Random Teachers are Good Teachers

Felix Sarnthein [1] Gregor Bachmann [1] Sotiris Anagnostidis [1] Thomas Hofmann [1]

Abstract

In this work, we investigate the implicit regularization induced by teacher-student learning dynamics in self-distillation. To isolate its effect, we describe a simple experiment where we consider teachers at random initialization instead of trained teachers. Surprisingly, when distilling a student into such a random teacher, we observe that the resulting model and its representations already possess very interesting characteristics; (1) we observe a strong improvement of the distilled student over its teacher in terms of probing accuracy. (2) The learned representations are data-dependent and transferable between different tasks but deteriorate strongly if trained on random inputs. (3) The student checkpoint contains sparse subnetworks, so-called lottery tickets, and lies on the border of linear basins in the supervised loss landscape. These observations have interesting consequences for several important areas in machine learning: (1) Self-distillation can work solely based on the implicit regularization present in the gradient dynamics without relying on any *dark knowledge*, (2) self-supervised learning can learn features even in the absence of data augmentation, and (3) training dynamics during the early phase of supervised training do not necessarily require label information. Finally, we shed light on an intriguing local property of the loss landscape: the process of feature learning is strongly amplified if the student is initialized closely to the teacher. These results raise interesting questions about the nature of the landscape that have remained unexplored so far. Code is available at `www.github.com/safelix/dinopl`.

1. Introduction

The teacher-student setting is a key ingredient in several areas of machine learning. Knowledge distillation is a common strategy to achieve strong model compression by training a smaller student on the outputs of a larger teacher model, leading to better performance compared to training the small model on the original data only (Bucilă et al., 2006; Ba & Caruana, 2013; Hinton et al., 2015; Polino et al., 2018; Beyer et al., 2022). In the special case of self-distillation, where the two architectures match, it is often observed in practice that the student manages to outperform its teacher (Yim et al., 2017; Furlanello et al., 2018; Yang et al., 2018). The predominant hypothesis in the literature attests this surprising gain in performance to the so-called *dark knowledge* of the teacher, i.e., its logits encode additional information about the data distribution (Hinton et al., 2015; Wang et al., 2021; Xu et al., 2018).

Another area relying on a teacher-student setup is self-supervised learning where the goal is to learn informative representations in the absence of targets (Caron et al., 2021; Grill et al., 2020; Chen & He, 2021; Zbontar et al., 2021; Assran et al., 2022). Here, the two models typically receive two different augmentations of a sample, and the student is forced to mimic the teacher's behavior. Such a learning strategy encourages representations that remain invariant to the employed augmentation pipeline, which in turn leads to better downstream performance.

Despite its importance as a building block, the teacher-student setting itself remains very difficult to analyze as its contribution is often overshadowed by stronger components in the pipeline, such as *dark knowledge* in the trained teacher or the inductive bias of data augmentation. In this work, we take a step towards simplifying and isolating the key components in the setup by devising a very simple experiment; instead of working with a trained teacher, we consider teachers at random initialization, stripping them from any data dependence and thus removing any *dark knowledge*. We also remove augmentations, making the setting completely symmetric between student and teacher and further reducing inductive bias. Counter-intuitively, we observe that even in this setting, the student still manages to learn from its teacher and even exceed it significantly in terms of representational quality, measured through linearly probing the features (see Fig. 1). This result shows the fol-

[1] Department of Computer Science, ETH Zürich, Switzerland. Correspondence to: Felix Sarnthein <safelix@ethz.ch>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

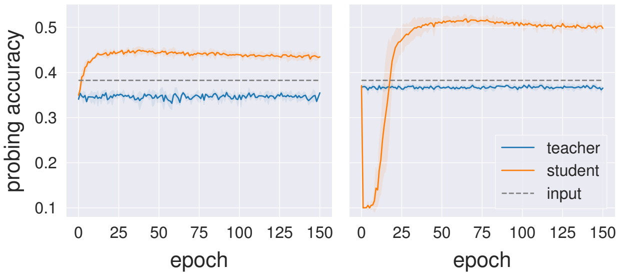

Figure 1. Linear probing accuracies of representations generated by teachers, students, and the flattened input images on *CIFAR10* as a function of training time. **Left:** *ResNet18*. **Right:** *VGG11* without batch normalization.

lowing: (1) Even in the absence of *dark knowledge*, relevant feature learning can happen for the student in the setting of self-distillation. (2) Data augmentation is the main but not only ingredient in non-contrastive self-supervised learning that leads to representation learning.

Surprisingly, we find that initializing the student close to the teacher further amplifies the implicit regularization present in the dynamics. This is in line with common practices in non-contrastive learning, where teacher and student are usually initialized closely together and only separated through small asymmetries in architecture and training protocol (Grill et al., 2020; Caron et al., 2021). We study this locality effect of the landscape and connect it with the *asymmetric valley* phenomenon observed in He et al. (2019).

The improvement in probing accuracy suggests that some information about the data is incorporated into the network's weights. To understand how this information is retained, we compare the behavior of supervised optimization to fine-tuning student networks. We find that some of the learning dynamics observable during the early phase of supervised training also occur during random teacher distillation. In particular, the student already contains sparse subnetworks and reaches the border of linear basins in the supervised loss landscape. This contrasts (Frankle et al., 2020) where training on a concrete learning task for a few epochs is essential. Ultimately, these results suggest that label-independent optimization dynamics exist and allow exploring the supervised loss landscape to a certain degree.

2. Related Work

Several works in the literature aim to analyze self-distillation and its impact on the student. Phuong & Lampert (2019) prove a generalization bound that establishes fast decay of the risk in the case of linear models. Mobahi et al. (2020) demonstrate an increasing regularization effect through repeated distillation for kernel regression. Ji & Zhu (2020) consider a similar approach and rely on the fact that very wide networks behave very similarly to the neural tangent kernel (Jacot et al., 2018) and leverage this connection to establish risk bounds. Allen-Zhu & Li (2020) on the other hand, study more realistic width networks and show that if the data satisfies a certain multi-view property, ensembling and distilling is provably beneficial. Yuan et al. (2020) study a similar setup as our work by considering teachers that are not perfectly pre-trained but of weaker (but still far from random) nature. They show that the *dark knowledge* is more of a regularization effect and that a similar boost in performance can be achieved by label smoothing. Stanton et al. (2021) further question the relevance of *dark knowledge* by showing that students outperform their teacher without fitting the *dark knowledge*. We would like to point out however that we study completely random teachers and our loss function does not provide the hard labels for supervisory signal, making our task completely independent of the targets.

Self-supervised learning can be broadly split into two categories, contrastive and non-contrastive methods. Contrastive methods rely on the notion of negative examples, where features are actively encouraged to be dissimilar if they stem from different examples (Chen et al., 2020; Schroff et al., 2015; van den Oord et al., 2018). Non-contrastive methods follow our setting more closely as only the notion of positive examples is employed (Caron et al., 2021; Grill et al., 2020; Chen & He, 2021). While these methods enjoy great empirical successes, a theoretical understanding is still largely missing. Tian et al. (2021) investigate the collapse phenomenon in non-contrastive learning and show in a simplified setting how the stop gradient operation can prevent it. Wang et al. (2022) extend this work and prove in the linear setting how a data-dependent projection matrix is learned. Zhang et al. (2022) explore a similar approach and prove that *SimSiam* (Chen & He, 2021) avoids collapse through the notion of extra-gradients. Anagnostidis et al. (2022) show that strong representation learning occurs with heavy data augmentations even if random labels are used. Despite this progress on the optimization side, a good understanding of feature learning has largely remained elusive.

The high-dimensional loss landscapes of neural networks remain very mysterious, and their properties play a crucial role in our work. Safran & Shamir (2017) prove that spurious local minima exist in the teacher-student loss of two-layer ReLU networks. Garipov et al. (2018); Draxler et al. (2018) show that two SGD solutions are always connected through a non-linear valley of low loss. Frankle & Carbin (2018); Frankle et al. (2019; 2020) investigate the capacity of overparameterized networks through pruning of weights. They find that sparse sub-networks develop already very early in neural network training. Zaidi et al. (2022); Benzing et al. (2022) investigate random initializations in supervised loss landscapes. Still, the field lacks a convincing explanation as to how simple first-order gradient-based methods such as SGD manage to navigate the landscape so efficiently.

3. Setting

Notation. Let us set up some notation first. We consider a family of parametrized functions $\mathcal{F} = \{f_{\boldsymbol{\theta}} : \mathbb{R}^d \to \mathbb{R}^m \big| \boldsymbol{\theta} \in \Theta\}$ where $\boldsymbol{\theta}$ denotes the (vectorized) parameters of a given model and Θ refers to the underlying parameter space. In this work, we study the teacher-student setting, i.e., we consider two models $f_{\boldsymbol{\theta}_T}$ and $f_{\boldsymbol{\theta}_S}$ from the same function space $\mathcal{F}$. We will refer to $f_{\boldsymbol{\theta}_T}$ as the teacher model and to $f_{\boldsymbol{\theta}_S}$ as the student model. Notice that here we assume that both teacher and student have the same architecture unless otherwise stated. Moreover, assume that we have access to $n \in \mathbb{N}$ input-output pairs $(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n) \overset{i.i.d.}{\sim} \mathcal{D}$ distributed according to some probability measure $\mathcal{D}$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{0, \dots, K-1\}$ encodes the class membership for one of the $K \in \mathbb{N}$ classes.

Supervised. The standard learning paradigm in machine learning is supervised learning, where a model $f_{\boldsymbol{\theta}} \in \mathcal{F}$ is chosen based on empirical risk minimization, i.e., given a loss function l, we train a model to minimize

$$L(\boldsymbol{\theta}) := \sum_{i=1}^{n} l(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i).$$

Minimization of the objective is usually achieved by virtue of standard first-order gradient-based methods such as SGD or ADAM (Kingma & Ba, 2014), where parameters $\boldsymbol{\theta} \sim \text{INIT}$ are randomly initialized and then subsequently updated based on gradient information.

