A More Empirical Study Details for Section 3

Algorithm 2 Baselines (the three curricula) in Section 3

input: $\{(x_i, y_i)\}_{i=1}^n$, $\ell(\cdot,\cdot)$, $f(\cdot;\theta)$, $\{\eta_t\}_{t=0}^{T_\kappa}$, $\{T_i\}_{i=0}^{\kappa}$, k
initialize: θ, $T_{-1}=0$, $k=n$, $\rho_i=0, g_i=f(x_i)\ \ \forall i\in[n]$
for $j \in \{0,\cdots,\kappa\}$ **do**
  for $t \in \{T_{j-1},\cdots,T_j\}$ **do**
    if $j>0$ **then**
      Baseline1: Alternating between the highest and lowest scored samples:
      if $j\%2=1$ **then**
        $S_t \leftarrow$ top-k samples with the **largest** score $\hat{a}_t(i)$;
      else
        $S_t \leftarrow$ top-k samples with the **smallest** score $\hat{a}_t(i)$;
      end if
      Baseline2: always selecting the highest-scored samples:
      $S_t \leftarrow$ top-k samples with the **largest** score $\hat{a}_t(i)$;
      Baseline3: always selecting the lowest-scored samples:
      $S_t \leftarrow$ top-k samples with the **smallest** score $\hat{a}_t(i)$;
      Update θ by mini-batch SGD with learning rate η_t to minimize the task's loss $\ell(\cdot)$ on S_t;
    end if
    Estimate linear dynamics $\left.\frac{\partial f(x_i;\theta_t)}{\partial t}\right|_D$ *and compute scores* $\hat{a}_{t+1}(i)$:
    Uniform sampling $D\subseteq[n]$ up to size n;
    Update θ by large-batch SGD with learning rate η_t to minimize L2 loss on D;
    Compute $f(x_i)$ for all samples $i\in[n]$;
    for $i\in\{1,\cdots,n\}$ **do**
      $\rho_i \leftarrow \rho_i+\eta_t$, $\frac{\partial f(x_i)}{\partial t}=\frac{f(x_i)-g_i}{\rho_i}$;
      Restore $\rho_i \leftarrow 0$ and $g_i \leftarrow f(x_i)$;
      Compute $a_t(i)$ by Eq. (7) (regression) or Eq. (12) (classification);
      Update the exponential moving average $\hat{a}_{t+1}(i)$ for all samples $i\in[n]$ using Eq. (8);
    end for
  end for
end for

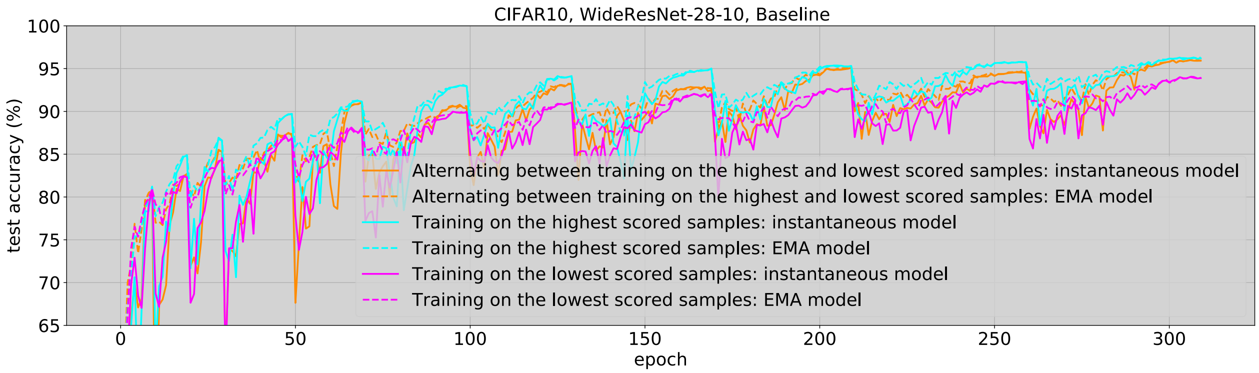

Figure 7: Test set accuracy of WideResNet-28-10 (instantaneous model) and its exponential moving average (EMA model) during the course of training when using the three data selection curricula.

