A. Full results tables

We provide a full summary of our experiments on ImageNet in Table 3, with a dash "-" marking settings we did not deem necessary to run, as the cost outweights the potential insights.

Furthermore, Table 4 gives numerical results for the models shown in Figure 4, our best models (according to validation) on the four smaller datasets at 128px resolution, together with baselines and the teacher.

B. BiT models download statistics

In Figure 8, we show the download statistics for models with different sizes: ResNet50, ResNet50x3, ResNet101, ResNet101x3 and ResNet152x4. It's clear that the smallest ResNet50 model is the most used, with a significant gap compared to the other models. The practitioners' behavior motivates our work of getting the best possible ResNet50 model.

C. More consistency plots

In Figures 9 to 12, we show the "consistency" plots (cf Figure 3 in the main paper) for all datasets and across all training durations. It is noteworthy that (relatively) short runs may provide deceptive signal on the best method, and only with the addition of "patience", *e.g.* when distilling for a long time, does it become clear that the full function-matching approach is the best choice.

D. Shampoo optimization details

For all experiments the learning rate schedule was a linear warm-up up to 1800 steps followed by a quadratic decay towards zero. Overhead of Shampoo is quite minimal due blocking trick (each preconditioner is atmost 128x128) and inverse is run in a distributed manner across the TPU cores every step, with nesterov momentum. These settings are identical to the the training recipe in [1] for training a ResNet-50 architecture on ImageNet from scratch efficiently at large batch sizes. All experiments uses weight decay of 0.000375.

E. Training, validation and test splits

Throughout our experiments we rely on the *tensorflow datasets* library[3] to access all datasets. A huge advantage of this library is that it enables a unified and reproducible way to access diverse datasets. To this end, we report our *train*, *validation* and *test* splits (following the library's notation) in Table 5.

[3] https://www.tensorflow.org/datasets

Table 3. Summary of all ImageNet distillation runs. Numbers represent top-1 accuracy on the validation set. By default, the student is always a ResNet50 and the teacher is BiT-M-R152x2.

Experiment	**30ep**	**90ep**	**300ep**	**600ep**	**1200ep**	**4800ep**	**9600ep**
Best from labels	-	76.59	78.08	-	78.15	76.59	-
Fixed teacher	73.75	76.45	77.76	77.99	78.11	77.56	76.95
consistent teacher	74.95	78.05	80.08	80.63	81.15	81.58	81.76
function matching (FunMatch)	73.89	78.00	80.30	81.17	81.54	82.18	82.31
consistent teacher 🚿🕯️	75.45	78.79	80.54	81.11	81.44	-	-
function matching 🚿🕯️	75.12	78.70	80.63	-	81.67	-	-
T224 → S160 (consistent teacher)	71.38	75.57	78.01	-	-	-	-
T224 → S160 (function matching)	70.22	75.34	78.17	79.07	79.61	0.10	80.49
FunMatch: T384 → S224	-	-	80.46	-	81.82	82.33	82.64
FunMatch: T384+224 → S224	-	-	-	-	82.12	82.71	82.82
FunMatch: MobileNet v3 (GN)	-	-	74.60	-	76.31	76.84	76.97
FunMatch: MobileNet v3 (GN, 2T)	-	-	74.85	-	76.51	-	-
FunMatch: MobileNet v3 (GN, Small)	-	-	65.61	-	67.57	-	-
FunMatch: MobileNet v3 (BN)	-	-	72.32	-	73.51	-	-
FunMatch: MobileNet v3 (BN, 2T)	-	-	73.28	-	-	-	-
Figure 5 (right): BiT-M-R50 init	77.52	79.43	80.47	80.83	81.11	81.45	-
Figure 7: SGDM	-	76.59	76.38	-	74.93	73.48	-
Figure 7: Adam	-	74.92	74.55	-	73.47	70.66	-
Figure 7: SGDM + Mixup	-	76.18	78.06	-	75.01	71.40	-
Figure 7: Adam + Mixup	-	76.17	78.08	-	78.15	76.59	-

