Replication Study: Does Knowledge Distillation Really Work?

Rasul Alakbarli [1] **Mahammad Nuriyev** [1] **Petko Petkov** [1] **Dimitrije Ždrale** [1]

Abstract

We replicate key experiments from *"Does Knowledge Distillation Really Work?"* by Stanton et al. (NeurIPS 2021). The paper investigates a fundamental tension in knowledge distillation: while distilled students often achieve good generalization, they exhibit surprisingly poor *fidelity* to their teachers' predictions. We reproduce experiments on CIFAR-100 using PreResNet-20 architectures, examining self-distillation, ensemble distillation, initialization proximity, and representational similarity via Centered Kernel Alignment (CKA). Our results confirm the paper's central finding: knowledge distillation improves student generalization but transfers limited knowledge from teacher to student, with optimization difficulties being the primary cause of low fidelity. We additionally compute pairwise agreement between independently trained teachers to establish how much they already agree without distillation, finding that distillation improves fidelity by only $\sim$8 percentage points over this baseline. We document the practical challenges of replication with modern software stacks.

1. Introduction

Knowledge distillation (KD), introduced by Hinton et al. (2015) and rooted in earlier model compression work by Buciluǎ et al. (2006), is one of the most widely-used techniques for deploying neural networks in resource-constrained settings. The core idea is to train a smaller *student* network to emulate a larger, more capable *teacher* model by matching the teacher's soft output distribution rather than (or in addition to) the hard training labels. This technique has found broad application in model deployment, ensemble compression, and continual learning.

The conventional understanding of KD is that the student learns a high-fidelity representation of the teacher's knowledge through its soft labels, namely that the "dark knowledge" encoded in the teacher's output distribution over non-target classes provides rich supervisory signal that hard labels alone cannot capture (Hinton et al., 2015). This narrative has motivated numerous extensions, including intermediate representation matching (Romero et al., 2015), contrastive distillation (Tian et al., 2020), and self-distillation (Furlanello et al., 2018), each aiming to transfer more information from teacher to student.

However, Stanton et al. (2021) challenge this narrative by carefully distinguishing between two properties that are often conflated:

Generalization refers to the student's performance on unseen test data, measured by accuracy, negative log-likelihood (NLL), and expected calibration error (ECE). *Fidelity* refers to the degree to which the student's predictions match the teacher's, measured by top-1 agreement and predictive KL divergence.

The key finding is that while KD often improves generalization, it frequently fails to produce high-fidelity students, even when the student has identical capacity to the teacher (self-distillation). This contradicts the standard narrative: if the student is simply "learning the teacher's knowledge," it should reproduce the teacher's predictions. The fact that it does not suggests that KD's benefit may come from regularization or label smoothing rather than genuine knowledge transfer.

The authors investigate multiple hypotheses: (1) *insufficient data*, that perhaps more distillation data would improve fidelity; (2) *optimization difficulties*, that perhaps standard optimizers cannot find the teacher's solution; and (3) *inductive biases*, that perhaps differences in augmentation or architecture prevent matching. They ultimately conclude that optimization is the primary bottleneck, demonstrating a sharp phase transition in the loss landscape that determines whether the student converges to the teacher's basin.

These findings have significant practical implications. Model compression via distillation is widely used in industry: deploying ensemble predictions through a single student, compressing large language models for edge devices, and transferring knowledge between modalities. If the student does not faithfully reproduce the teacher's be-

[1] Course Project, Learning Theory and Advanced Machine Learning, February 2026. Correspondence to: Mahammad Nuriyev <mahammad.nuriyev@universite-paris-saclay.fr>.

havior, then the compressed model may have different failure modes, different calibration properties, and different biases than the original, even if aggregate metrics (like accuracy) look similar. For safety-critical applications, understanding whether distillation preserves the teacher's specific behavior (not just its aggregate performance) is essential.

In this replication study, we reproduce the core experiments using CIFAR-100 with PreResNet-20 architectures. We focus on the experiments that most directly support the paper's key claims: self-distillation fidelity (Figure 1a in the original), ensemble distillation fidelity (Figure 1b), student initialization proximity and loss landscape structure (Figure 6b), and CKA representational similarity analysis (Table 1). We additionally contribute an *additional baseline analysis* not present in the original work: computing pairwise agreement between independently trained teacher models. This baseline provides essential context for interpreting the fidelity gap; specifically, it reveals how much agreement improvement distillation provides beyond what independent training achieves.