Teacher-Student Loss. A similar but distinct way to perform learning is the teacher-student setting. Here we first fix a teacher model $f_{\boldsymbol{\theta}_T}$ where $\boldsymbol{\theta}_T$ is usually a parameter configuration arising from training in a supervised fashion on the same task. The task of the student $f_{\boldsymbol{\theta}_S}$ is then to mimic the teacher's behavior on the training set by minimizing a distance function d between the two predictions,

$$L(\boldsymbol{\theta}_S) := \sum_{i=1}^{n} d\left(f_{\boldsymbol{\theta}_S}(\boldsymbol{x}_i), f_{\boldsymbol{\theta}_T}(\boldsymbol{x}_i)\right). \tag{1}$$

We have summarized the setting schematically in Fig. 2. We experiment with several choices for the distance function but largely focus on the KL divergence. We remark that the standard definition of distillation (Hinton et al., 2015) consider a combination of losses of the form

$$L(\boldsymbol{\theta}_S) := \sum_{i=1}^{n} d\left(f_{\boldsymbol{\theta}_S}(\boldsymbol{x}_i), f_{\boldsymbol{\theta}_T}(\boldsymbol{x}_i)\right) + \beta \sum_{i=1}^{n} l(f_{\boldsymbol{\theta}_S}(\boldsymbol{x}_i), y_i),$$

for $\beta > 0$, thus the objective is also informed by the true labels y. Here we set $\beta = 0$ to precisely test how much performance is solely due to the implicit regularization present in the learning dynamics and the inductive bias of the model.

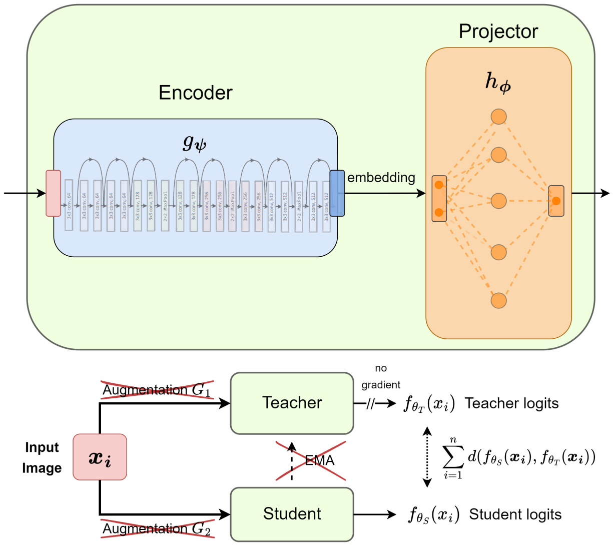
Projector
h_{ϕ}
Encoder
g_{ψ}
embedding
no gradient
~~Augmentation G_1~~
Teacher
$f_{\theta_T}(x_i)$ Teacher logits
Input Image
$\boldsymbol{x_i}$
~~EMA~~
$\sum_{i=1}^{n} d(f_{\theta_S}(\boldsymbol{x_i}), f_{\theta_T}(\boldsymbol{x_i}))$
Student
$f_{\theta_S}(x_i)$ Student logits
~~Augmentation G_2~~

Figure 2. Schematic drawing of the teacher-student setup. The model consists of an encoder and projector. The same image is passed to both student and teacher, and the outputs of the projectors are compared. The student weights are then adjusted to mimic the output of the teacher. In this work, we consider a simplified setting without augmentations and without teacher updates such as EMA.

Somewhat counter-intuitively, it has been observed in many empirical works that the resulting student often outperforms its teacher. It has been hypothesized in many prior works that the teacher logits $f_{\boldsymbol{\theta}_T}(\boldsymbol{x})$ encode some additional, relevant information for the task that benefits learning (*dark knowledge*), i.e., wrong but similar classes might have a non-zero probability under the teacher model (Hinton et al., 2015; Wang et al., 2021; Xu et al., 2018). In the following, we will explore this hypothesis by systematically destroying the label information in the teacher.

Non-Contrastive. Self-supervised learning is a recently developed methodology enabling the pretraining of vision models on large-scale unlabelled image corpora, akin to the autoregressive loss in natural language processing (Devlin et al., 2019). A subset of these approaches is formed by non-contrastive methods. Consider a set of image augmentations $\mathcal{G}$ where any $G \in \mathcal{G}$ is a composition of standard augmentation techniques such as random crop, random flip, color jittering, etc. The goal of non-contrastive learning is to learn a parameter configuration that is invariant to the employed data augmentations while avoiding simply collapsing to a constant function. Most non-contrastive objectives can be summarized to be of the form

$$L(\boldsymbol{\theta}_S) := \sum_{i=1}^{n} \mathbb{E}_{G_1, G_2}\left[d\left(f_{\boldsymbol{\theta}_S}(G_1(\boldsymbol{x}_i)), f_{\boldsymbol{\theta}_T}(G_2(\boldsymbol{x}_i))\right)\right],$$

where the expectation is taken uniformly over the set of augmentations $\mathcal{G}$. We summarize this pipeline schematically in Fig. 2. While the teacher does not directly receive any gradient information, the parameters $\boldsymbol{\theta}_T$ are often updated based on an exponentially weighted moving average,

$$\boldsymbol{\theta}_T \longleftarrow (1-\gamma)\boldsymbol{\theta}_T + \gamma\boldsymbol{\theta}_S$$

which is applied periodically at a fixed frequency. In this work, we will consider a simplified setting without augmentations and where the teacher remains frozen at random initialization, $\gamma = 0$.

Probing. Since minimizing the teacher-student loss is a form of unsupervised learning if the teacher itself has not seen any labels, we need a way to measure the quality of the resulting features. Here we rely on the idea of probing representations, a very common technique from self-supervised learning (Chen & He, 2021; Chen et al., 2020; Caron et al., 2021; Bardes et al., 2021; Grill et al., 2020). As illustrated in Fig. 2, the network is essentially split into an encoder $g_{\boldsymbol{\psi}} : \mathbb{R}^d \to \mathbb{R}^r$ and a projector $h_{\boldsymbol{\phi}} : \mathbb{R}^r \to \mathbb{R}^m$ where it holds that $f_{\boldsymbol{\theta}} = h_{\boldsymbol{\phi}} \circ g_{\boldsymbol{\psi}}$. The encoder is usually given by the backbone of a large vision model such as *ResNet* (He et al., 2016) or *VGG* (Simonyan & Zisserman, 2014), while the projector is parametrized by a shallow MLP. We then *probe* the representations $g_{\boldsymbol{\psi}}$ by learning a linear layer on top, where we now leverage the label information $y_1, \dots, y_n$. Notice that the weights of the encoder remain frozen while learning the linear layer. The idea is that a linear model does not add more feature learning capacity, and the resulting probing accuracy hence provides an adequate measure of the quality of the representations. Unless otherwise stated, we perform probing on the *CIFAR10* dataset (Krizhevsky & Hinton, 2009) and aggregate mean and standard deviation over three runs.

4. Random Teacher Distillation

Distillation. Let us denote by $\boldsymbol{\theta} \sim$ *INIT* a randomly initialized parameter configuration, according to some standard initialization scheme INIT. Throughout this text, we rely on *Kaiming* initialization (He et al., 2015). In standard self-distillation, the teacher is a parameter configuration $\boldsymbol{\theta}_T^{(l)}$ resulting from training in a supervised fashion for $l \in \mathbb{N}$ epochs on the task $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$.

In a next step, the teacher is then distilled into a student, i.e., the student is trained to match the outputs of the pre-trained teacher $f_{\boldsymbol{\theta}_T^{(l)}}$. In this work, we change the nature of the teacher and instead consider a teacher at random initialization $\boldsymbol{\theta}_T \sim$ INIT (we drop the superscript 0 for convenience). The teacher has thus not seen any data at all and is hence of a similar (bad) quality as the student. This experiment, therefore, serves as the ideal test bed to measure the implicit regularization present in the optimization itself without relying on any *dark knowledge* about the target distribution. Due to the absence of targets, the setup also closely resembles the learning setting of non-contrastive methods. Through that lens, our experiment can also be interpreted as a non-contrastive pipeline without *augmentations* and exponential moving average.

We minimize the objective (1) with the ADAM optimizer (Kingma & Ba, 2014) using a learning rate $\eta = 0.001$. We analyze two encoder types based on the popular *ResNet18* and *VGG11* architectures, and similarly to Caron et al. (2021), we use a 2-hidden layer MLP with an L_2 bottleneck, as a projector. To assess whether batch-dependent statistics play a role, we remove the batch normalization layers (Ioffe & Szegedy, 2015) from the *VGG11* architecture. For more details on the architecture and hyperparameters, we refer to App. E.

DATASET	MODEL	TEACHER	STUDENT	INPUT
CIFAR10	*ResNet18*	35.50	**46.02**	39.02
	VGG11	36.55	**51.98**	
CIFAR100	*ResNet18*	11.58	**21.50**	14.07
	VGG11	12.05	**26.62**	
STL10	*ResNet18*	24.24	**40.58**	31.51
	VGG11	24.67	**46.20**	
TinyImageNet	*ResNet18*	4.85	**10.40**	3.28
	VGG11	5.25	**12.88**	

Table 1. Linear probing accuracies (in percentage) of the representations for various datasets for teacher, student and flattened input images. Students outperform the baselines in all cases.

We display the linear probing accuracy of both student and teacher as a function of training time in Fig. 1. We follow the protocol of non-contrastive learning and initialize the student closely to the teacher. We will expand more on this choice of initialization in the next paragraph. Note that while the teacher remains fixed throughout training, accuracies can vary due to stochastic optimization of linear probing. The dashed line represents the linear probing accuracy obtained directly from the (flattened) inputs. We clearly see that the student significantly outperforms its teacher throughout the training. Moreover, it also improves over probing on the raw inputs, demonstrating that not simply less signal is lost due to random initialization but rather that meaningful learning is performed. We expand our experimental setup to more datasets, including *CIFAR100* (Krizhevsky & Hinton, 2009), *STL10* (Coates et al., 2011) and *TinyImageNet* (Le & Yang, 2015). We summarize the results in Table 1. We observe that across all tasks, distilling a random teacher into its student proves beneficial in terms of probing accuracy. For further ablations on the projection head, we refer to the App. B. Moreover, we find similar results for more architectures and k-NN instead of linear probing in App. C.

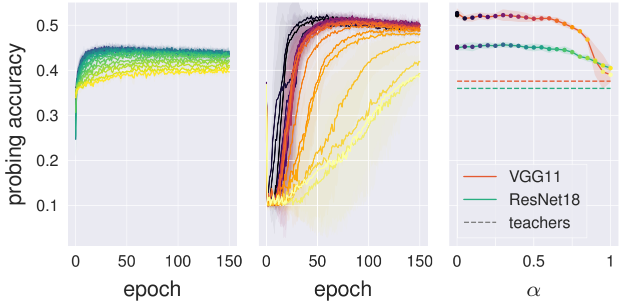

Figure 3. Linear probing accuracies as a function of the locality parameter α on *CIFAR10*. The color gradient (bright → dark) reflects the value of α ($0 \to 1$) for *ResNet18* in green and *VGG11* in red tones. **Left:** *ResNet18*. **Middle:** *VGG11*. **Right:** Summary.