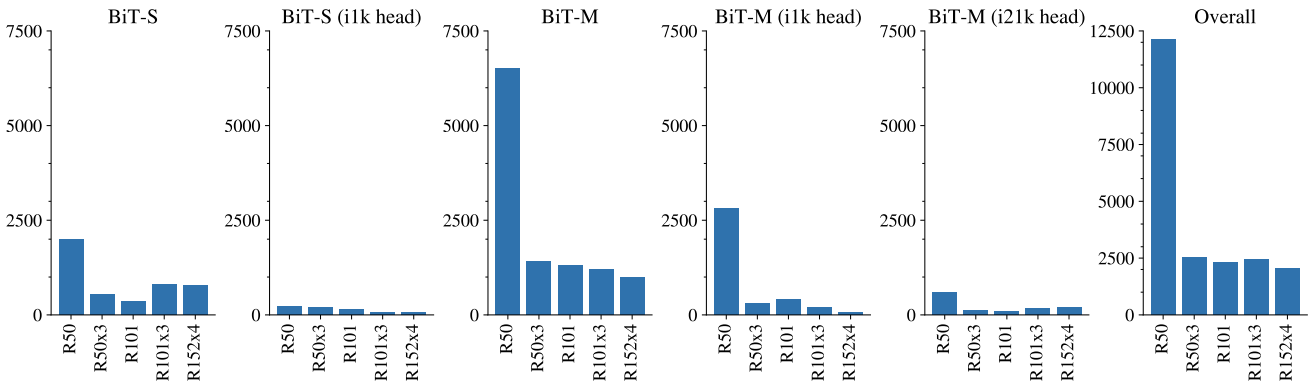

Figure 8. BiT models download statistics according to https://tfhub.dev/google/collections/bit. "BiT-S"/"BiT-M" denotes the BiT model for feature extraction, while the figures with a mention of "head" correspond to the classifiers. The rightmost overall plot shows the total download counts for each size. It is clear that ResNet-50 is by far the most widely used model.

Table 4. Tabular representation of the results from Figure 4.

Model	Epochs	Final Test Acc	T	LR	WD
		Flowers102			
ResNet-50x1 student	1000	77.51%	10	0.003	0.001
ResNet-50x1 student	10 000	92.83%	10	0.003	0.0003
ResNet-50x1 student	100 000	95.54%	1	0.001	0.0001
ResNet-50x1 student	1 000 000	96.93%	1	0.0003	1e-05
ResNet-152x2 teacher	-	97.82%	-	-	-
Best transfer ResNet50	10 000	97.50%	LR=0.01, Mixup=0.0		
Best from-scratch ResNet50	10 000	66.38%	LR=0.01, Mixup=1.0		
		Pet37			
ResNet-50x1 student	300	82.75%	2	0.01	1e-05
ResNet-50x1 student	1000	88.01%	5	0.01	0.001
ResNet-50x1 student	3000	90.08%	10	0.003	0.0003
ResNet-50x1 student	10 000	90.98%	2	0.001	0.0001
ResNet-50x1 student	30 000	91.06%	2	0.003	1e-05
ResNet-152x2 teacher	-	91.03%	-	-	-
Best transfer ResNet50	10 000	88.20%	LR=0.001, Mixup=1.0		
Best from-scratch ResNet50	10 000	74.24%	LR=0.01, Mixup=1.0		
		Food101			
ResNet-50x1 student	100	83.29%	10	0.01	0.001
ResNet-50x1 student	1000	86.64%	10	0.001	0.0003
ResNet-50x1 student	3000	87.20%	5	0.01	0.0001
ResNet-152x2 teacher	-	86.24%	-	-	-
Best transfer ResNet50	1000	85.05%	LR=0.001, Mixup=1.0		
Best from-scratch ResNet50	1000	74.56%	LR=0.01, Mixup=1.0		
		Sun397			
ResNet-50x1 student	100	68.28%	10	0.01	0.001
ResNet-50x1 student	1000	73.46%	10	0.003	0.0001
ResNet-50x1 student	3000	74.26%	10	0.01	3e-05
ResNet-152x2 teacher	-	74.22%	-	-	-
Best transfer ResNet50	1000	71.61%	LR=0.001, Mixup=1.0		
Best from-scratch ResNet50	1000	60.63%	LR=0.01, Mixup=1.0		

Table 5. *Train*, *validation* and *test* splits. Split definitions follow notation from the *tensorflow datasets* library and can be directly used to access relevant data splits using the library.