2. Background

2.1. Knowledge Distillation

In the supervised classification setting with c classes, a classifier $f : \mathcal{X} \times \Theta \to \mathbb{R}^c$ produces logits defining a predictive distribution $\hat{p}(y = i|x) = \sigma_i(f(x, \theta))$ via the softmax σ.

Given a trained teacher with parameters θ_t, Hinton et al. (2015) proposed minimizing:

$$\mathcal{L}_s = \alpha \mathcal{L}_{\text{NLL}} + (1 - \alpha)\mathcal{L}_{\text{KD}}, \quad (1)$$

where $\mathcal{L}_{\text{NLL}} = -\log \hat{p}(y = y^*|x)$ is the standard cross-entropy and:

$$\mathcal{L}_{\text{KD}}(z_s, z_t) = -\tau^2 \sum_{j=1}^{c} \sigma_j\left(\frac{z_t}{\tau}\right) \log \sigma_j\left(\frac{z_s}{\tau}\right), \quad (2)$$

with z_s, z_t being student and teacher logits and $\tau > 0$ a temperature hyperparameter. The τ^2 factor ensures gradients scale appropriately. As $\tau \to \infty$, the softmax approaches a uniform distribution, and gradient information is dominated by relative logit magnitudes.

Following the original paper, we set $\alpha = 0$ in all main experiments to isolate the effect of distillation without confounding from hard labels. With $\alpha = 0$, any accuracy improvement is entirely attributable to the teacher's soft labels.

2.2. Metrics

We report both *generalization* and *fidelity* metrics. The generalization metrics are **top-1 accuracy** (fraction of test examples correctly classified), **negative log-likelihood** (NLL; average $-\log \hat{p}(y = y^*|x)$ on the test set, measuring the quality of the full predictive distribution), and **expected calibration error** (ECE; whether predicted confidences match true correctness probabilities, computed using 10 equal-width bins).

The fidelity metrics, which are the primary focus of this study, are **top-1 agreement** (fraction of test examples where teacher and student predict the same class: $\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}[\hat{y}_t(x_i) = \hat{y}_s(x_i)]$), **predictive KL** (average KL divergence from teacher to student: $\frac{1}{N}\sum_{i=1}^{N} \text{KL}(\hat{p}_t(\cdot|x_i) \| \hat{p}_s(\cdot|x_i))$, capturing differences in the full distribution), and **CKA** (Centered Kernel Alignment (Kornblith et al., 2019) between intermediate representations, measuring structural similarity in learned feature spaces). A perfect distillation would yield 100% agreement and zero KL divergence. The central question is how far real distillation falls short. Importantly, the paper argues that agreement and accuracy can be *negatively* correlated in the self-distillation setting: the student's best accuracy comes not from matching the teacher but from finding its own, different solution.

2.3. Ensemble Teachers

Deep ensembles (Lakshminarayanan et al., 2017) combine m independently trained models. The ensemble teacher logits are:

$$z_t = \log\left(\frac{1}{m}\sum_{i=1}^{m} \sigma(z_i)\right), \quad (3)$$

corresponding to the model-averaged distribution. Ensemble distillation is practically important because ensembles are expensive to deploy (Hinton et al., 2015).

2.4. Self-Distillation

In *self-distillation*, teacher and student share the same architecture and capacity. Furlanello et al. (2018) showed self-distillation can *improve* accuracy beyond the teacher, a phenomenon they called "Born Again Neural Networks." This creates a paradox: the student supposedly learns from the teacher yet outperforms it. Stanton et al. resolve this by showing the student is *not* faithfully learning the teacher; it improves precisely because it diverges from the teacher's predictions. The soft labels act as a form of regularization that guides the student to a different (and sometimes better) solution, rather than transferring the teacher's specific decision boundaries.

This perspective reframes self-distillation as a *training* *strategy* rather than a *knowledge transfer* method. The teacher provides useful training signal, but the student uses this signal to find its own solution rather than reproducing the teacher's. Understanding this distinction is crucial for practitioners who expect the distilled model to behave "like the teacher but smaller."