Local Initialization. It turns out that the initialization of the student and its proximity to the teacher plays a crucial role. To that end, we consider initializations of the form

$$\boldsymbol{\theta}_S(\alpha) = \frac{1}{\delta}\left((1-\alpha)\boldsymbol{\theta}_T + \alpha\tilde{\boldsymbol{\theta}}\right),$$

where $\tilde{\boldsymbol{\theta}} \sim$ INIT is a fresh initialization, $\alpha \in [0,1]$ and $\delta = \sqrt{\alpha^2 + (1-\alpha)^2}$ ensures that the variance remains constant $\forall\alpha \in [0,1]$. By increasing α from 0 towards 1, we can gradually separate the student initialization from the teacher and ultimately reach the more classical setup of self-distillation where the student is initialized independently from the teacher. Note, that in the non-contrastive learning setting, teacher and student are initialized at the same parameter values (i.e., $\alpha = 0$), and only minor asymmetries in the architectures lead to different overall functions.

We now study how the locality parameter α affects the resulting quality of the representations of the student in our setup. In Fig. 3, we display the probing accuracy as a function of the training epoch for different choices of $alpha$. Furthermore, we summarize the resulting accuracy of the student as a function of the locality parameter α. Surprisingly, we observe that random teacher distillation behaves very similarly for all $\alpha \in [0, 0.6]$. Increasing α more slows down the convergence and leads to worse overall probing performance. However, even initializing the student independently of the teacher ($\alpha = 1$) results in a considerable improvement over the teacher. In other words, we show that representation learning can occur in self-distillation for any random teacher without *dark knowledge*. To the best of our knowledge, we are the first to observe such a locality phenomenon in the teacher-student landscape. We investigate this phenomenon in more detail in the next section and, for now, if not explicitly stated otherwise, use initializations with small locality parameter $\alpha \sim 10^{-10}$. Safran & Shamir (2017) prove that spurious local minima exist in the teacher-student loss of two-layer ReLU networks. We speculate that this might be the reason why initializing students close to the teacher is beneficial, and provide evidence in App. D

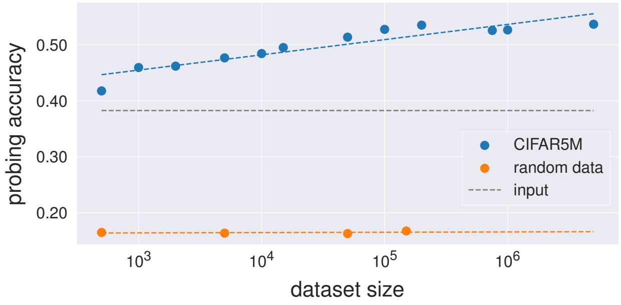

Figure 4. Linear probing accuracies of a *VGG11* trained on *CIFAR5M* or Gaussian noise inputs and evaluated on *CIFAR10* as a function of sample size n. Representations are data dependent.

Data-Dependence. In a next step, we aim to understand better to which degree the learned features are data dependent, i.e., tuned to the particular input distribution $\boldsymbol{x} \sim p_{\boldsymbol{x}}$. While the improvement over the raw input probe already suggests non-trivial learning, we want to characterize the role of the input data more precisely.

As a first experiment, we study how the improvement of the student over the teacher evolves as a function of the sample size n involved in the teacher-student training phase. We use the *CIFAR5M* dataset, where the standard *CIFAR10* dataset has been extended to 5 million data points using a generative adversarial network (Nakkiran et al., 2021). We train the student for different sample sizes in the interval $[5 \times 10^2, 5 \times 10^6]$ and probe the learned features on the standard *CIFAR10* training and test set. We display the resulting probing accuracy as a function of sample size in Fig. 4 (blue line). Indeed, we observe a steady increase in the performance of the student as the size of the data corpus grows, highlighting that data-dependent feature learning is happening.

As further confirmation, we replace the inputs $\boldsymbol{x}_i \sim p_{\boldsymbol{x}}$ with pure Gaussian noise, i.e. $\boldsymbol{x}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbb{1})$, effectively removing any relevant structure in the samples. The linear probing, on the other hand, is again performed on the clean data. This way, we can assess whether the teacher-student training is simply moving the initialization in a favorable way (e.g. potentially uncollapsing it), which would still prove beneficial for meaningful tasks. We display the probing accuracy for these random inputs in Fig. 4 as well (orange line) and observe that such random input training does not lead to an improvement of the student across all dataset sizes. This is another indication that data-dependent feature learning is happening, where in this case, adapting to the noise inputs of course proves detrimental for the clean probing.

Transferability. As a final measure for the quality of the learned features, we test how well a set of representations obtained on one task transfers to a related but dif-

DATASET	MODEL	TEACHER	STUDENT
CIFAR10	*ResNet18*	35.50	**46.06**
	VGG11	36.55	**52.45**
CIFAR100	*ResNet18*	11.58	**22.60**
	VGG11	12.05	**27.49**
STL10	*ResNet18*	24.24	**41.42**
	VGG11	24.67	**45.86**

Table 2. Linear probing accuracies of the representations for various datasets for teacher and student. Students distilled from random teachers on *TinyImageNet* generalize out of distribution.

ferent task. More precisely, we are given a source task $\mathcal{A} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n} \overset{i.i.d.}{\sim} \mathcal{D}_{\mathcal{A}}$ and a target task $\mathcal{B} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{\tilde{n}} \overset{i.i.d.}{\sim} \mathcal{D}_{\mathcal{B}}$ and assume that both tasks are related, i.e., some useful features on $\mathcal{A}$ also prove to be useful on task $\mathcal{B}$. We first use the source task $\mathcal{A}$ to perform random teacher distillation and then use the target task $\mathcal{B}$ to train and evaluate the linear probe. Clearly, we should only see an improvement in the probing accuracy over the (random) teacher if the features learned on the source task encode relevant information for the target task as well. We use *TinyImageNet* as the source task and evaluate on *CIFAR10*, *CIFAR100*, and *STL10* as target tasks for our experiments. We illustrate the results in Table 2 and observe that transfer learning occurs. This suggests that the features learned by random teacher distillations can encode common properties of natural images which are shared across tasks.

5. Loss and Probing Landscapes

Visualization. We now revisit the locality property identified in the previous section, where initializations with α closer to zero outperformed other configurations. To gain further insight into the inner workings of this phenomenon, we visualize the teacher-student loss landscape as well as the resulting probing accuracies as a function of the model parameters. Since the loss function is a very high-dimensional function of the parameters, only slices of it can be visualized at once. More precisely, given two directions $\boldsymbol{v}_1, \boldsymbol{v}_2$ in parameter space, we form a visualization plane of the form

$$\boldsymbol{\theta}(\lambda_1, \lambda_2) = \lambda_1 \boldsymbol{v}_1 + \lambda_2 \boldsymbol{v}_2, \qquad (\lambda_1, \lambda_2) \in [0, 1]^2$$

and then collect loss and probing values at a certain resolution. Such visualization strategy is very standard in the literature, see e.g., Li et al. (2018); Garipov et al. (2018); Izmailov et al. (2021). Denote by $\boldsymbol{\theta}_S^*(\alpha)$ the student trained until convergence initialized with locality parameter α. We study two choices for the landscape slices. First, we refer to a *non-local view* as the plane defined by the random teacher $\boldsymbol{\theta}_T$, the student at a fresh initialization $\boldsymbol{\theta}_S(1)$ and the resulting trained student $\boldsymbol{\theta}_S^*(1)$, i.e., we set $\boldsymbol{v}_1 = \boldsymbol{\theta}_S(1) - \boldsymbol{\theta}_T$ and $\boldsymbol{v}_2 = \boldsymbol{\theta}_S^*(1) - \boldsymbol{\theta}_T$. As a second choice, we refer to a *shared view* as the plane defined by the random teacher $\boldsymbol{\theta}_T$, the trained student starting from a fresh initialization $\boldsymbol{\theta}_S^*(1)$ and the trained student $\boldsymbol{\theta}_S^*(0)$ initialized closely to the teacher, i.e., we set $\boldsymbol{v}_1 = \boldsymbol{\theta}_S^*(0) - \boldsymbol{\theta}_T$ and $\boldsymbol{v}_2 = \boldsymbol{\theta}_S^*(1) - \boldsymbol{\theta}_T$. Note that α is not exactly zero but around 10^{-10}.

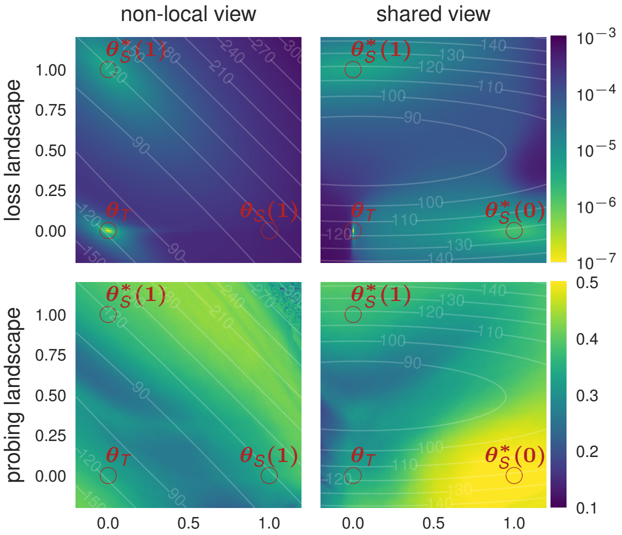

Figure 5. Visualization of the loss and probing landscape. The left column corresponds to the *non-local view* with $\alpha = 1$. The right column depicts the *shared view*, containing both the local ($\alpha = 0$) and the non-local solution ($\alpha = 1$). The first row displays the loss landscape and the second one shows the probing landscape. Contours lines represent $||\boldsymbol{\theta}||_2$, orthogonal projections are in App. C.3.