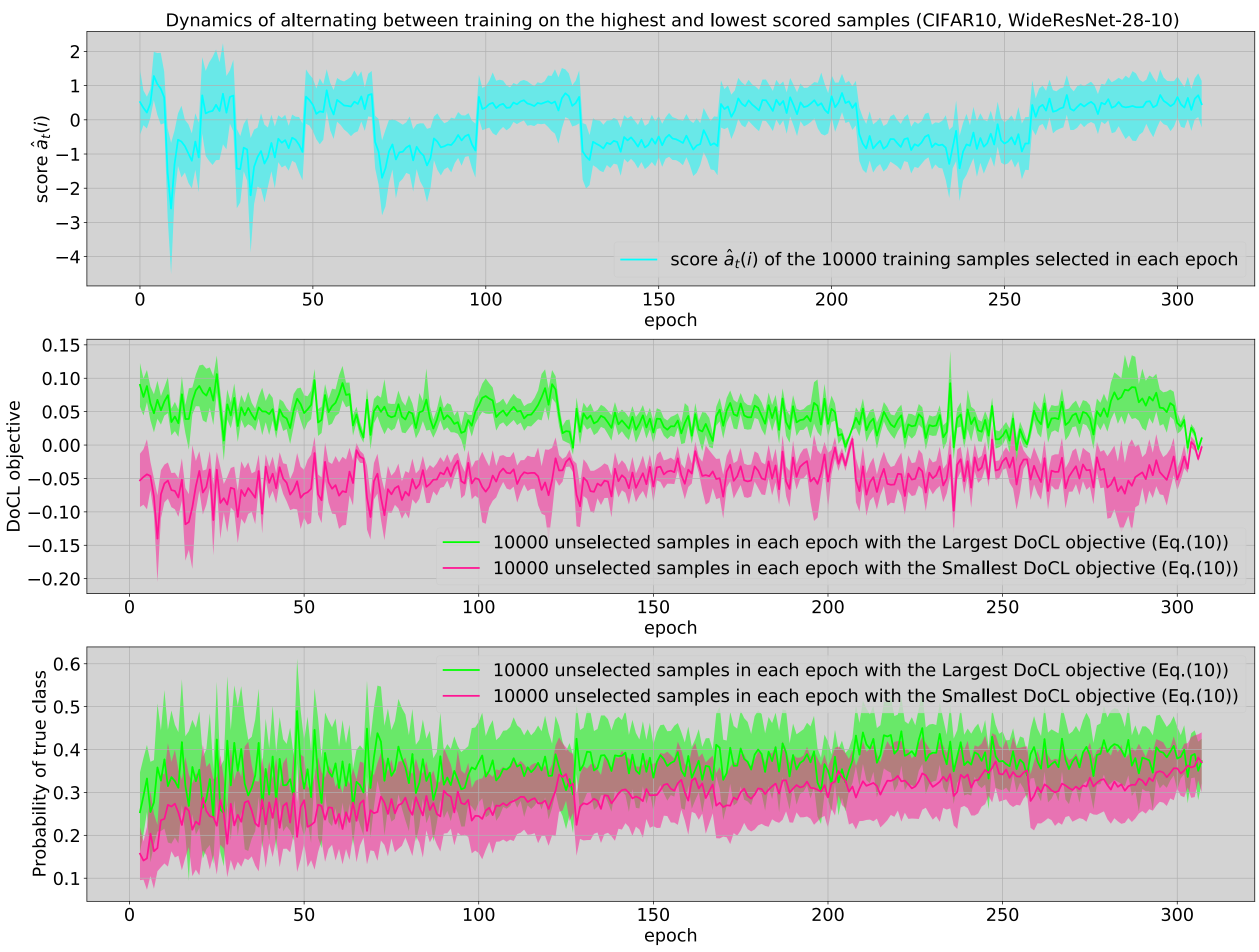

Figure 8: *Training alternates between the highest and lowest scored samples*: Dynamics (mean±std) for (**Top**) the score $\hat{a}(i)$ (Eq. (8)) of the selected samples, (**Middle**) the DoCL objective (Eq. (10)) values of unselected samples, and (**Bottom**) the output true-class probability for unselected samples. We split the unselected samples in each epoch into two groups with the largest/smallest DoCL objective values.