2.5. CKA: Centered Kernel Alignment

Two networks may produce similar predictions while organizing their internal representations differently, or conversely, learn similar representations yet disagree on outputs. CKA (Kornblith et al., 2019) quantifies the similarity of intermediate representations between networks, invariant to orthogonal transformations and isotropic scaling. Given the same inputs, we extract activation matrices from corresponding layers of the teacher and student and compute, for representation matrices X and Y:

$$\text{CKA}(X, Y) = \frac{\text{HSIC}(K_X, K_Y)}{\sqrt{\text{HSIC}(K_X, K_X) \cdot \text{HSIC}(K_Y, K_Y)}}, \tag{4}$$

where HSIC is the Hilbert-Schmidt Independence Criterion. CKA ranges from 0 (dissimilar) to 1 (identical up to invariances). We compute CKA at each of PreResNet-20's three residual stages.

3. Experimental Setup

3.1. Architecture

We use Pre-activation ResNet-20 (PreResNet-20) (He et al., 2016) throughout, with ∼0.27M parameters, three residual stages with [16, 32, 64] filters, and pre-activation batch normalization. Input size is 32 × 32 (CIFAR-100).

3.2. Training Configuration

Teachers are trained for 200 epochs with SGD (momentum=0.9, Nesterov, weight decay 10^{-4}), learning rate 0.1 with cosine annealing, batch size 256. We train 5 teachers with different random seeds.

Students are trained for 300 epochs with learning rate 5×10^{-2}, cosine annealing to 10^{-6}, SGD with Nesterov momentum, weight decay 10^{-4}, batch size 128. Default: temperature $\tau = 4$, $\alpha = 0$ (pure soft labels).

Data augmentation: random crops (32×32, 4px padding) and random horizontal flips. Pixels normalized to $[-1, 1]$ via unitcube normalization. These settings serve as the default configuration; individual experiments modify specific factors as described in their respective sections.

Table 1. Teacher model results (CIFAR-100, 200 epochs).

Model	Test Acc (%)	Train Acc (%)
Teacher 0	65.22	94.18
Teacher 1	65.34	93.86
Teacher 2	64.83	93.56
Teacher 3	64.63	93.36
Teacher 4	65.21	93.76
Mean	65.05 ± 0.27	93.74 ± 0.28

Table 2. Key deviations from the original setup.

Aspect	Original	Ours
Python	3.8	3.12
PyTorch	1.10+cu113	2.10+cu128
GPU	Not specified	RTX 4060 Ti
Norm (Exp. 3)	LayerNorm	BatchNorm
λ values	Dense sweep	Sparse

3.3. Teacher Training Results

We trained 5 PreResNet-20 teachers on CIFAR-100 (Table 1).

Our teacher accuracy (∼65%) is lower than the ∼70.5% in the original paper, likely due to PyTorch version differences (2.10 vs. 1.10) affecting cuDNN algorithms and batch normalization numerics. This gap does not affect qualitative findings about fidelity versus generalization, which are the paper's core contribution.

3.4. Deviations from Original Setup

The original paper's codebase targets Python 3.8 and PyTorch 1.10, which are difficult to reproduce exactly with current CUDA drivers and hardware. We use modern versions and document the resulting differences in Table 2.

A notable deviation concerns the initialization proximity experiment (Exp. 3) only: the original paper specifies LayerNorm for this experiment because BatchNorm statistics depend on the training data, making interpolation between models less straightforward. All other experiments use BatchNorm in both the original work and our replication. However, the released codebase uses BatchNorm throughout with no LayerNorm variant provided, and we treated the codebase as the source of truth. As we show in Figure 2, the phase transition is still clearly observed with BatchNorm, suggesting this deviation does not undermine the core finding. Additionally, the original paper sweeps λ densely to produce a smooth phase transition curve, whereas we test only a few values due to compute constraints, as each requires a full training run.

Table 3. Self-distillation results (3 trials). The student outperforms the teacher but with limited agreement.

	Acc (%)	Agree (%)	KL
Teacher	65.22	N/A	N/A
Student	68.62±0.17	71.36±0.27	0.854±0.005
Indep. teachers	65.05±0.27	63.19±0.33	1.672±0.019

3.5. Infrastructure

As no team member had access to a local GPU, we allocated a budget of 15 EUR to rent an RTX 4060 Ti machine through the vast.ai cloud platform. Each team member accessed the machine via SSH with a separate Linux user account, providing isolation and traceability for all operations. We used Git and GitHub for version control to coordinate the work.[1]

4. Experiments and Results

4.1. Experiment 1: Self-Distillation

We replicate the self-distillation setting (Figure 1a) where a PreResNet-20 teacher is distilled into an identical student. If distillation transferred knowledge perfectly, the student would be a functional copy of the teacher.