We show the results in Fig. 5, where the left and right columns represent the *non-local* and the *shared view* respectively, while the first and the second row display loss and probing landscapes respectively. Let us focus on the *non-local view* first. Clearly, for $\alpha = 1$ the converged student $\boldsymbol{\theta}_S^*(1)$ ends up in a qualitatively different minimum than the teacher, i.e., the two points are separated by a significant loss barrier. This is expected as the student is initialized far away from the teacher. Further, we see that the probing landscape is largely unaffected by moving from the initialization $\boldsymbol{\theta}_S(0)$ to the solution $\boldsymbol{\theta}_S^*(0)$, confirming our empirical observation in Fig. 3 that far way initialized students only improve slightly. The *shared view* reveals more structure. We see that although it was initialized very closely to the teacher, the student $\boldsymbol{\theta}_S^*(0)$ moved considerably. While the loss barrier is lower as in the case of $\boldsymbol{\theta}_S^*(1)$, it is still very apparent that $\boldsymbol{\theta}_S^*(0)$ settled for a different, local minimum that coincides with a region of high probing accuracy. This is surprising as the teacher itself is the global loss minimum. For more visualizations, including the loss landscape for the encoder, we refer to App. C.3.

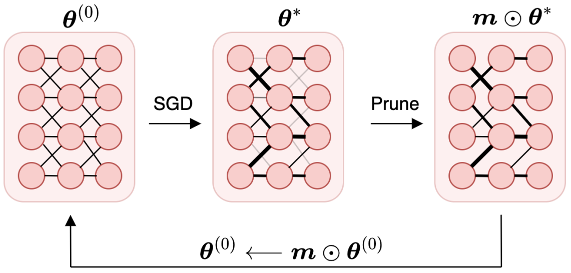
$\boldsymbol{\theta}^{(0)}$
SGD
$\boldsymbol{\theta}^*$
Prune
$\boldsymbol{m} \odot \boldsymbol{\theta}^*$
$\boldsymbol{\theta}^{(0)} \longleftarrow \boldsymbol{m} \odot \boldsymbol{\theta}^{(0)}$

Figure 6. Illustration of the lottery ticket hypothesis and iterative magnitude-based pruning.

Asymmetric valleys. A striking structure in the loss landscape of the *shared view* is the very pronounced asymmetric valley around the teacher $\boldsymbol{\theta}_T$. While there is a very steep increase in loss towards the left of the view (dark blue), the loss increases only gradually in the opposite direction (light turquoise) and quickly decreases into the local minimum of the converged student $\boldsymbol{\theta}_S^*(0)$. Surprisingly, this direction orthogonal to the cliff identifies a region of high accuracy in the probing landscape. A fact remarkably in line with this situation is proven by He et al. (2019). They show that being on the flatter side of an asymmetric valley (i.e., towards $\boldsymbol{\theta}_S^*(0)$) provably leads to better generalization compared to lying in the valley itself (i.e., $\boldsymbol{\theta}_T$). Initializing the student closely to the teacher seems to capitalize on that fact and leads to systematically better generalization. Still, it remains unclear why such an asymmetric valley is only encountered close to the teacher and not for initializations with $\alpha = 1$. We leave a more in-depth analysis of this phenomenon for future work.

6. Connection to Supervised Optimization

Lottery Tickets. A way to assess the structure present in neural networks is through sparse network discovery, i.e., the *lottery ticket hypothesis*. The lottery ticket hypothesis by Frankle & Carbin (2018) posits the following: Any large network possesses a sparse subnetwork that can be trained as fast and which achieves or surpasses the test error of the original network. They prove this using the power of hindsight and discover such sparse networks through the following iterative pruning strategy:

1. Fix an initialization $\boldsymbol{\theta}^{(0)} \sim$ INIT and train a network to convergence in a supervised fashion, leading to $\boldsymbol{\theta}^*$.
2. Prune the parameters based on some criterion, leading to a binary mask $\boldsymbol{m}$ and pruned parameters $\boldsymbol{m} \odot \boldsymbol{\theta}^*$.
3. Prune the initialized network $\boldsymbol{m} \odot \boldsymbol{\theta}^{(0)}$ and re-train it.

The above procedure is repeated for a fixed number of times r, and in every iteration, a fraction $k \in [0, 1]$ of the weights is pruned, leading to an overall pruning rate of $p_r = \sum_{i=0}^{r-1}(1-k)^i \times k$ percentage of weights. We illustrate the algorithm in Fig. 6. The choice of pruning technique is flexible, but in the common variant *iterative magnitude pruning (IMP)*, the globally smallest weights are pruned. The above recipe turns out to work very well for MLPs and smaller convolutional networks, and indeed very sparse solutions can be discovered without any deterioration in terms of training time or test accuracy (Frankle & Carbin, 2018). However, for more realistic architectures such as *ResNets*, the picture changes and subnetworks can only be identified if the employed learning rate is low enough. Surprisingly, Frankle et al. (2019) find that subnetworks in such architectures develop very early in training and thus add the following modification to the above strategy: Instead of rewinding back to the initialization $\boldsymbol{\theta}^{(0)}$ and applying the pruning there, another checkpoint $\boldsymbol{\theta}^{(l)}$ early in training is used and $\boldsymbol{m} \odot \boldsymbol{\theta}^{(l)}$ is re-trained instead of $\boldsymbol{m} \odot \boldsymbol{\theta}^{(0)}$.

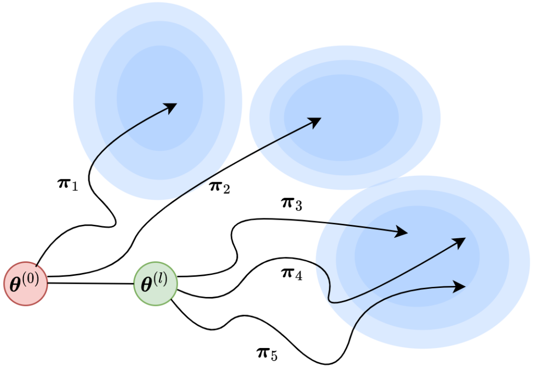

Figure 7. Illustration of stability of SGD and linear mode-connectivity. Blue contour lines indicate a basin of low test loss, $\boldsymbol{\pi}_i$ denote different batch orderings for SGD.

Frankle et al. (2019) demonstrate that checkpoints as early as 1 epoch can suffice to identify lottery tickets, even at standard learning rates. Interestingly, Frankle et al. (2019) further show that the point in time l where lottery tickets can be found coincides with the time where SGD becomes stable to different batch orderings $\boldsymbol{\pi}$, i.e., different runs of SGD with distinct batch orderings but the same initialization $\boldsymbol{\theta}^{(l)}$ end up in the same linear basin. This property is also called linear mode connectivity; we provide an illustration in Fig. 7. Notice that in general, linear mode-connectivity does not hold, i.e., two SGD runs from the same initialization end up in two disconnected basins (Frankle et al., 2019; Garipov et al., 2018).

IMP from the Student. A natural question that emerges now is whether rewinding to a student checkpoint $\boldsymbol{\theta}_S^*$, obtained through random teacher distillation, already developed sparse structures in the form of lottery tickets. We com-

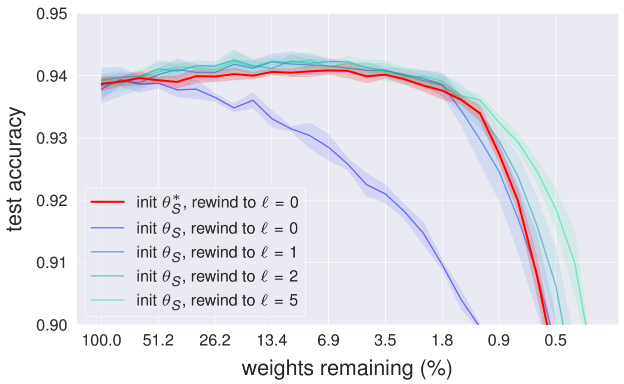

Figure 8. Test accuracy as a function of sparsity for different initialization and rewinding strategies. Fresh initializations $\boldsymbol{\theta}_S$ are not robust to IMP with rewinding to initialization ($l = 0$), this only emerges with rewinding to $l \geq 1$. Student checkpoints $\boldsymbol{\theta}_S^*$ are always robust to IMP even with rewinding to $l = 0$. One epoch corresponds to 196 steps. Aggregation is done over 5 checkpoints.

pare the robustness of our student checkpoints $\boldsymbol{\theta}_S^*$ with random initialization at different rewinding points $\boldsymbol{\theta}^{(l)}$, closely following the setup in Frankle et al. (2019). We display the results in Fig. 8, where we plot test performance on *CIFAR10* as a function of the sparsity level. We use a *ResNet18* and iterative magnitude pruning, reducing the network by a fraction of 0.2 every round. We compare against rewinding to supervised checkpoints $\boldsymbol{\theta}^{(l)}$ for $l \in \{0, 1, 2, 5\}$ where l is measured in number of epochs.

We observe that rewinding to random initialization ($l = 0$), as shown in Frankle & Carbin (2018); Frankle et al. (2019), incurs strong losses in terms of test accuracy at all pruning levels and thus $\boldsymbol{\theta}_S$ does not constitute a lottery ticket. The distilled student $\boldsymbol{\theta}_S^*$, on the other hand, contains a lottery ticket, as it remains very robust to strong degrees of pruning. In fact, $\boldsymbol{\theta}_S^*$ shows similar behavior to the networks rewound to epoch 1 and 2 in supervised training. This suggests that random teacher distillation imitates some of the learning dynamics in the first epochs of supervised optimization. We stress here that no label information was required for sparse subnetworks to develop. This aligns with results in (Frankle et al., 2020), showing that auxiliary tasks such as rotation prediction can lead to lottery tickets. However, this is no surprise, as Anagnostidis et al. (2022) show that the data-informed bias of augmentations can already lead to strong forms of learning. We believe our result is more powerful since random teacher distillation relies solely on implicit regularization in SGD and does not require a task at all.

Linear Mode Connectivity. In light of the observation regarding the stability of SGD in Frankle et al. (2019), we verify whether a similar stability property holds for the student checkpoint $\boldsymbol{\theta}_S^*$. To that end, we train several runs of SGD in a supervised fashion with initialization $\boldsymbol{\theta}_S^*$ on different batch orderings $\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_b$ and study the test accuracies occurring along linear paths between different solutions $\boldsymbol{\theta}_{\boldsymbol{\pi}_i}^*$ for $i = 1, \dots, b$, i.e.