Dataset	train split	validation split	test split
Flowers102	`train`	`validation`	`test`
Pets37	`train[:90%]`	`train[90%:]`	`test`
Food101	`train[:90%]`	`train[90%:]`	`test`
Sun397	`train`	`validation`	`test`
ImageNet	`train[:98%]`	`train[98%:]`	`validation`

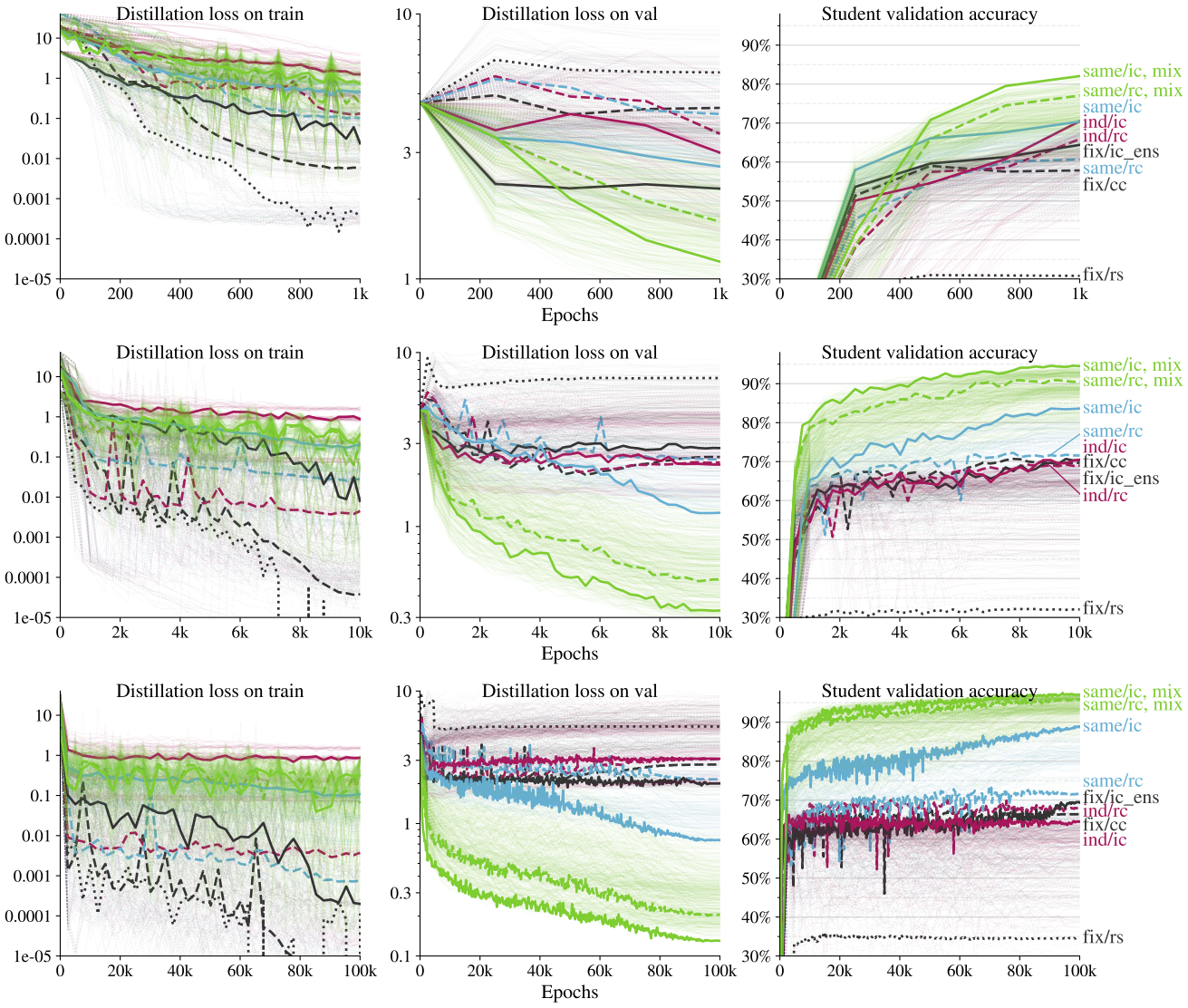

Figure 9. Consistency plots for the Flowers102 dataset, when training for 1 000 epochs, 10 000 epochs, and 100 000 epochs, from top to bottom respectively.