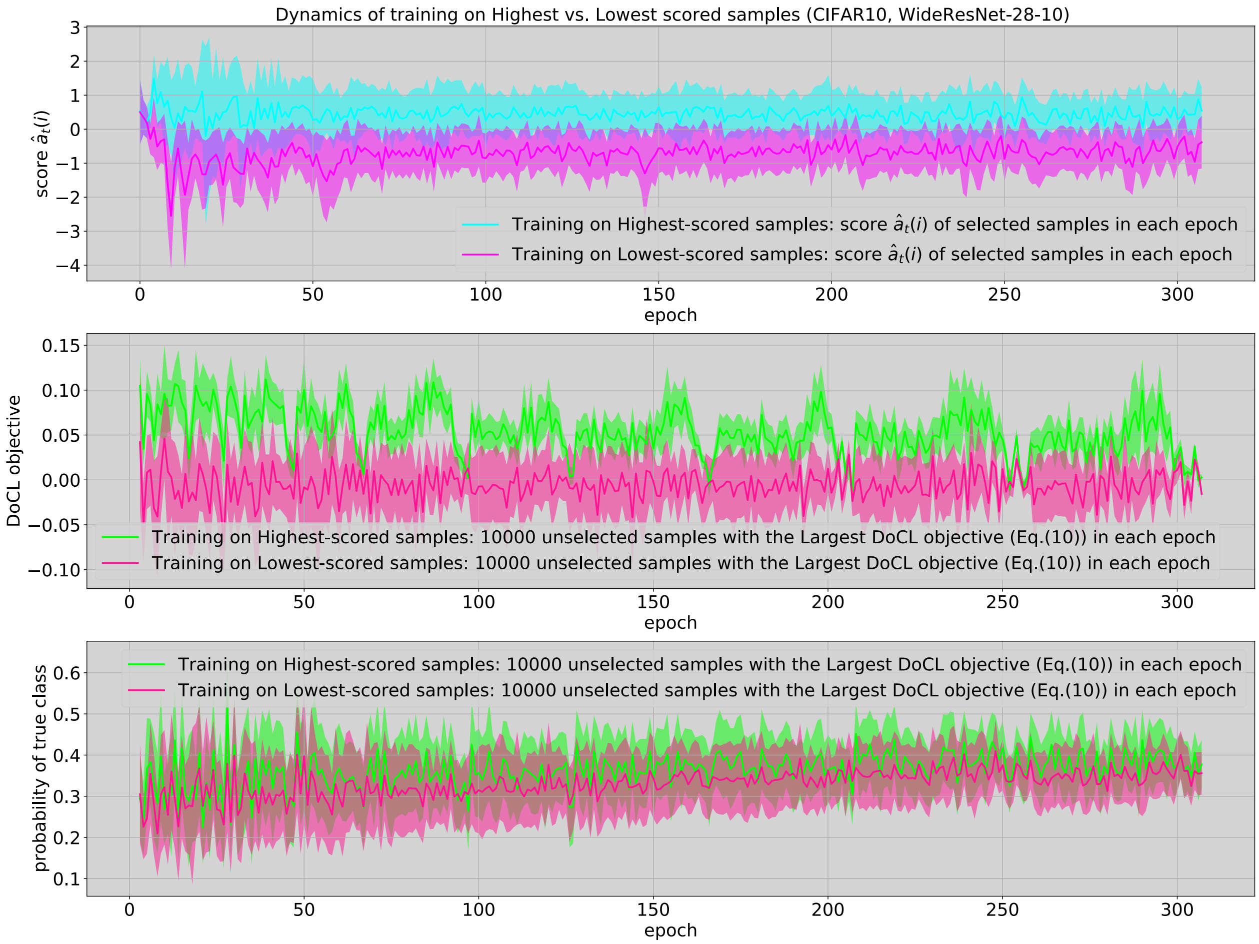

Figure 9: *Training with highest-scored vs. lowest-scored samples*: Dynamics (mean±std) for (**Top**) the score $\hat{a}(i)$ (Eq. (8)) of the selected samples, (**Middle**) the DoCL objective (Eq. (10)) values and (**Bottom**) the output true-class probability for unselected samples with the largest DoCL objective values.