Analysis. The student scores higher on test accuracy than its teacher (68.62% vs. 65.22%), confirming the self-distillation improvement first reported by Furlanello et al. (2018). Yet the student only agrees with the teacher on 71.36% of test examples, meaning it gives a different prediction on nearly 29% of inputs despite outperforming the teacher overall. The KL divergence of 0.854 tells the same story: the two models' full probability distributions differ substantially.

To put these numbers in context, we compute *pairwise agreement between independently trained teachers* as a baseline (Table 4). Across all $\binom{5}{2} = 10$ teacher pairs, mean agreement is 63.19% with mean symmetric KL of 1.672. Distillation does improve fidelity over independent training: 8.2 percentage points more agreement, with KL roughly halved. Still, this improvement is modest. Distillation transfers some of the teacher's behavior, but the transfer is partial at best.

All 10 teacher pairs in Table 4 agree at 62.44% to 63.85%, a range of only 1.4pp. This narrow spread indicates that the agreement level is a property of the dataset itself (which examples are "easy" vs. "hard") rather than of the random seed. In other words, $\sim$63% agreement is what any two independently trained models achieve "for free" simply by learning from the same data. The distilled student's 71.36% agreement with its teacher is well above this baseline, confirming that the extra $\sim$8pp comes from distillation transferring information specific to that particular teacher, beyond what independent training provides.

Table 4. Pairwise agreement between independently trained teachers. This serves as a baseline for evaluating distillation fidelity: if distillation produces similar agreement to independent training, it adds no fidelity beyond what random training achieves.

Pair	Agreement (%)	Sym. KL
T0 vs T1	63.21	1.643
T0 vs T2	63.44	1.688
T0 vs T3	63.28	1.676
T0 vs T4	63.85	1.649
T1 vs T2	63.21	1.673
T1 vs T3	63.09	1.678
T1 vs T4	63.02	1.657
T2 vs T3	62.44	1.709
T2 vs T4	63.20	1.684
T3 vs T4	63.18	1.667
Mean	63.19 ± 0.33	1.672 ± 0.019

The student additionally achieves NLL of 20.15 and ECE of 0.14, indicating imperfect calibration consistent with the literature (Tang et al., 2020). The high ECE suggests that the student's confidence estimates are not well-calibrated, which has practical implications for applications where reliable uncertainty estimates are important (e.g., medical diagnosis, autonomous driving).

Figure 1 reveals the training dynamics in detail. Several observations are noteworthy:

Early commitment. Agreement reaches $\sim$70% within the first 50 epochs (out of 300) and barely improves thereafter. This suggests the student settles into its own basin of the loss landscape early in training, and subsequent epochs refine the solution within that basin rather than moving it toward the teacher's. The learning rate schedule (cosine annealing from 5×10^{-2} to 10^{-6}) shrinks the step size over time, making a jump between basins increasingly unlikely.

Accuracy continues to improve after agreement plateaus. While agreement stalls at $\sim$71%, student accuracy continues improving from $\sim$65% to $\sim$69%. This directly demonstrates that the student's accuracy gains come from improving *within* its own basin rather than from moving *closer* to the teacher's basin. The student finds a better solution than the teacher but one that produces different predictions on $\sim$29% of test examples.

Low cross-trial variance. The three trials show remarkably consistent trajectories (shaded bands in Figure 1), indicating that the training dynamics are highly deterministic given the same teacher. Different random initializations

