

# Overfreezing Meets Overparameterization: A Double Descent Perspective on Transfer Learning of Deep Neural Networks

Yehuda Dar  
Ben-Gurion University  
ydar@bgu.ac.il

Lorenzo Luzi  
Rice University  
enzo@rice.edu

Richard G. Baraniuk  
Rice University  
richb@rice.edu

## Abstract

We study the generalization behavior of transfer learning of deep neural networks (DNNs). We adopt the overparameterization perspective — featuring interpolation of the training data (i.e., approximately zero train error) and the double descent phenomenon — to explain the delicate effect of the transfer learning setting on generalization performance. We study how the generalization behavior of transfer learning is affected by the dataset size in the source and target tasks, the number of transferred layers that are kept frozen in the target DNN training, and the similarity between the source and target tasks. We show that the test error evolution during the target DNN training has a more significant double descent effect when the target training dataset is sufficiently large with some label noise. In addition, a larger source training dataset can delay the arrival to interpolation and double descent peak in the target DNN training. Moreover, we demonstrate that the number of frozen layers can determine whether the transfer learning is effectively underparameterized or overparameterized and, in turn, this may affect the relative success or failure of learning. Specifically, we show that too many frozen layers may make a transfer from a less related source task better or on par with a transfer from a more related source task; we call this case **overfreezing**. We establish our results using image classification experiments with the residual network (ResNet) and vision transformer (ViT) architectures.

## 1. Introduction

Transfer learning is a common practice for training deep neural networks (DNNs) based on pre-trained DNNs (e.g., [2, 12, 15, 17, 19]). The learning from scratch of the vast number of parameters in modern DNNs requires considerable amounts of training data and computational resources. Both data and compute power are often lacking, a key issue that can be addressed via transfer learning. The implementation of transfer learning requires a series of design

choices such as the number of layers to transfer from a given pre-trained (source) DNN and how much to allow them to change in the training of the target task. Specifically, the pre-trained transferred layers can be set fixed (“frozen”) or serve as initialization in a short (fine tuning) or long training process. These design choices are pivotal and affect the generalization performance in intricate ways that are still far from being sufficiently understood [7, 10, 13, 16, 20], leaving the common practice to extensively rely on trial-and-error.

Modern DNNs are *overparameterized* models, namely, they have much more learnable parameters than training data samples; moreover, such models are usually trained to interpolate (i.e., perfectly fit, having training error numerically zero) their training data [1, 5, 21, 22]. Despite interpolating their training data, DNNs can generalize excellently to (test) data beyond their training dataset; this contradicts the conventional machine learning guideline that associates overfitting with poor generalization performance.

Many overparameterized models have test errors that follow a *double descent* shape when examined with respect to the learned model complexity (e.g., number of learnable parameters) [1]. This double descent behavior has two parts: In the underparameterized regime, where the learned models cannot interpolate its training data, the test error follows the “U-shape” of the classical bias–variance tradeoff; in the overparameterized regime, where the learned model interpolates its training data, the test error can decrease as the learned model complexity increases (e.g., more learnable parameters).

The double descent phenomenon in DNNs has received a detailed empirical analysis in [14]. Specifically, evaluating the test and train errors (i.e., the relative portion of wrong classification in the test and train datasets, respectively) for a range of DNN widths and in each of the many training epochs shows two kinds of double descent phenomena of the test error: (i) as a function of the model width, and (ii) as a function of the training epoch; both have a peaking test error around or towards the entrance to the interpolation regime where the train error is zero (or numerically close to zero). These double descent phenomena are usually more