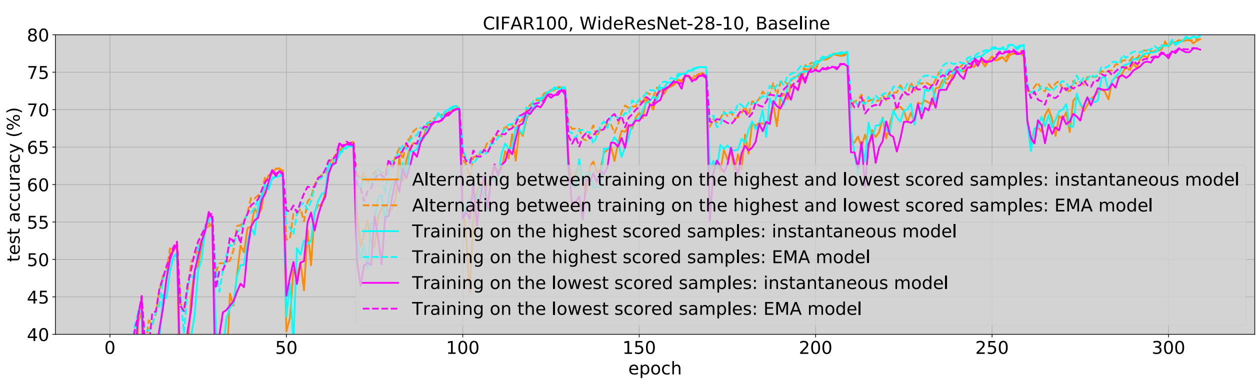

Figure 10: Test set accuracy of WideResNet-28-10 (instantaneous model) and its exponential moving average (EMA model) during the course of training when using the three data selection curricula.