[1] Repository: `https://github.com/MahammadNuriyev62/gnosis`

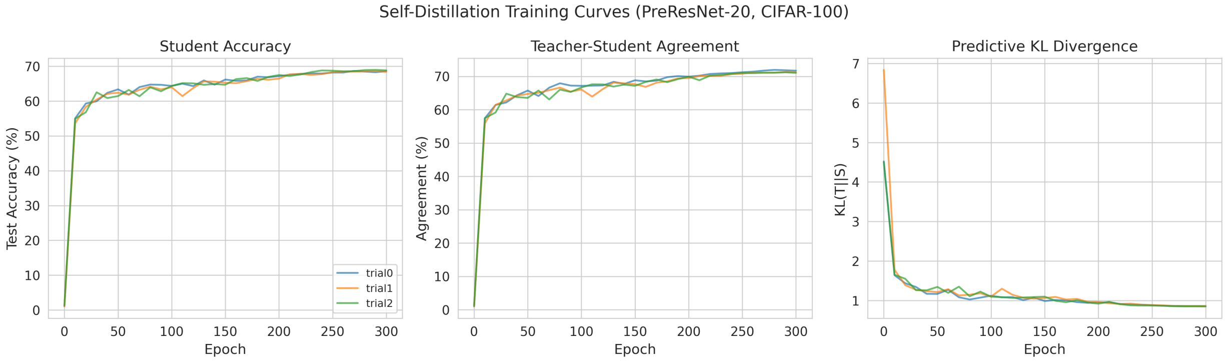

Figure 1. Self-distillation training curves (3 trials). **Left**: Student accuracy converges above teacher (∼69% vs. 65%). **Center**: Agreement plateaus at ∼71%, showing continued training does not improve fidelity. **Right**: KL drops rapidly then plateaus at ∼0.85, far from zero. These dynamics reveal that the student quickly settles into a distinct basin.

Table 5. Ensemble distillation results (1 trial each).

Teacher	T. Acc	S. Acc	Agree	KL
1-teacher	65.22	68.62	71.36	0.854
3-teacher	71.56	69.88	78.23	0.574
5-teacher	73.20	69.79	79.86	0.468

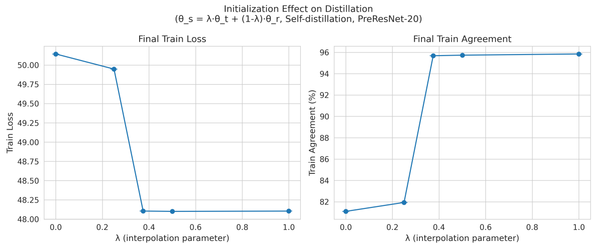

Figure 2. Initialization proximity vs. agreement. A phase transition at $\lambda \approx 0.3$ to 0.4 reveals distinct basins in the loss landscape.

lead to different solutions that are equally distant from the teacher; the basin structure is robust to initialization randomness.

4.2. Experiment 2: Ensemble Distillation

We distill 3-component and 5-component teacher ensembles into single students, replicating Figure 1b. Here the teacher is strictly more capable than any individual student.

Table 5 reveals several trends. First, ensemble teachers are substantially more accurate than individual models: the 3-teacher ensemble reaches 71.56% and the 5-teacher ensemble 73.20%, compared to 65.22% for a single teacher. Second, agreement increases with ensemble size: from 71.36% (1-teacher) to 78.23% (3-teacher) to 79.86% (5-teacher), while KL divergence decreases correspondingly from 0.854 to 0.574 to 0.468. This confirms the paper's finding that fidelity and generalization become positively correlated in the ensemble setting, unlike self-distillation where the student outperforms the teacher while disagreeing with it.

However, the student's accuracy (∼69.8%) remains well below the 5-teacher ensemble (73.20%), and agreement plateaus near 80%, far from perfect. The fidelity gap persists even with stronger teachers, confirming that the optimization barrier is a general phenomenon, not an artifact of the self-distillation setup.

Unlike self-distillation, where low fidelity can be harmless (the student finds a better solution), low fidelity in ensemble distillation represents a genuine loss: the student cannot capture the ensemble's diversity-driven improvements. Additionally, ensemble distributions encode disagreement among committee members, making them richer but potentially *harder* for a single student to match.

4.3. Experiment 3: Initialization Proximity

This experiment (replicating Figure 6b) is perhaps the most revealing. We initialize the student as a convex combination:

$$\theta_s = \lambda \cdot \theta_t + (1 - \lambda) \cdot \theta_r, \tag{5}$$

where θ_t are the teacher's trained parameters, θ_r is a random initialization, and $\lambda \in [0, 1]$ controls proximity. At $\lambda = 0$ (standard distillation), the student starts randomly. At $\lambda = 1$, it starts at the teacher's weights. We scale the learning rate as $\text{lr} = \text{lr}_0 \cdot (1 - \lambda)$.

Analysis. The original paper shows a sharp transition at $\lambda \approx 0.3$ to 0.4: below this threshold, the student converges to a basin far from the teacher (∼80% train agreement); above it, the student remains in the teacher's basin (>95% agreement). This phase transition is the most compelling evidence for the optimization hypothesis.

Our results confirm the phase transition. At $\lambda = 0.0$ and $\lambda = 0.25$, the student converges to a distant basin

Table 6. CKA analysis: random vs. teacher initialization.