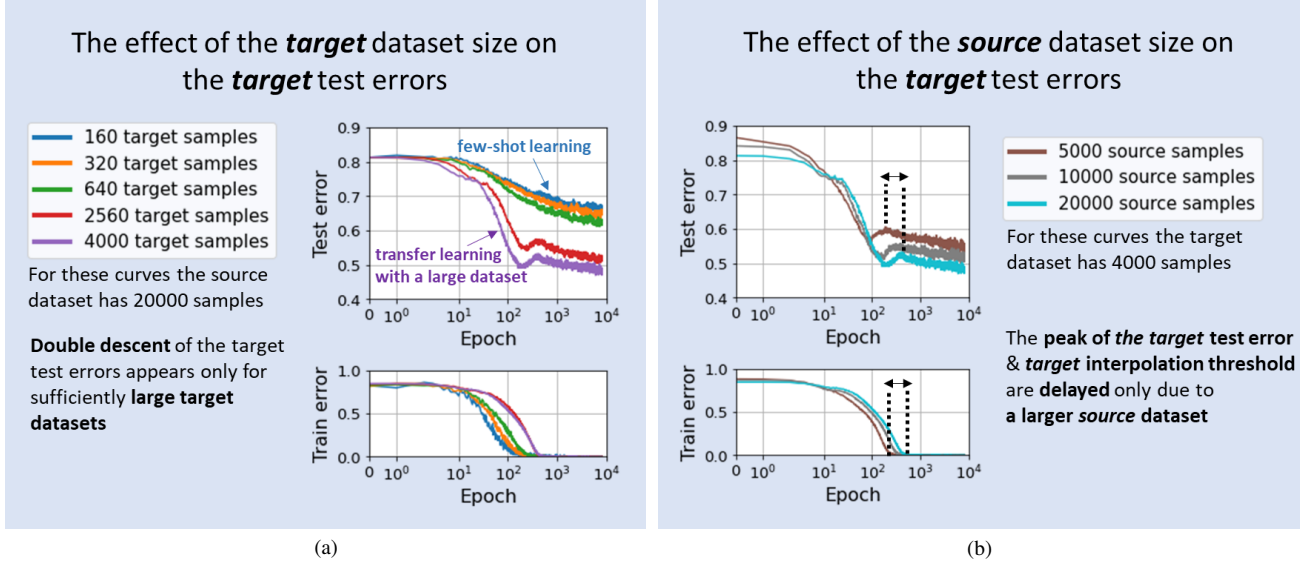


Figure 1. Double descent in transfer learning of a ResNet-18 for classification of 40 classes from CIFAR-100 (target data has 20% label noise). The pre-trained source model is a ResNet-18 for classification of another subset of 40 classes from CIFAR-100 (source data does not have label noise). This figure shows how the training evolution of the errors of the target model is affected by (a) the target training dataset size for the same pre-trained source model, and (b) the source training dataset size for the same target dataset. The legends specify the total size (i.e., number of samples for all the 40 classes together) of the examined training datasets. See more details in Section 3.

noticeable when the data is noisy (e.g., label noise in classification problems). The results in [14] consider diverse learning settings and, yet, all of them are for learning DNNs from scratch.

In this paper we go considerably beyond the scope of [14] and study the generalization and double descent phenomenon in transfer learning. Our first contribution is the characterization of the transfer learning settings where the double descent phenomenon is likely to emerge. As in learning from scratch, label noise in the training dataset is an important promoter of double descent; yet, in transfer learning, such label noise needs to exist only in the target dataset and not necessarily in the source dataset. Moreover, double descent in transfer learning requires a sufficiently large target dataset (Fig. 1a); this implies that double descent is less likely to appear in few-shot learning settings (i.e., when a very limited number of training samples are available for the target task [19]). Also, we show that the size of the *source* training dataset importantly determines the speed of which the transfer learning of the target task achieves interpolation (of the target dataset) and arrives to the corresponding double descent peak (Fig. 1b).

Our second contribution is in showing how freezing the transferred pre-trained layers affects the generalization behavior. Freezing layers determines the number of learnable parameters in the target DNN and, hence, it is a unique way of transfer learning to control the parameterization level without changing the overall DNN architecture. In trans-

fer learning with a large target dataset, we show that freezing (relatively many) layers can eliminate the double descent that exists when no or moderate freezing is applied. In transfer learning with a small target dataset (but not extremely small), we show that freezing layers can induce a double descent phenomenon that does not appear when no or moderate freezing is applied.

Our third contribution is the examination of the effect of the similarity between the source and target tasks on the generalization behavior, especially in cases with double descent behaviors. Higher task similarity is usually expected to yield better generalization in transfer learning. Interestingly, our results exemplify that *transfer from a less related task may generalize better or on par with transfer from a more related task*. We show that such scenarios may result from freezing too many of the transferred layers, namely, *overfreezing* (see Fig. 2). Specifically, this overfreezing can be detrimental when the target model is around its interpolation threshold, a property that also depends on the training duration. Hence, when double descent is likely to emerge, a good utilization of a pre-trained model from a similar task requires to be extra careful in choosing the training duration and number of frozen layers.

### 1.1. Related Work

Previous works provide analytical theories for the double descent phenomenon in transfer learning of simple models such as linear [3, 4] and two-layer nonlinear networks [8].