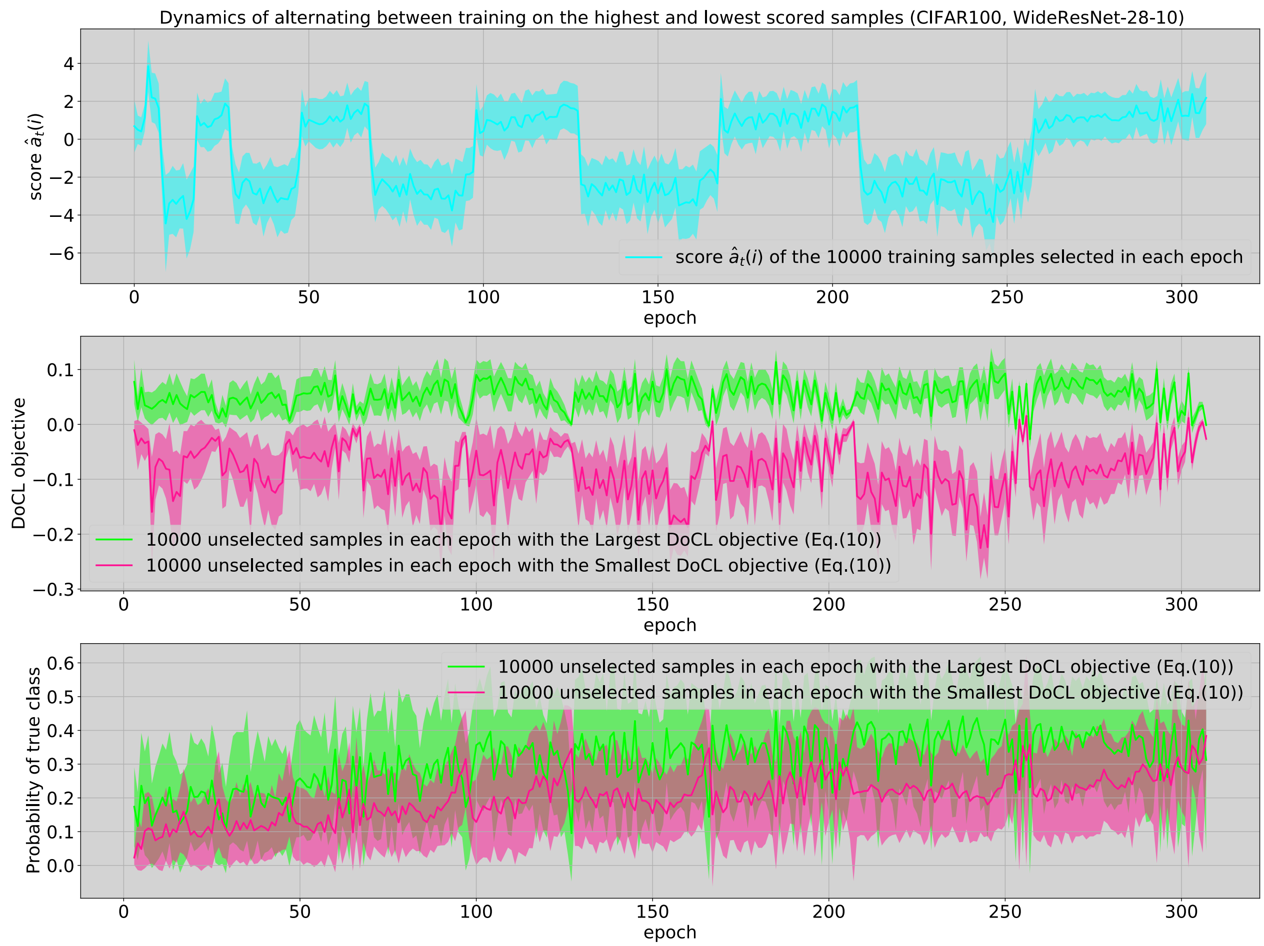

Figure 11: *Training alternates between the highest and lowest scored samples*: Dynamics (mean±std) for (**Top**) the score $\hat{a}(i)$ (Eq. (8)) of the selected samples, (**Middle**) the DoCL objective (Eq. (10)) values of unselected samples, and (**Bottom**) the output true-class probability for unselected samples. We split the unselected samples in each epoch into two groups with the largest/smallest DoCL objective values.