Init.	Agree	KL	CKA$_1$	CKA$_2$	CKA$_3$
Random	72.32	0.781	0.871	0.915	0.985
Teacher	72.92	0.746	0.935	0.932	0.985

(test agreement 72.14% and 72.65%, respectively). At $\lambda = 0.375$, agreement jumps sharply to 96.51%, and remains high at 96.92% ($\lambda = 0.5$) and 98.02% ($\lambda = 1.0$). The transition occurs between $\lambda = 0.25$ and $\lambda = 0.375$, precisely in the range (0.3 to 0.4) predicted by the paper. Above the threshold, accuracy drops to ∼65.5%, matching the teacher's 65.22%, confirming that the student stays in the teacher's basin at the cost of not finding its own (better) solution.

Note that we use BatchNorm throughout, unlike the paper's LayerNorm. BatchNorm statistics (running mean and variance) are not straightforward to interpolate, potentially introducing confounds at intermediate λ values. Despite this, the phase transition is clearly observed, confirming the fundamental claim about loss landscape structure.

4.4. Experiment 4: CKA Analysis

We compare students initialized randomly versus from teacher weights (replicating Table 1), using CKA at each of the three residual stages.

Table 6 shows that CKA values are high in both conditions (0.871 to 0.985), indicating strong representational similarity between teacher and student regardless of initialization. Teacher initialization increases CKA at stage 1 (0.935 vs. 0.871) and stage 2 (0.932 vs. 0.915), but agreement barely changes: 72.92% vs. 72.32%, a difference of only 0.6pp. KL divergence is similarly unaffected (0.746 vs. 0.781). This confirms the paper's finding: sharing an initialization increases representational similarity but does *not* meaningfully improve fidelity.

This disconnect between CKA and agreement has theoretical significance. Networks can learn nearly identical internal representations while producing different outputs, suggesting the mapping from representations to predictions (the final classification head) introduces substantial variability. If CKA similarity guaranteed functional similarity, then distillation methods targeting intermediate representations (e.g., FitNets (Romero et al., 2015)) would also produce high agreement. Our results confirm that this is not the case, motivating future work on methods that explicitly target functional similarity rather than representational alignment.

Table 7. Temperature and augmentation effects on distillation fidelity (1 trial each, 5-teacher ensemble). All rows use the same teacher ensemble for a fair comparison.

Setting	Acc (%)	Agree (%)	KL
$\tau = 1$, no MixUp	69.02	75.60	0.568
$\tau = 4$, no MixUp	69.84	80.51	0.465
$\tau = 4$, MixUp	68.76	80.36	0.421

4.5. Supplementary: Temperature and Augmentation Effects

We additionally investigate the effect of temperature and data augmentation on fidelity, replicating aspects of Figure 3 in the original paper. These experiments test the *inductive bias* hypothesis: that choices like temperature and augmentation might be responsible for the fidelity gap.

We train students under three conditions: (1) $\tau = 1$, where the teacher's softmax output is sharply peaked and the student sees little information about the teacher's uncertainty; (2) $\tau = 4$ (our default), where the softened output spreads probability across classes, revealing more of the teacher's internal ranking; and (3) $\tau = 4$ with MixUp augmentation ($\alpha_{\text{mixup}} = 1.0$) (Zhang et al., 2018), which blends pairs of training images and their labels to create synthetic examples that expose the student to more of the teacher's behavior across the input space.

The original paper finds that higher temperature and MixUp both improve agreement modestly, but neither eliminates the fidelity gap. Temperature $\tau = 4$ yields better fidelity than $\tau = 1$ because softer labels provide richer gradient information about the teacher's full distribution. MixUp provides additional training diversity but does not fundamentally change the optimization landscape structure.

Table 7 shows that higher temperature substantially improves fidelity: $\tau = 4$ achieves 80.51% agreement compared to 75.60% for $\tau = 1$, a gain of nearly 5pp, with KL decreasing from 0.568 to 0.465. Adding MixUp on top of $\tau = 4$ reduces KL further (0.421 vs. 0.465) but agreement is essentially unchanged (80.36% vs. 80.51%). This confirms the paper's finding: MixUp provides marginal improvement in distributional similarity but does not close the fidelity gap, consistent with the optimization hypothesis being the dominant factor. All three rows use the same 5-teacher ensemble, ensuring a fair comparison.

