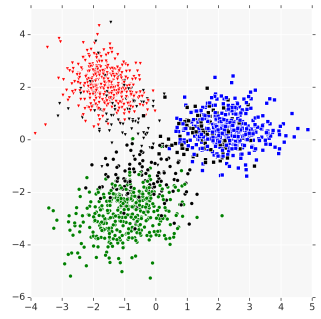
(a) Max logit, all data.

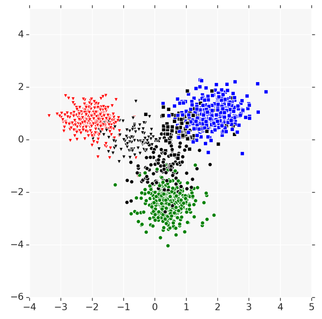
(b) True label, non-noisy part.

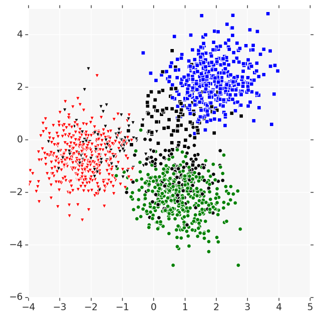
Figure 7. Density plots showing distribution differences between maximum logit (or corresponding to true label) and the average over all logits on different portions of train data and from different label smearing methods. Results using $\alpha = 0.2$, dataset CIFAR-100, and the ResNet-32 model.

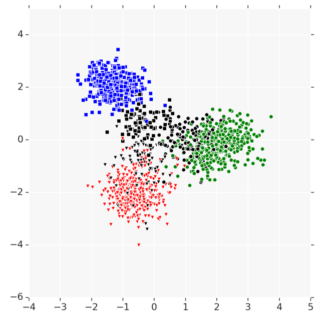
C.2. Logit Visualisation Plots

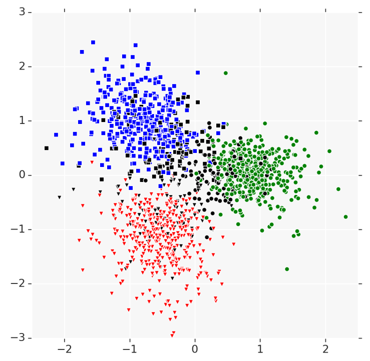
In this section we present additional pre-logit visualization plots - for CIFAR-100 trained with ResNet-56 in Figure 8 (a-d), for CIFAR-10 trained with ResNet-32 in Figure 8(e-g). Figure 9 visualises the pre-logits for backward and forward correction on CIFAR-100 trained with ResNet-32. As before, we see that both methods are able to denoise the noisy instances.

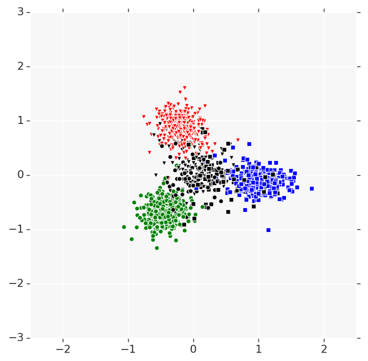

(a) LS $\alpha = 0$. Classes (0,1,2) CIFAR-100.


(b) LS $\alpha = 0.2$. Classes (0,1,2) CIFAR-100.


(c) LS $\alpha = 0$. Classes (3,4,5) CIFAR-100.


(d) LS $\alpha = 0.2$. Classes (3,4,5) CIFAR-100.


(e) LS $\alpha = 0$. Classes (0,1,2) CIFAR-10.


(f) LS $\alpha = 0.2$. Classes (0,1,2) CIFAR-10.

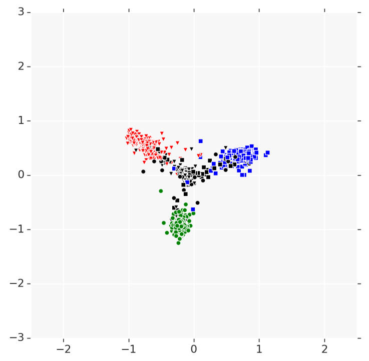
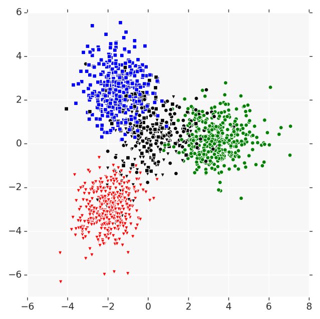
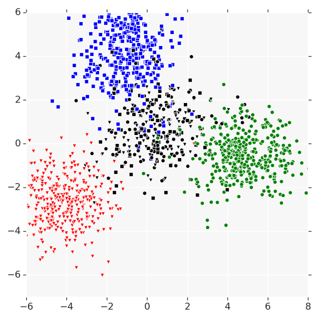
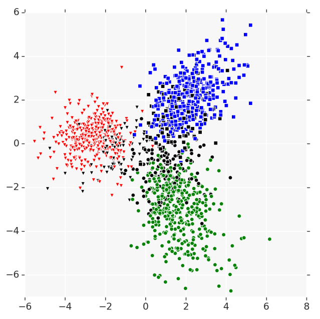

(g) LS $\alpha = 0.7$. Classes (0,1,2) CIFAR-10.

Figure 8. Effect of label smoothing on pre-logits (penultimate layer output) under label noise. Here, we visualise the pre-logits of a ResNet-56 for three classes on CIFAR-100 (in the top figures), a ResNet-32 for three classes on CIFAR-10 (in the bottom figures), using the procedure of Müller et al. (2019). The black markers denote instances which have been labeled incorrectly due to noise. Smoothing is seen to have a denoising effect: the noisy instances' pre-logits become more uniform, and so the model learns to not be overly confident in their label.


(a) Forward correction $\alpha = 0.1$.


(b) Forward correction $\alpha = 0.7$.


(c) Backward correction $\alpha = 0.1$.

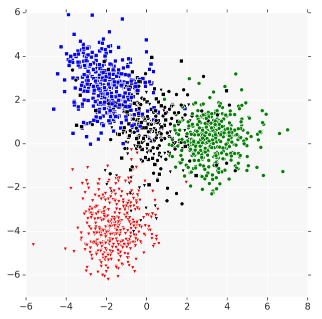

(d) Backward correction $\alpha = 0.7$.

Figure 9. Visualisation of logits for backward and forward correction of a ResNet-32 for three classes on CIFAR-100, using the procedure of Müller et al. (2019). The red, blue and green colours denote instances from three different classes, and the black coloured points have label noise.