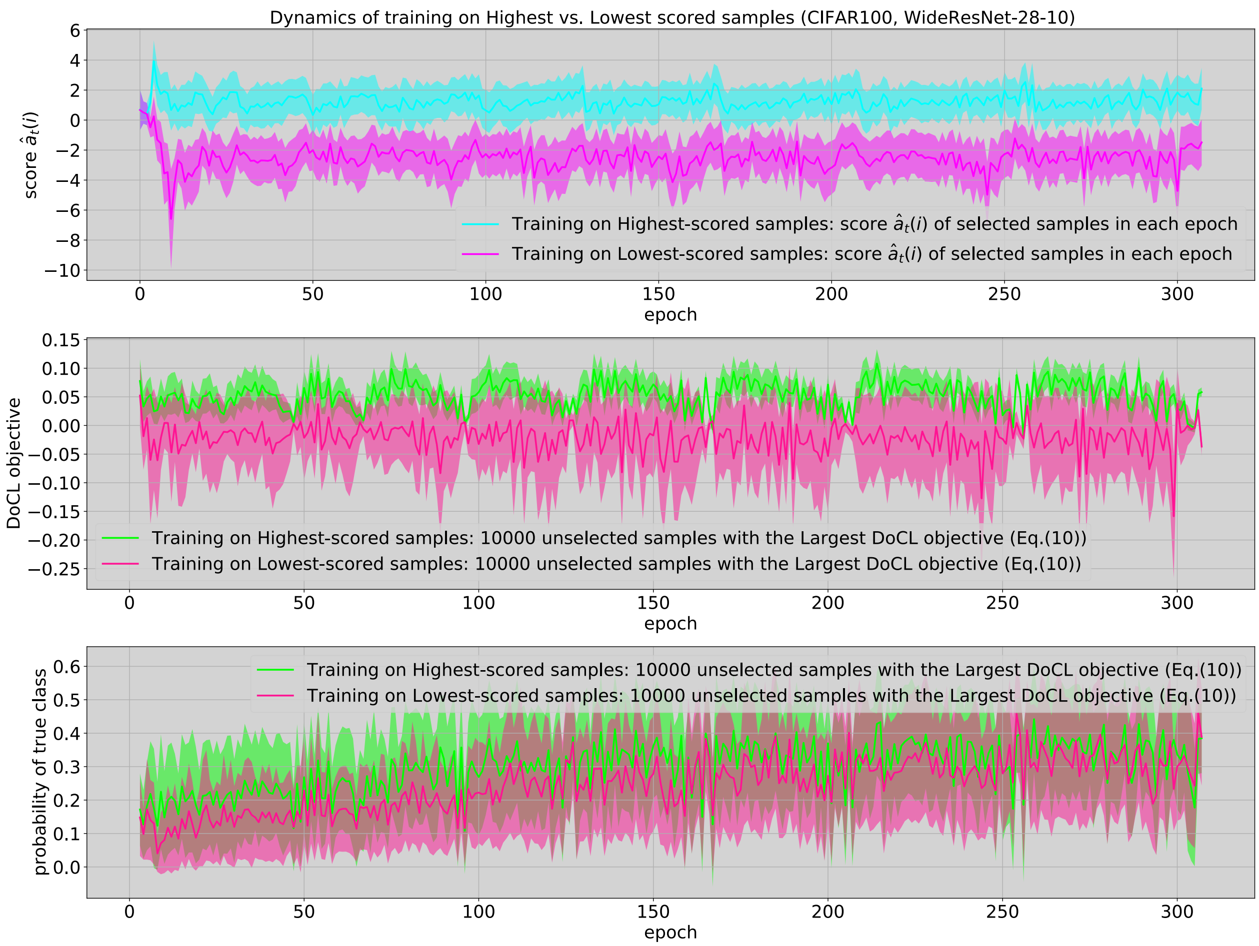

Figure 12: *Training with highest-scored vs. lowest-scored samples*: Dynamics (mean±std) for (**Top**) the score $\hat{a}(i)$ (Eq. (8)) of the selected samples, (**Middle**) the DoCL objective (Eq. (10)) values and (**Bottom**) the output true-class probability for unselected samples with the largest DoCL objective values.

B More Experimental Details for Section 5

On each dataset, for all the methods, we use the same cosine annealing learning rate schedule for multiple episodes. The ending epochs of cycles $\{T_i\}_{i=0}^{\kappa}$ in our learning rate schedule used on different datasets are listed below.

- CIFAR10, CIFAR100, SVHN, FMNIST: $(5, 10, 15, 20, 30, 40, 60, 90, 140, 210, 300)$;
- ImageNet: $(5, 10, 15, 20, 30, 45, 75, 120, 200)$;
- Food-101, Birdsnap, FGVCaircraft, StanfordCars (double the ImageNet cycles): $(10, 20, 30, 40, 60, 90, 150, 240, 400) = 2 \times (5, 10, 15, 20, 30, 45, 75, 120, 200)$;

Table 2: Details regarding the datasets and training settings (#Feature denotes the number of features after cropping if applied), "lr_start" and "lr_target" denote the starting and target learning rate for the first episode of cosine annealing schedule, they are gradually decayed over the rest episodes.

Dataset	CIFAR10	CIFAR100	Food-101	ImageNet	SVHN
#Training	50000	50000	75750	1281167	73257
#Test	10000	10000	25250	50000	26032
#Feature	$(3, 32, 32)$	$(3, 32, 32)$	$(3, 224, 224)$	$(3, 224, 224)$	$(3, 32, 32)$
#Class	10	100	101	1000	10
#Epoch T	300	300	400	200	300
BatchSize	128	128	80	256	128
lr_start	2×10^{-1}	2×10^{-1}	2×10^{-1}	2×10^{-1}	2×10^{-2}
lr_target	5×10^{-4}	5×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-3}
weight decay	1×10^{-4}	1×10^{-4}	1×10^{-5}	1×10^{-5}	1×10^{-4}