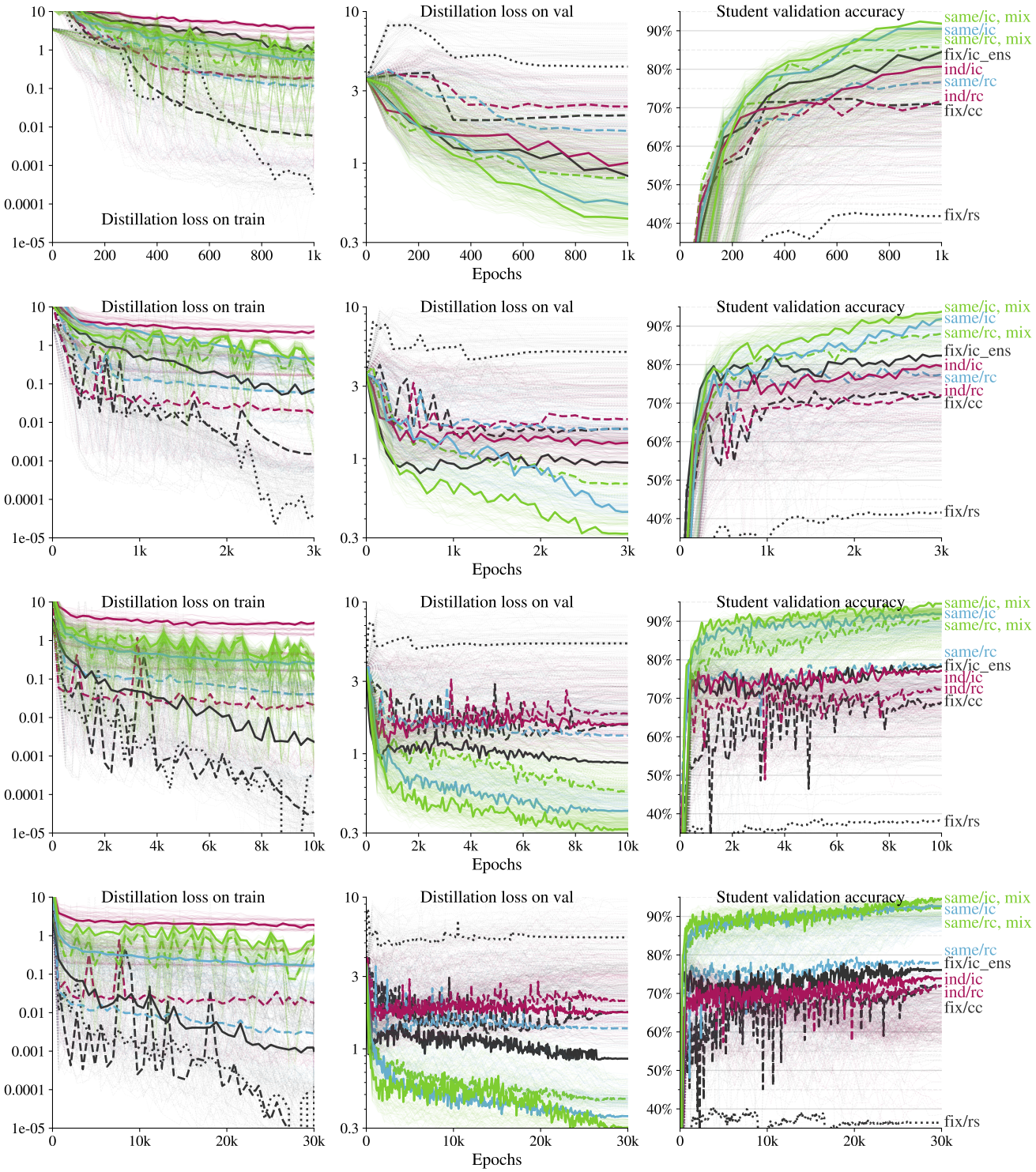

Figure 10. Consistency plots for the Pet37 dataset, when training for 1 000 epochs, 3 000 epochs, 10 000 epochs, and 30 000 epochs, from top to bottom respectively.