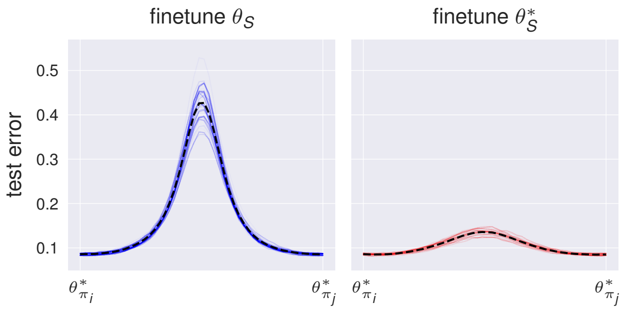

Figure 9. Test error when interpolating between networks that were trained from the same initialization. **Left:** Networks initialized at the teacher location, i.e., random initialization. **Right:** Networks initialized at the converged student $\boldsymbol{\theta}_S^*(0)$. Aggregation is done over 3 initializations and 5 different data orderings $\boldsymbol{\pi}_i$.

$$\boldsymbol{\theta}_{\boldsymbol{\pi}_i \rightarrow \boldsymbol{\pi}_j}(\gamma) := \gamma \boldsymbol{\theta}_{\boldsymbol{\pi}_i}^* + (1 - \gamma) \boldsymbol{\theta}_{\boldsymbol{\pi}_j}^* .$$

If the test accuracy along the path does not significantly worsen, we call $\boldsymbol{\theta}_{\boldsymbol{\pi}_i}^*$ and $\boldsymbol{\theta}_{\boldsymbol{\pi}_j}^*$ *linearly mode-connected.* We contrast the results with the interpolation curves for SGD runs started from the original, random initialization $\boldsymbol{\theta}_S$. We display the interpolation curves in Fig. 9, where we used three *ResNet18* student checkpoints and finetuned each in five SGD runs with different seeds on *CIFAR10*. We observe that, indeed, the resulting parameters $\boldsymbol{\theta}_{\boldsymbol{\pi}_i}^*$ all lie in approximately the same linear basin. However, the networks trained from the random initialization face a significantly larger barrier. This confirms that random teacher distillation converges towards parameterizations $\boldsymbol{\theta}_S^*$, which are different from those at initialization $\boldsymbol{\theta}_S$. In particular, such $\boldsymbol{\theta}_S^*$ would only appear later in supervised optimization when SGD is already more stable to noise. Ultimately, it shows that random teacher distillation obeys similar dynamics as supervised optimization and can navigate toward linear basins of the supervised loss landscape.

7. Discussion and Conclusion

In this work, we examined the teacher-student setting to disentangle its implicit regularization from other very common components such as *dark knowledge* in trained teachers or data augmentations in self-supervised learning. Surprisingly, students learned strong structures even from random teachers in the absence of data augmentation. We studied the quality of the students and observed that (1) probing accuracies significantly improve over the teacher, (2) features are data-dependent and transferable across tasks, and (3) student checkpoints develop sparse subnetworks at the border of linear basins without training on a supervised task.

The success of teacher-student frameworks such as knowledge distillation and non-contrastive learning can thus at least partially be attributed to the regularizing nature of the learning dynamics. These label-independent dynamics allow the student to mimic the early phase of supervised training by navigating the supervised loss landscape without label information. The simple and minimal nature of our setting makes it an ideal test bed for better understanding this early phase of learning. We hope that future theoretical work can build upon our simplified framework.

Acknowledgements

We thank Sidak Pal Singh for his valuable insights and interesting discussions on various aspects of the topic.

References

Allen-Zhu, Z. and Li, Y. Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning. In *11th International Conference on Learning Representations (ICLR)*, 2 2020. URL https://arxiv.org/abs/2012.09816.

Anagnostidis, S., Bachmann, G., Noci, L., and Hofmann, T. The Curious Case of Benign Memorization. In *11th International Conference on Learning Representations (ICLR)*, 2022. doi: 10.48550/arxiv.2210.14019. URL https://arxiv.org/abs/2210.14019.

Assran, M., Caron, M., Misra, I., Bojanowski, P., Bordes, F., Vincent, P., Joulin, A., Rabbat, M., and Ballas, N. Masked Siamese Networks for Label-Efficient Learning. In *European Conference on Computer Vision (ECCV)*, 2022. doi: 10.48550/arxiv.2204.07141. URL https://arxiv.org/abs/2204.07141.

Ba, J. and Caruana, R. Do deep nets really need to be deep? In *28th Conference on Neural Information Processing Systems (NeurIPS)*, 2013. URL https://arxiv.org/abs/1312.6184.

Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. In *10th International Conference on Learning Representations (ICLR)*, 2021. doi: 10.48550/arxiv.2105.04906. URL https://arxiv.org/abs/2105.04906.

Benzing, F., Schug, S., Ch, S., Meier, R., Von Oswald, J., Ch, V., Akram, Y., Zucchet, N., Aitchison, L., Steger, A., and Ch, S. E. Random initialisations performing above chance and how to find them. *ArXiv*, 9 2022. URL https://arxiv.org/abs/2209.07509.

Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., and Kolesnikov, A. Knowledge distillation: A good teacher is patient and consistent. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. URL https://arxiv.org/abs/2106.05237.

Bucilă, C., Caruana, R., and Niculescu-Mizil, A. Model compression. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2006. URL https://dl.acm.org/doi/abs/10.1145/1150402.1150464.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging Properties in Self-Supervised Vision Transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. URL https://arxiv.org/abs/2104.14294.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In *37th International Conference on Machine Learning (ICML)*, 2020. ISBN 9781713821120. doi: 10.48550/arxiv.2002.05709. URL https://arxiv.org/abs/2002.05709.

Chen, X. and He, K. Exploring Simple Siamese Representation Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. ISBN 9781665445092. doi: 10.48550/arxiv.2011.10566. URL https://arxiv.org/abs/2011.10566.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2011. URL https://proceedings.mlr.press/v15/coates11a.html.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019. doi: 10.18653/v1/N19-1423. URL https://arxiv.org/abs/1810.04805.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations (ICLR)*, 2020. URL https://arxiv.org/abs/2010.11929.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. A. Essentially No Barriers in Neural Network Energy Landscape. In *35th International Conference on Machine Learning (ICML)*, 2018. ISBN 9781510867963. URL https://arxiv.org/abs/1803.00885.

Frankle, J. and Carbin, M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. In *7th International Conference on Learning Representations (ICLR)*, 2018. doi: 10.48550/arxiv.1803.03635. URL https://arxiv.org/abs/1803.03635.

Frankle, J., Dziugaite, G. K., Roy, D. M., and Carbin, M. Linear Mode Connectivity and the Lottery Ticket Hypothesis. In *37th International Conference on Machine Learning (ICML)*, 2019. URL https://arxiv.org/abs/1912.05671.

Frankle, J., Schwab, D. J., and Morcos, A. S. The Early Phase of Neural Network Training. In *8th International Conference on Learning Representations (ICLR)*, 2020. doi: 10.48550/arxiv.2002.10365. URL https://arxiv.org/abs/2002.10365.

Furlanello, T., Lipton, Z. C., Tschannen, M., Itti, L., and Anandkumar, A. Born Again Neural Networks. *35th International Conference on Machine Learning (ICML)*, 2018. doi: 10.48550/arxiv.1805.04770. URL https://arxiv.org/abs/1805.04770.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D., and Wilson, A. G. Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs. In *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018. doi: 10.48550/arxiv.1802.10026. URL https://arxiv.org/abs/1802.10026.

Grill, J. B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., and Valko, M. Bootstrap your own latent: A new approach to self-supervised Learning. In *34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020. doi: 10.48550/arxiv.2006.07733. URL https://arxiv.org/abs/2006.07733.

He, H., Huang, G., and Yuan, Y. Asymmetric Valleys: Beyond Sharp and Flat Local Minima. In *33rd Conference on Neural Information Processing Systems (NeurIPS)*, 2019. URL https://arxiv.org/abs/1902.00744.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. URL https://arxiv.org/abs/1502.01852.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. URL https://arxiv.org/abs/1512.03385.

Hinton, G., Vinyals, O., and Dean, J. Distilling the Knowledge in a Neural Network. *ArXiv*, 2015. doi: 10.48550/arxiv.1503.02531. URL https://arxiv.org/abs/1503.02531.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning (ICML)*, 2015. URL https://proceedings.mlr.press/v37/ioffe15.html.

Izmailov, P., Vikram, S., Hoffman, M. D., and Wilson, A. G. What Are Bayesian Neural Network Posteriors Really Like? 2021.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018. URL https://arxiv.org/abs/1806.07572.

Ji, G. and Zhu, Z. Knowledge distillation in wide neural networks: Risk bound, data efficiency and imperfect teacher. In *34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020. URL https://arxiv.org/abs/2010.10090.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2014. URL https://arxiv.org/abs/1412.6980.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. Technical report, Stanford University, 2015. URL http://vision.stanford.edu/teaching/cs231n/reports/2015/pdfs/yle_project.pdf.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018. URL https://arxiv.org/abs/1712.09913.

Mobahi, H., Farajtabar, M., and Bartlett, P. L. Self-Distillation Amplifies Regularization in Hilbert Space. In *34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020. doi: 10.48550/arxiv.2002.05715. URL https://arxiv.org/abs/2002.05715.

Nakkiran, P., Neyshabur, B., and Sedghi, H. The deep bootstrap framework: Good online learners are good offline generalizers. In *9th International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=guetrIHLFGI.

Phuong, M. and Lampert, C. Towards understanding knowledge distillation. In *36th International Conference on Machine Learning (ICML)*, 2019. URL https://arxiv.org/abs/2105.13093.

Polino, A., Pascanu, R., and Alistarh, D. Model compression via distillation and quantization. In *6th International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=S1XolQbRW.

Safran, I. and Shamir, O. Spurious Local Minima are Common in Two-Layer ReLU Neural Networks. *35th International Conference on Machine Learning (ICML)*, 2017. URL https://arxiv.org/abs/1712.08968.

Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. doi: 10.1109/CVPR.2015.7298682. URL https://arxiv.org/abs/1503.03832.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations (ICLR)*, 2014. URL https://arxiv.org/abs/1409.1556.

Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A. A., and Wilson, A. G. Does Knowledge Distillation Really Work? In *35th Conference on Neural Information Processing Systems (NeurIPS)*, 2021. ISBN 9781713845393. URL https://arxiv.org/abs/2106.05945.

Tian, Y., Chen, X., and Ganguli, S. Understanding self-supervised Learning Dynamics without Contrastive Pairs. In *38th International Conference on Machine Learning (ICML)*, 2021. doi: 10.48550/arxiv.2102.06810. URL https://arxiv.org/abs/2102.06810.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *ArXiv*, 2018. URL https://arxiv.org/abs/1807.03748.