Dataset	Birdsnap	FGVCaircraft	StanfordCARs	FMNIST
#Training	47386	6667	8144	50000
#Test	2443	3333	8041	10000
#Feature	$(3, 224, 224)$	$(3, 224, 224)$	$(3, 224, 224)$	$(1, 28, 28)$
#Class	500	100	196	10
#Epoch T	400	400	400	300
BatchSize	258	256	256	128
lr_start	4×10^{-1}	4×10^{-1}	4×10^{-1}	4×10^{-2}
lr_target	1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-3}
weight decay	1×10^{-5}	1×10^{-5}	1×10^{-5}	1×10^{-4}

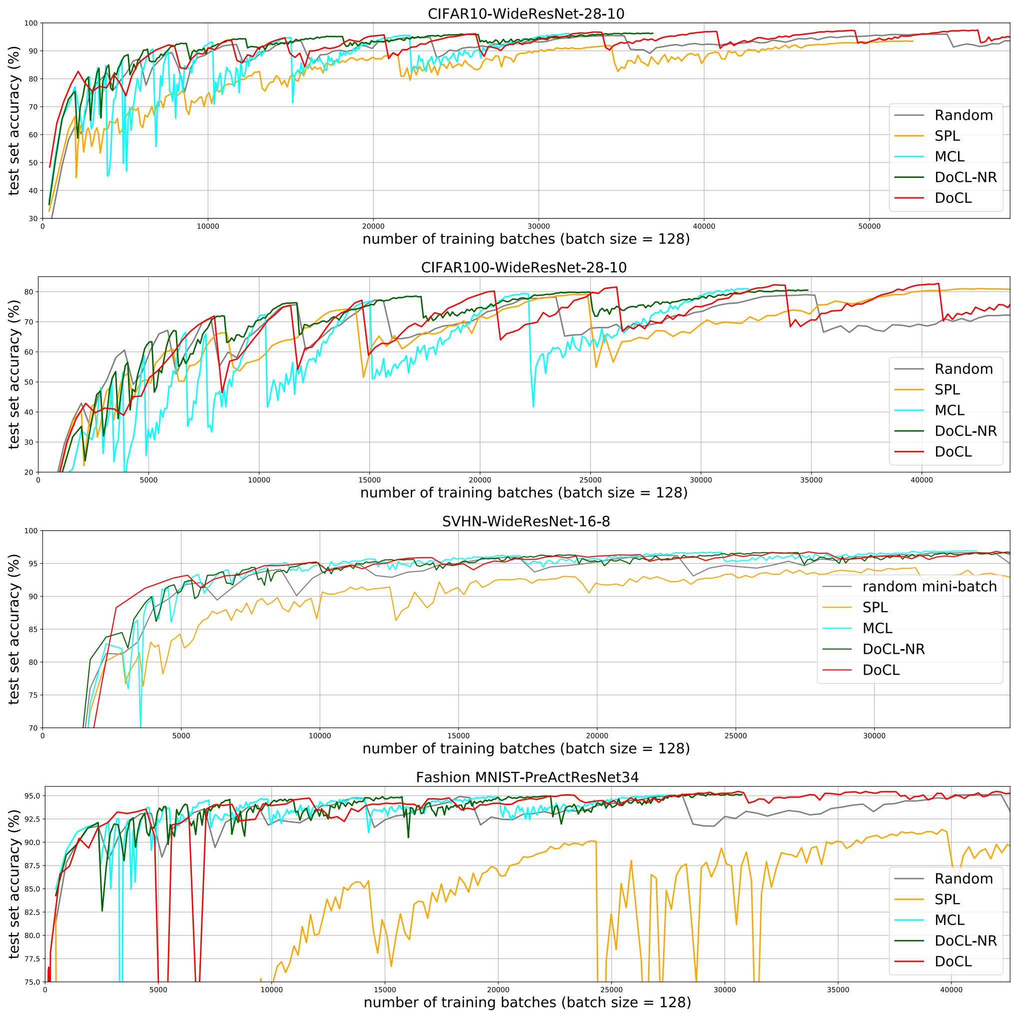

Figure 13: Training DNNs by using DoCL, DoCL-NR, SPL [31], MCL [59], and random mini-batch SGD on 3 datasets, i.e., CIFAR10, CIFAR100, SVHN and Fashion MNIST. We report how the test accuracy changes with the number of training batches for each method.