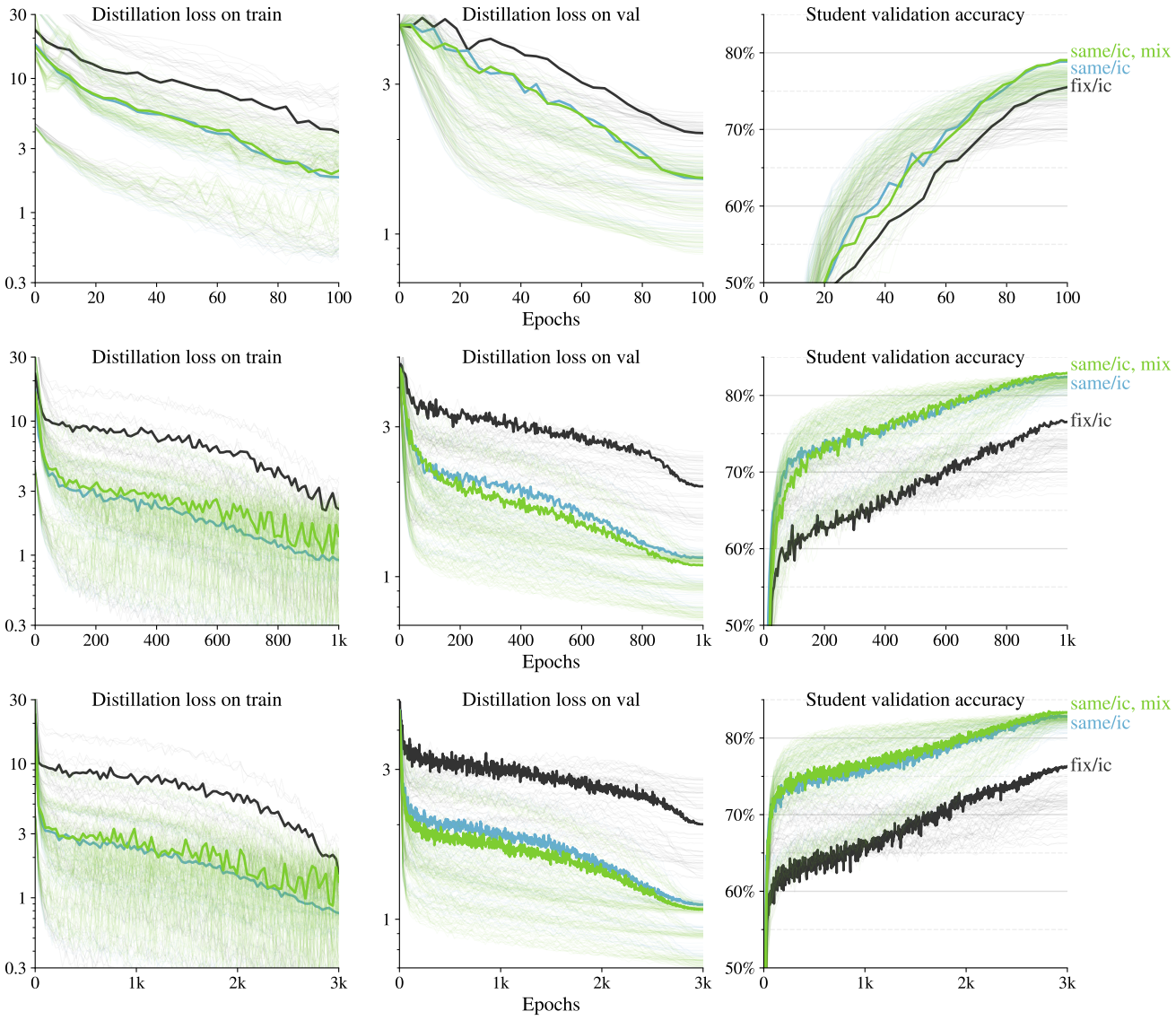

Figure 11. Consistency plots for the Food101 dataset, when training for 100 epochs, 1 000 epochs, and 3 000 epochs, from top to bottom respectively.