We additionally investigate the effect of optimizer choice and training duration (replicating Figure 6a). With SGD for 300 epochs (single teacher), the student achieves 70.88% agreement; doubling to 600 epochs yields 71.41%, a negligible improvement of 0.53pp. Switching from SGD to Adam at 300 epochs yields 70.61% agreement, comparable

Table 8. Comparison with original paper. Our independent teacher baseline is an additional contribution not present in the original work.

Metric	Original	Ours
Teacher Acc (%)	$\sim$70.5	65.22
Student Acc (%)	$\sim$71	68.62
Agreement (%)	$\sim$72	71.36
KL(T‖S)	$\sim$0.7	0.854
Indep. teacher Agree (%)	N/A	63.19
Indep. teacher KL	N/A	1.672

to SGD. Neither longer training nor a different optimizer closes the fidelity gap.

5. Discussion

5.1. Summary of Findings

Our replication confirms the central conclusions of Stanton et al. (2021): good accuracy does not imply good fidelity, optimization is the primary bottleneck, students initialized far from the teacher converge to different solutions regardless of training duration, and ensemble distillation aligns fidelity with generalization more than self-distillation does.

5.2. Quantitative Comparison with Original

While absolute accuracy values are lower ($\sim$65% vs. $\sim$70.5% for teachers), the *relative* patterns are preserved: student outperforms teacher by $\sim$3.4pp, and agreement is nearly identical at $\sim$71%. The accuracy gap between our setup and the original is consistent across both teacher and student, suggesting it reflects a uniform shift due to software environment rather than any fundamental difference in the distillation dynamics. Importantly, the *agreement* metric (which is the paper's core contribution) is almost identical ($\sim$71% vs. $\sim$72%), confirming that fidelity is robust to implementation details even when absolute accuracy is not.

The independent teacher baseline puts the fidelity gap in sharper perspective. Random models agree on 63% of test examples simply because CIFAR-100 has dominant visual patterns that any competent classifier captures. Distillation lifts agreement to 71%, a real but modest improvement. The remaining 29% disagreement reflects the optimization landscape structure that the paper identifies. To decompose this further: of the 10,000 CIFAR-100 test examples, approximately 6,319 are classified identically by any pair of independently trained models (the "easy" examples), approximately 817 additional examples are brought into agreement by distillation (the "transferred knowledge"), and approximately 2,864 examples remain in disagreement (the "fidelity gap").

5.3. Why Does the Student Outperform the Teacher?

As discussed in Section 2.4, soft labels act as regularization (Tang et al., 2020). More concretely, when the teacher assigns probability 0.7 to "cat," 0.1 to "lynx," and 0.05 to "tiger," these targets encode inter-class similarity that one-hot labels lack: the student learns that "cat" is closer to "lynx" than to "airplane." Soft labels also reduce memorization by discouraging the model from pushing logits for the correct class toward infinity. These benefits are largely independent of fidelity; the student improves because soft labels provide richer gradient signal, not because it faithfully reproduces the teacher.

5.4. The Optimization Landscape

Experiment 3 has broader implications for understanding optimization in deep learning (Allen-Zhu & Li, 2023). Recent work has suggested that different solutions found by gradient descent are often connected by paths of low loss in weight space. However, the sharp phase transition we observe suggests this does not hold between the teacher's and student's solutions: a high barrier separates them, and the student can only reach the teacher's solution if initialized close enough to cross it.

5.5. Critical Assessment of the Paper

While our replication broadly confirms the paper's conclusions, several aspects warrant critical examination:

Is 71% agreement actually low? Our independent baseline analysis provides a principled answer: two independently trained models agree only 63%. Distillation improves this by 8pp, which is meaningful but far short of perfect fidelity. The paper's central claim, that distillation fails to achieve high fidelity, is validated, though the picture is more nuanced than "distillation doesn't work." It works for generalization; it partially works for fidelity.

The $\alpha = 0$ choice. By setting $\alpha = 0$ (no hard labels), the paper maximizes the attribution of results to soft-label effects. However, in practice, $\alpha > 0$ is common. The paper briefly explores this but does not exhaustively characterize the α-fidelity relationship. It is possible that including hard labels (which provide the "correct" answer) helps the student converge to a more teacher-like solution.