Wang, X., Chen, X., Du, S. S., and Tian, Y. Towards demystifying representation learning with non-contrastive self-supervision. *ArXiv*, 2022. URL https://arxiv.org/abs/2110.04947.

Wang, Y., Li, H., Chau, L.-p., and Kot, A. C. Embracing the dark knowledge: Domain generalization using regularized knowledge distillation. In *29th ACM International Conference on Multimedia*, 2021. doi: 10.1145/3474085.3475434. URL https://doi.org/10.1145/3474085.3475434.

Xu, K., Park, D. H., Yi, C., and Sutton, C. Interpreting deep classifier by visual distillation of dark knowledge. *ArXiv*, 2018. URL https://arxiv.org/abs/1803.04042.

Yang, C., Xie, L., Su, C., and Yuille, A. L. Snapshot distillation: Teacher-student optimization in one generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. URL https://arxiv.org/abs/1812.00123.

Yim, J., Joo, D., Bae, J., and Kim, J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. URL https://ieeexplore.ieee.org/document/8100237.

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. Revisiting knowledge distillation via label smoothing regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. URL https://arxiv.org/abs/1909.11723.

Zaidi, S., Berariu, T., Kim, H., Bornschein, J., Clopath, C., Teh, Y. W., and Pascanu, R. When Does Re-initialization Work? *Understanding Deep Learning Through Empirical Falsification (NeurIPS Workshop)*, 6 2022. URL https://arxiv.org/abs/2206.10011.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. In *38th International Conference on Machine Learning (ICML)*, 2021. doi: 10.48550/arxiv.2103.03230. URL https://arxiv.org/abs/2103.03230.

Zhang, C., Zhang, K., Zhang, C., Pham, T. X., Yoo, C. D., and Kweon, I. S. How Does SimSiam Avoid Collapse Without Negative Samples? A Unified Understanding with Self-supervised Contrastive Learning. In *10th International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2203.16262.

A. The Algorithm

Distillation from a random teacher has two important details. The outputs are very high-dimensional, 2^{16}-d. And a special component, the *l2-bottleneck*, is hidden in the architecture of the projection head just before the softmax. It linearly maps a feature vector to a low-dimensional space, normalizes it, and computes the dot product with a normalized weight matrix, i.e.

$$x \to \tilde{V}^T \frac{W^T x + b}{||W^T x + b||_2} \quad \text{with } ||\tilde{V}_{:,i}||_2 = 1$$

for $x \in \mathbb{R}^n$, $W \in \mathbb{R}^{n \times k}$, $b \in \mathbb{R}^k$, $\tilde{V} \in \mathbb{R}^{k \times m}$. This architecture is heavily inspired by DINO (Caron et al., 2021). Let us summarize the method in pseudo-code:

```
encoder, head, wn_layer = ResNet(512), MLP(2048,2048,256), Linear(2^16)

student = initialize(encoder, head, wn_layer)
teacher = copy(student) # initialize with same parameters
for x, y in repeat(data, n_epochs):
    # apply weight-normalization
    normalized_weight_t = normalize(teacher.wn_layer.weight)
    normalized_weight_s = normalize(student.wn_layer.weight)

    # prepare target
    x_t = teacher.head(teacher.encoder(x))
    x_t = normalize(x_t)
    x_t = dot(normalized_weight_t, x_t)
    target = softmax(x_t)

    # prepare prediction
    x_s = student.head(student.encoder(x))
    x_s = normalize(x_s)
    x_s = dot(normalized_weight_s, x_s)
    prediction = softmax(x_s)

    # compute loss, backpropagate and update
    loss = sum(target * -log(prediction)) # cross-entropy
    loss.backward()
    optimizer.step(student) # update only student
```

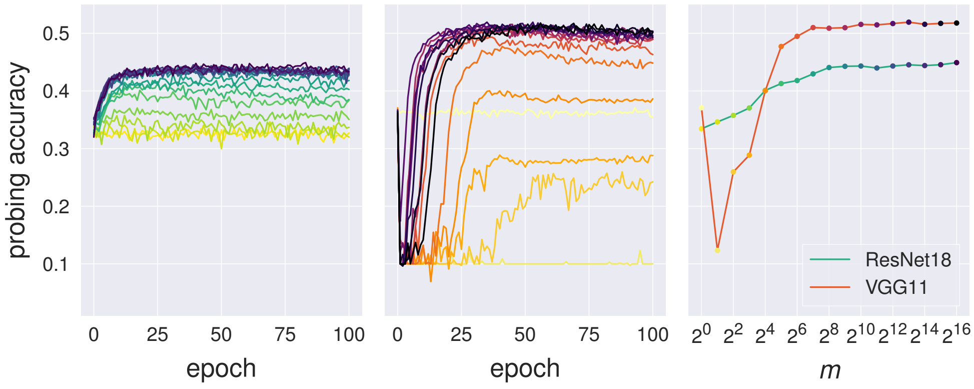

Figure 10. Comparing different output dimensions m of the projection head. Large $m = 2^{16}$ are not crucial for feature learning, but there is phase transition at the bottleneck dimension $m = 2^8 = 256$ Linear probing on *CIFAR10*. **Left:** *ResNet18* (red). **Right:** *VGG11* (green).

B. Ablating the Projector

B.1. Ablating Normalization Layers

If the teacher is used in evaluation mode, then one possible source of asymmetry is introduced by batch normalization layers. But is the effect caused by this batch-dependent signal? Or does the batch dependency amplify the mechanism? In Fig. 11 we compare different types of normalization layers and no normalization (Identity). We observe that although BN stabilizes training, the effect also occurs with batch-independent normalization. Further, networks without normalization reach similar performance but take longer to converge.

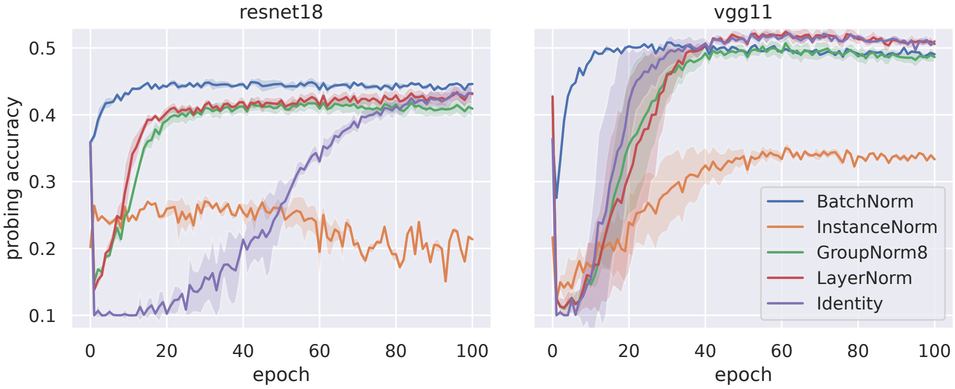

Figure 11. Comparing different types of normalization layers on *CIFAR10*. **Left**: *ResNet18*. **Right**: *VGG11*.

B.2. Ablating the L2-Bottleneck

The *l2-Bottleneck* is a complex layer with many unexplained design choices. We compare different combinations of weight-normalization (wn), linear layer (lin), and feature normalization (fn) for the first and second part of the bottleneck in Figures 12 for a *ResNet18* and a *VGG11* respectively. While the default setup is clearly the most performant, removing feature normalization is more destructive than removing weight normalization. In particular, only one linear layer followed by a feature normalization still exhibits a similar trend and does not break down.

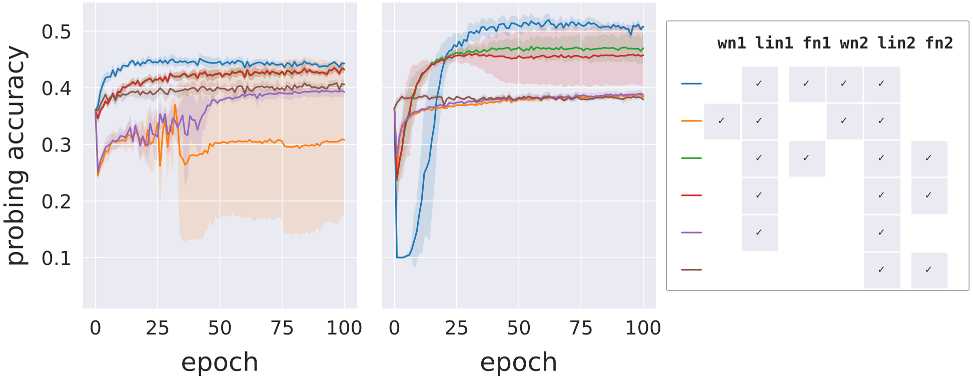

Figure 12. Ablating components of the *l2-bottleneck* on *CIFAR10*. **Left**: *ResNet18*. **Right**: *VGG11*.

C. Additional Results

We present additional experimental results that serve to better understand the regularization properties of self-distillation with random teachers.

C.1. K-NN probing

A different probing choice, instead of learning a linear layer on top of the extracted embeddings, is to perform K-NN classification on the features. We apply K-nearest-neighbour classification with the number of neighbors set to $K = 20$, as commonly done in practice. As in Table 1 in the main text, we present results under K-NN evaluation in Table 3. Also, as in Table 2, we evaluate using K-NN probing the transferability of the learned embeddings from *TinyImageNet* in Table 4.

DATASET	MODEL	TEACHER	STUDENT	INPUT
CIFAR10	*ResNet18*	37.65	**44.67**	33.61
	VGG11	44.92	**51.32**	
CIFAR100	*ResNet18*	13.77	**20.22**	14.87
	VGG11	18.10	**23.53**	
STL10	*ResNet18*	31.71	**37.41**	28.94
	VGG11	36.92	**43.58**	
TinyImageNet	*ResNet18*	4.59	**7.11**	3.44
	VGG11	5.98	**9.23**	

Table 3. K-NN probing accuracies (in percentage) of the representations for various datasets for teacher, student, and raw pixel inputs.

DATASET	MODEL	TEACHER	STUDENT
CIFAR10	*ResNet18*	37.65	**44.45**
	VGG11	44.92	**51.48**
CIFAR100	*ResNet18*	13.77	**19.48**
	VGG11	18.10	**23.95**
STL10	*ResNet18*	31.71	**38.86**
	VGG11	36.92	**42.26**

Table 4. K-NN probing accuracies (in percentage) of the representations for various datasets for teacher and student when transferred from *TinyImageNet*.