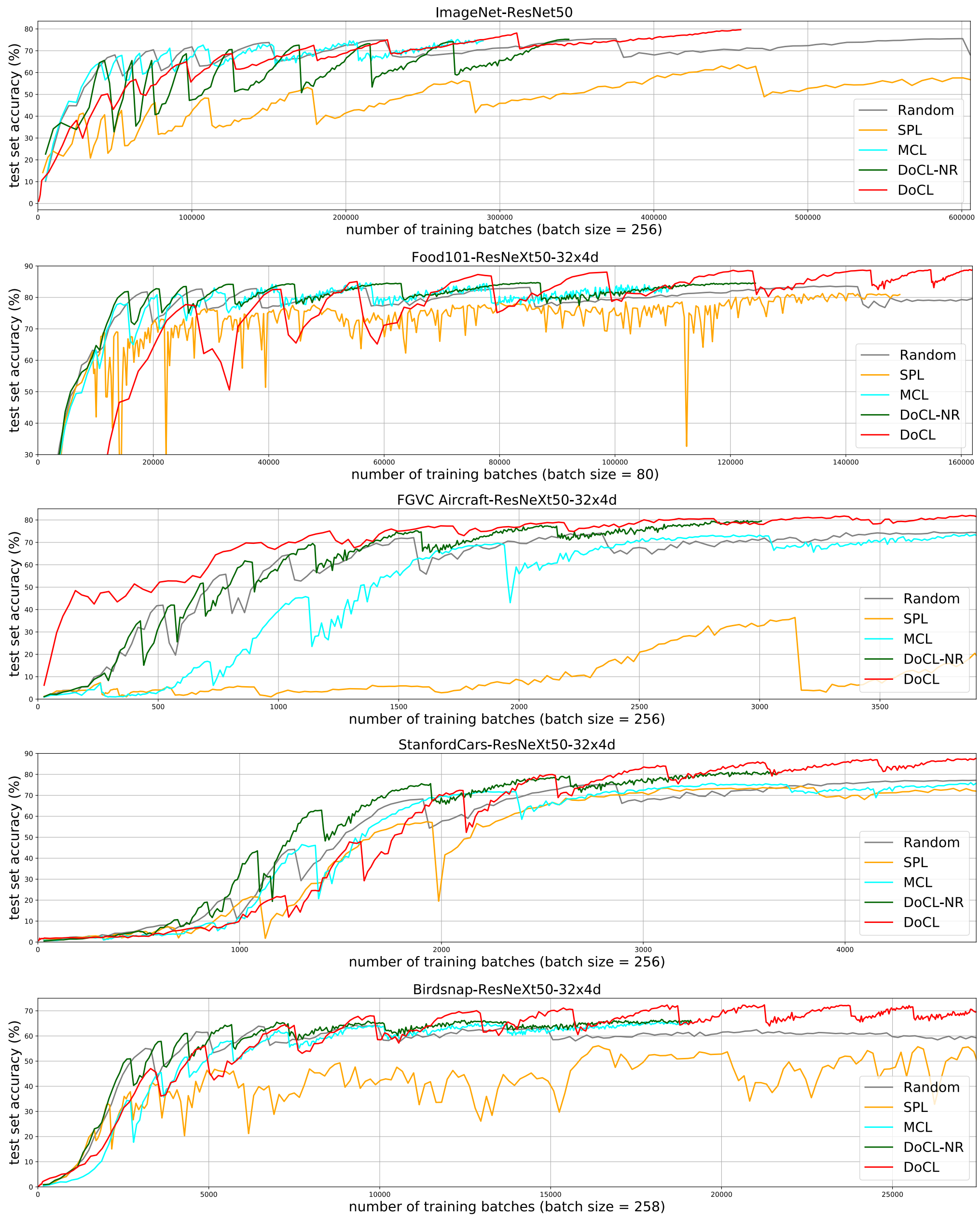

Figure 14: Training DNNs by using DoCL, DoCL-NR, SPL [31], MCL [59], and random mini-batch SGD on 3 datasets, i.e., ImageNet, Food101, FGVC Aircraft, Stanford Cars and Birdsnap. We report how the test accuracy changes with the number of training batches for each method.