Richness of soft labels. CIFAR-100's classes are relatively distinct from one another, so the teacher's soft labels say little beyond which class is most likely. When classes are more numerous or more similar, soft labels carry more useful information about the teacher's reasoning. Large language models, for instance, predict over tens of thousands of tokens, and each soft distribution says a great deal about which outputs the teacher considers plausible. The fidelity gap we observe here may be smaller in such settings.

The role of model capacity. All of our experiments use PreResNet-20 ($\sim$0.27M parameters) as both teacher and student. In practice, knowledge distillation is most commonly used to compress a large teacher into a smaller student, yet we do not test this setting. The paper's argument is that fidelity is low even when the student has full capacity to match the teacher, implying it can only be worse with a capacity gap. While this logic is sound, directly measuring fidelity in the capacity-gap setting would strengthen the practical relevance of these findings. It also remains unclear whether highly overparameterized networks, which may have flatter loss landscapes and more connected basins, would exhibit the same fidelity gap.

5.6. Implications for Practitioners

For practitioners who need *generalization*, standard KD works well, and the student will likely match or exceed the teacher's accuracy. However, for those who need *fidelity*, standard KD is insufficient, and novel optimization strategies, fidelity-aware objectives, or initialization tricks are needed. For *ensemble compression*, fidelity and accuracy are more aligned than in the self-distillation setting, but a significant gap remains. Finally, regarding *calibration*, the high KL divergence and ECE values suggest that calibration properties do not transfer well through distillation, which is relevant for applications requiring reliable uncertainty estimates.

5.7. Difficulties Encountered

Replication of ML experiments presents challenges beyond algorithmic correctness. **Environment compatibility** was a significant hurdle: the codebase targets Python 3.8 and PyTorch 1.10, and running on Python 3.12 with PyTorch 2.10 required fixing setuptools incompatibilities and handling torchtext ABI issues. **Numerical reproducibility** proved elusive: different PyTorch versions use different cuDNN algorithms, yielding $\sim$5pp accuracy differences even with identical seeds. **Computational constraints** were substantial: each 300-epoch distillation run takes $\sim$2 hours on an RTX 4060 Ti, making the full experimental suite a multi-day endeavor on a single GPU. We observed significant **GPU underutilization**: PreResNet-20 uses only $\sim$850 MiB ($\sim$5%) of GPU memory, with the bottleneck being CPU data loading rather than GPU compute; running 5 experiments in parallel improved throughput $\sim$3x while using only 13% of VRAM. Finally, **codebase assumptions** required attention: the original code uses Hydra for configuration management, and we bypassed it with a standalone experiment runner, which required understanding the codebase's internal APIs in detail.

These challenges underscore the value of containerized environments for exact reproduction and publishing qualitative claims that are robust to implementation details.

6. Conclusion

We replicated the core experiments from "Does Knowledge Distillation Really Work?" using different software (PyTorch 2.10 vs. 1.10) and hardware (RTX 4060 Ti) than the original, and confirmed all central findings. Our additional baseline of pairwise teacher agreement (63%) contextualizes the fidelity gap: distillation lifts agreement to 71%, a real but partial improvement.

Limitations. Our teacher accuracy ($\sim$65%) is lower than the original ($\sim$70.5%), which may affect the magnitude of fidelity metrics though not their direction. Our computational budget limited us to fewer trials for some experiments and a sparse λ sweep, reducing statistical power.

Despite these limitations, the qualitative consistency of our results across a substantially different software and hardware stack reinforces the robustness of the original findings and suggests they reflect fundamental properties of knowledge distillation rather than artifacts of a particular setup.

References

Allen-Zhu, Z. and Li, Y. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. In *International Conference on Learning Representations*, 2023.

Buciluă, C., Caruana, R., and Niculescu-Mizil, A. Model compression. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 535–541, 2006.

Furlanello, T., Lipton, Z. C., Tschannen, M., Itti, L., and Anandkumar, A. Born again neural networks. In *International Conference on Machine Learning*, pp. 1607–1616, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pp. 630–645, 2016.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529, 2019.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations*, 2015.

Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A. A., and Wilson, A. G. Does knowledge distillation really work? In *Advances in Neural Information Processing Systems*, volume 34, pp. 6906–6919, 2021.

Tang, J., Shivanna, R., Zhao, Z., Lin, D., Singh, A., Chi, E. H., and Jain, S. Understanding and improving knowledge distillation. *arXiv preprint arXiv:2002.03532*, 2020.

Tian, Y., Krishnan, D., and Isola, P. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.