C.2. Architectures

For our experiments in the main text, we used the very common *VGG11* and *ResNet18* architectures. Here, we report results for different types of architectures to provide a better picture of the relevance of architectural inductive biases. In particular, we compare with the *Vision Transformer (ViT)* (Dosovitskiy et al., 2020) (patch size 8 for 32×32 images of *CIFAR10*) and find that the effect of representation learning is still present, albeit less pronounced. More generally, we observe that with less inductive bias, the linear probing accuracy diminishes but never breaks down.

MODEL	#PARAMS	TEACHER	STUDENT
NONE (INPUT)	0	39.02	39.02
VGG11	$9'220'480$	36.55	**51.98**
VGG13	$9'404'992$	34.73	**49.26**
VGG16	$14'714'688$	33.08	**46.35**
VGG19	$20'024'384$	30.84	**43.90**
*ResNet20**	$271'824$	28.68	**36.62**
*ResNet56**	$855'120$	14.05	**27.92**
ResNet18	$11'168'832$	35.50	**46.02**
ResNet34	$21'276'992$	28.18	**41.04**
ResNet50	$23'500'352$	19.69	**27.53**
ViT-Tiny	$594'048$	32.93	**35.76**
ViT-Small	$2'072'832$	38.57	**41.68**
ViT-Medium	$3'550'208$	41.09	**43.13**
ViT-Base	$7'684'608$	41.71	**44.38**

Table 5. Linear probing accuracies (in percentage) of the representations for various architectures for teacher, student, and flattened inputs on *CIFAR10*. *ResNet20** and *ResNet56** are the smaller CIFAR-variants from He et al. (2016). The students outperform their teachers in all cases.

C.3. Loss landscapes

The parameter plane visualized in Fig. 5 is defined by interpolation between three parameterizations, thus, distances and angles are not preserved. In the following Fig. 13, we orthogonalize the basis of the parameter plane to achieve a distance and angle-preserving visualization. We note that both converged solutions of the students $\boldsymbol{\theta}_S^*(0)$ and $\boldsymbol{\theta}_S^*(1)$ stay comparably close to their initializations. Further, we provide a zoomed crop of the asymmetric valley around the teacher $\boldsymbol{\theta}_{S_T}$ in Fig. 14.

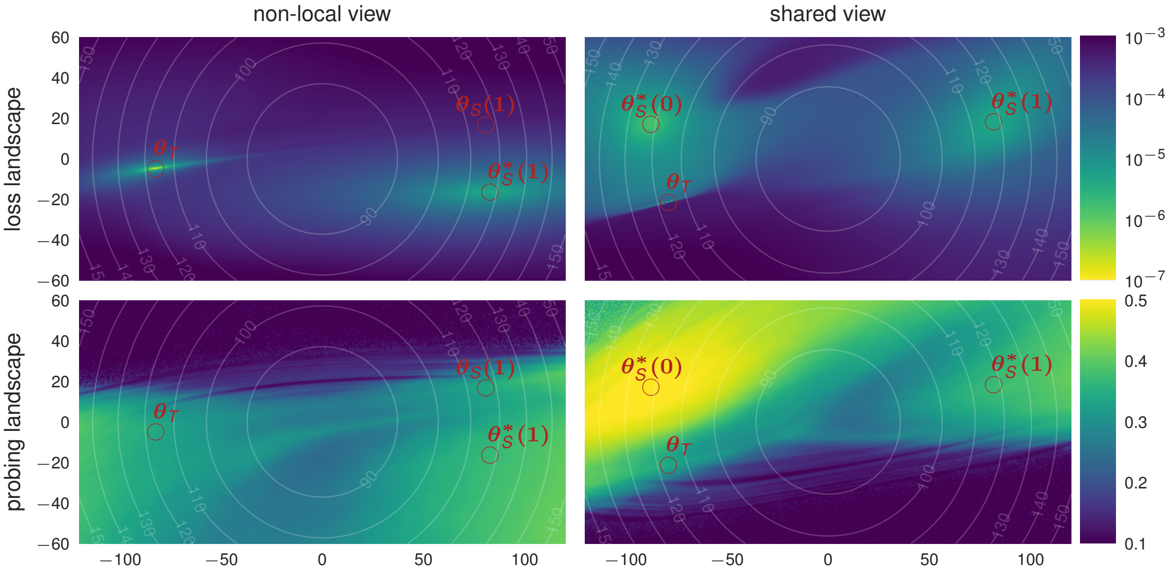

Figure 13. Orthogonal projection of the loss landscape in the parameter plane.

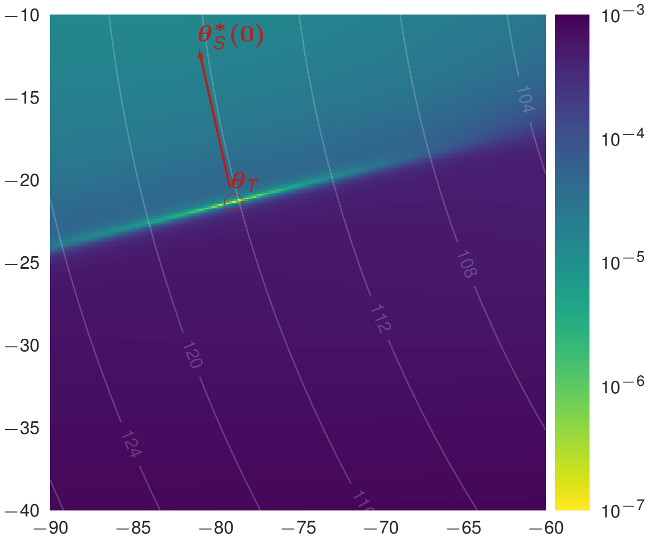

Figure 14. Higher resolution crop of the global optimum around the teacher.

The same visualization technique allows plotting the KL divergence between embeddings produced by the teacher and other parametrization in the plane. While in Fig,13, the basin of the local solution matches with the area of increased probing accuracy, such a correlation is not visible if one only considers the encoder.

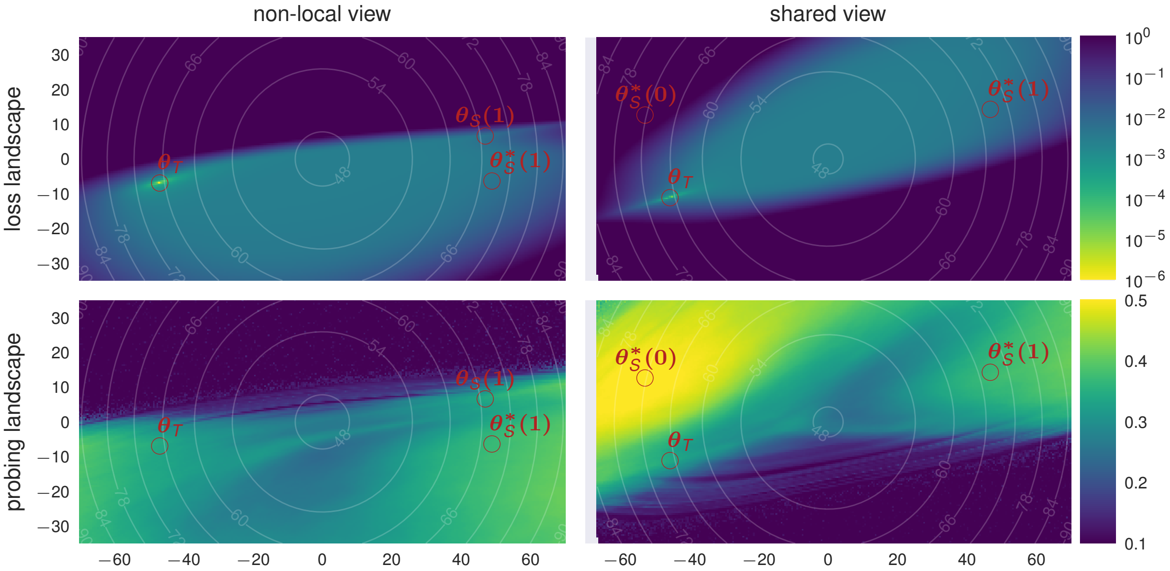

Figure 15. Orthogonal projection of the embedding KL divergence landscape in the parameter plane.

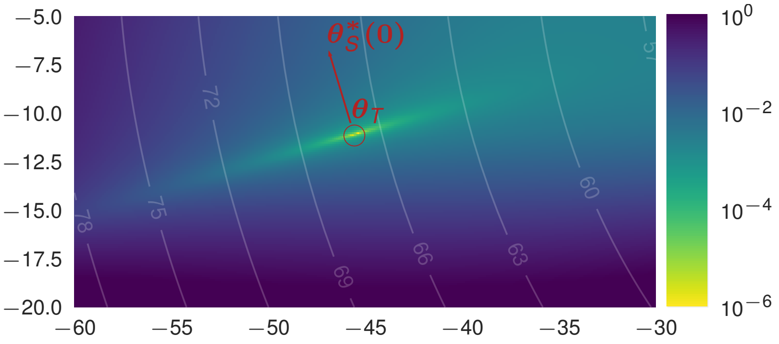

Figure 16. Higher resolution crop of the global optimum around the teacher.

D. Optimization Metrics

To convince ourselves that independently initialized students ($\alpha = 1$) are more difficult to optimize, we provide an overview of the KL-Divergence and distance from initialization for all $\alpha \in [0, 1]$ in Fig. 17. We observe that, indeed, for students initialized far away from their teacher, the loss cannot be reduced as efficiently. This coincides with worse probing performance. Note, however, that even the students with $\alpha = 1$ are able to outperform their teachers.