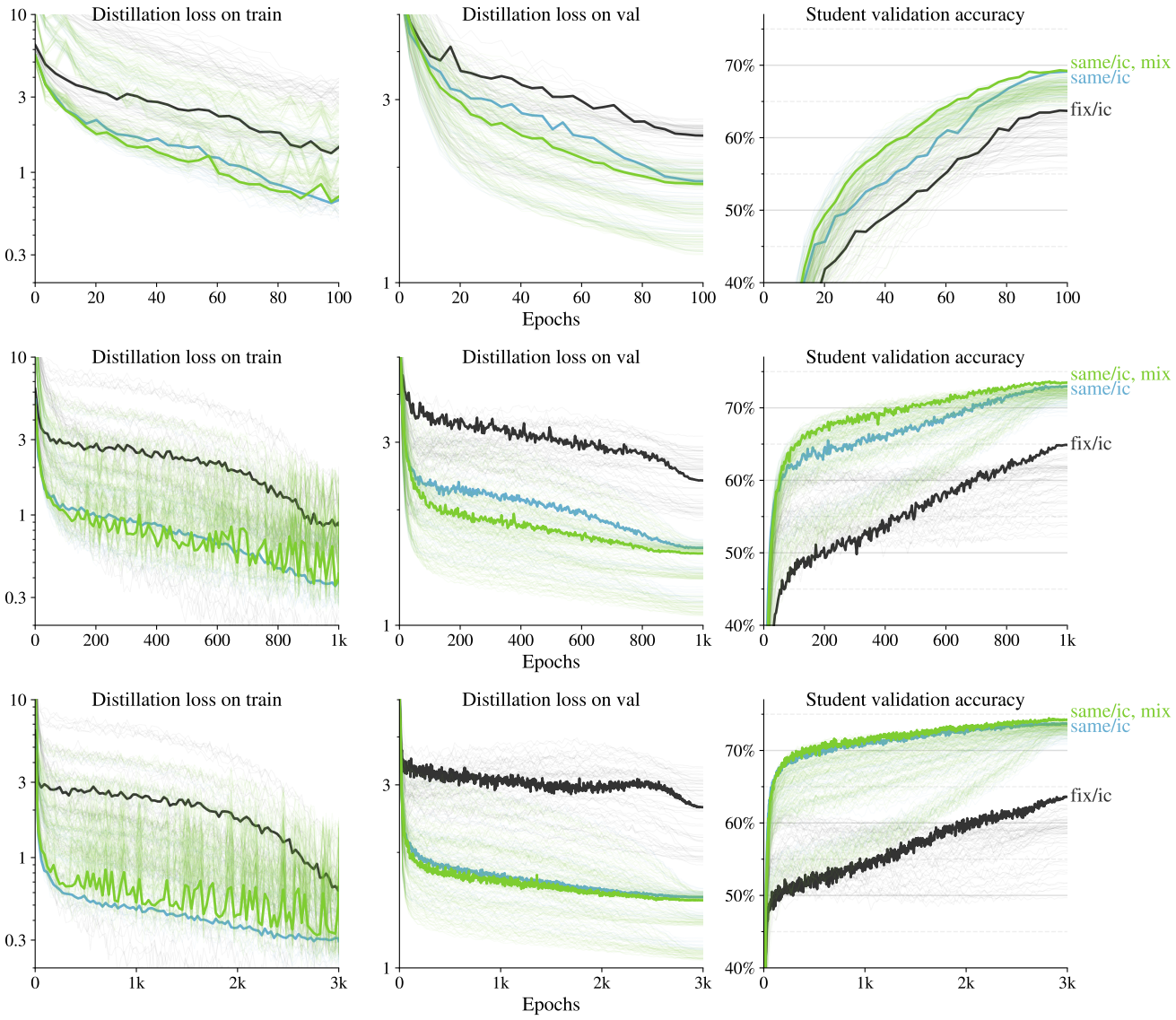

Figure 12. Consistency plots for the SUN397 dataset, when training for 100 epochs, 1 000 epochs, and 3 000 epochs, from top to bottom respectively.