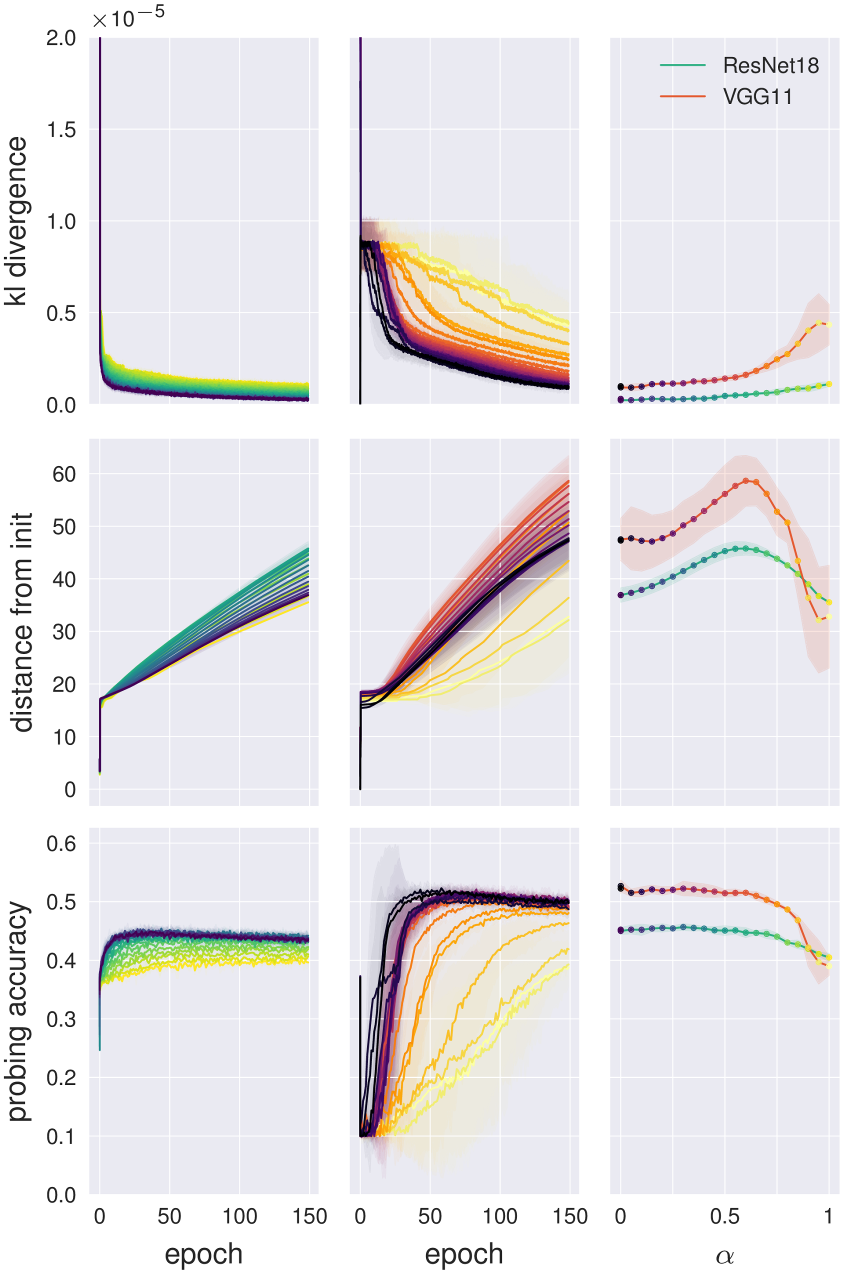

Figure 17. Optimization metrics for locality parameter α on *CIFAR10*. **Left:** *ResNet18*. **Middle:** *VGG11*. **Right:** Summary.

D.1. Restarting

An evident idea would be to restart the random teacher distillation procedure in some way or another. We considered several approaches, such as reintroducing the exponential moving average of the teacher, but were not successful. In Fig. 18, we show the most straightforward approach, where the student is reused as a new teacher, and a second round of distillation is performed. The gradient dynamics around the restarted student seem much more stable, and the optimization procedure does not even begin.

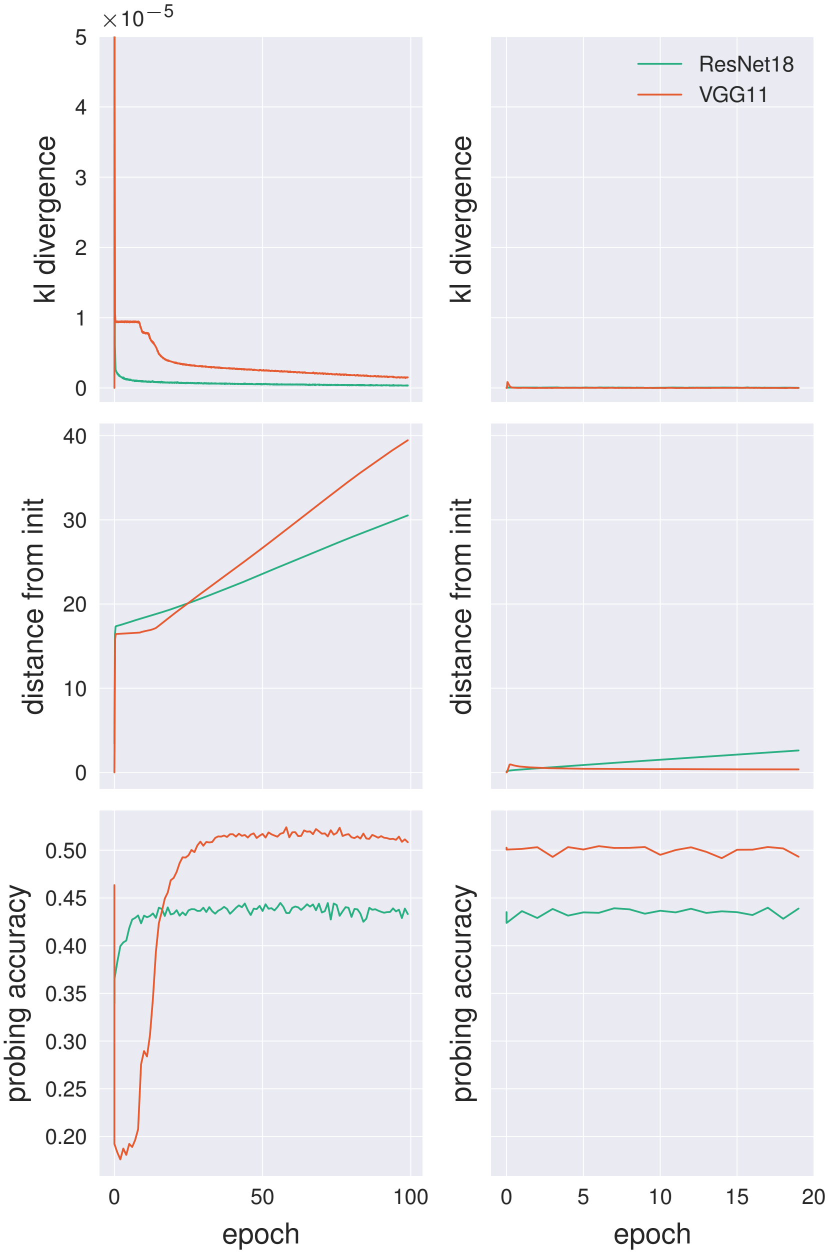

Figure 18. Restarting random teacher distillation on *CIFAR10* with *ResNet18* and *VGG11*. **Left:** First round of distillation. **Right:** Second round of distillation

E. Experimental Details

Our main goal is to demystify the properties of distillation in a simplistic setting, removing a series of ‘tricks’ used in practice. For clarity reasons, we here present a comprehensive comparison with the popular framework of DINO (Caron et al., 2021).

E.1. Architecture

Configuration	
Encoder	ResNet18&VGG1 from torchvision, without fc or classification layers (embedding $\in \mathbb{R}^{512}$) (ResNet18 adjusted stem for CIFAR: conv from 7x7 to 3x3, remove maxpool)
Projection Head	3-Layer MLP: $512 \rightarrow 2048 \rightarrow 2048 \rightarrow$ l2-bottleneck$(256) \rightarrow 2^{16}$ (GELU activation, no batchnorms, init: trunc_normal with $\sigma = 0.02$, biases=0)
L2-Bottleneck(in, mid, out)	for $x \in \mathbb{R}^{in}$, $W \in \mathbb{R}^{in \times mid}$, $b \in \mathbb{R}^{mid}$, $\tilde{V} \in \mathbb{R}^{mid \times out}$ 1. linear to bottleneck: $z = W^T x + b \in \mathbb{R}^{mid}$ 2. feature normalization: $\tilde{z} = z / \|\|z\|\|_2$ 3. weightnormalized linear: $y = \tilde{V}^T \tilde{z} \in \mathbb{R}^{out}$, with $\|\|\tilde{V}_{:,i}\|\|_2 = 1$ $\Rightarrow f_{\tilde{V},W}(x) = \tilde{V}^T \frac{W^T x + b}{\|\|W^T x + b\|\|_2}$ with $\|\|\tilde{V}_{:,i}\|\|_2 = 1$

E.2. Data

Configuration	DINO default	Random Teacher
Augmentations	Multicrop $(2 \times 224^2 + 10 \times 96^2)$ + SimCLR-like	**None** $(\mathbf{1 \times 32^2})$
Training batchsize	64 per GPU	256
Evaluation batchsize	128 per GPU	256

E.3. DINO Hyperparameters

Configuration	DINO default	Random Teacher
Teacher update	ema with momentum $0.996 \overset{cos}{\to} 1$	no updates
Teacher BN update	BN in train mode	BN in eval mode
Teacher centering	track statistics with momentum 0.9	not applied
Teacher sharpening	temperature 0.04 (paper: $0.04 \overset{lin}{\to} 0.07$)	temperature 1
Student sharpening	temperature 0.1	temperature 1
Loss function	opposite-crop cross-entropy	single-crop cross-entropy

E.4. Random Teacher Training

Configuration	DINO default	Random Teacher
Optimizer	AdamW	AdamW
Learning rate	$0 \overset{lin}{\to} 0.0005 \overset{cos}{\to}$ `1e-6` schedule	0.001 (torch default)
Weight decay	$0.04 \overset{lin}{\to} 0.4$ schedule	not applied
Gradient Clipping	to norm 3	not applied
Freezing of last layer	during first epoch	not applied

E.5. IMP Training

Configuration	Lottery Ticket Hypothesis (Frankle et al., 2020)	Random Teacher
Training Epochs	160	160
Optimizer	SGD (momentum 0.9)	SGD (momentum 0.9)
Learning rate	MultiStep: $0.1 \overset{\text{80 epochs}}{\to} 0.01 \overset{\text{40 epochs}}{\to} 0.001$	MultiStep: $0.1 \overset{\text{80 epochs}}{\to} 0.01 \overset{\text{40 epochs}}{\to} 0.001$
Weight decay	0.0001	0.0001
Augmentations	Random horizontal flip & padded crop (4px)	Random horizontal flip & padded crop (4px)