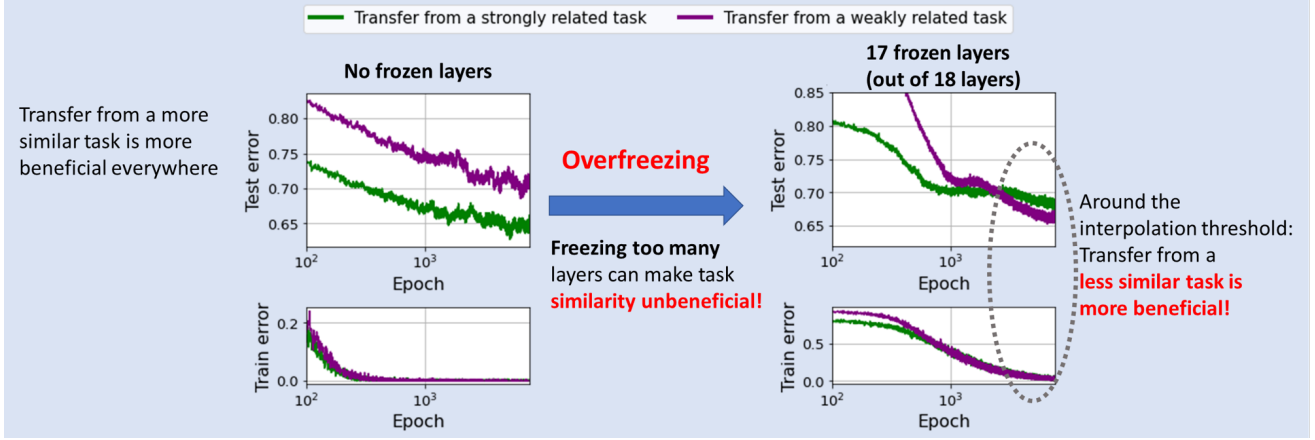


Figure 2. An example for *overfreezing* in transfer learning of a ResNet-18 model for classification of 40 classes from CIFAR-100. The sizes of the target and source datasets are 320 and 20000 samples, respectively. The classification tasks are defined in Section 2 (where the weakly related source task in this figure corresponds to definition #1). More details and examples are provided in Section 5.

This work is the first to examine the double descent phenomenon in transfer learning of DNNs; specifically, we provide new insights that cannot be obtained in simple models and are directly relevant to the deep learning practice.

More broadly, this work also relates to the ongoing research efforts to elucidate the conditions for a successful transfer learning of DNNs [7, 10, 13, 16, 20]. Compared to these works, our contributions are in the examination of (i) the double descent phenomenon, (ii) the evolution of the test and training errors throughout the DNN training and its dependency on the specific transfer learning setting, e.g., number of frozen layers and the similarity between the source and target tasks.

## 1.2. Paper Outline

This paper is organized as follows. In Section 2 we provide the details of the transfer learning settings and the classification problems. In Section 3 we show the effect of the source and target dataset sizes on the emergence of the double descent phenomenon in transfer learning. In Section 4 we demonstrate the effect of freezing layers on the double descent phenomenon. In Section 5 we examine transfer from tasks at various similarity levels and show the detrimental effect of overfreezing. We conclude the paper in Section 6. Additional experimental details are provided in the Appendices.

## 2. The Transfer Learning Settings

This paper studies transfer learning of DNNs in the ResNet-18 [9] and the Vision Transformer (ViT) [6] architectures (see Appendix A for more details on the architectures). We consider the following image classification problems that are defined using data from the CIFAR-10 and

CIFAR-100 datasets<sup>1</sup> [11]. We consider the following transfer learning settings:

1. *10-class target task*: Classification among 10 classes from CIFAR-100. These 10 classes are the cloud, bus, butterfly, raccoon, cattle, fox, crab, camel, sea, pickup truck; and they were chosen due to their relative similarity to the CIFAR-10 classes. For this target task we consider in this paper a strongly related, 10-class source task that corresponds to the CIFAR-10 datasets. Unless otherwise specified, this source DNN was trained using all the 50K training samples of the CIFAR-10 dataset.
2. *40-class target task*: Classification among 40 classes from CIFAR-100. Recall that the CIFAR-100 dataset includes 100 classes that are grouped into 20 super-classes of 5 classes each. The 40 classes in the target task are composed by two classes from each of the 20 CIFAR-100 superclasses (see more details in Appendix B). For this target task we consider in this paper the following source tasks:
  - (a) strongly related, 40-class source task: The 40 classes in this source task are composed by two classes from each of the 20 CIFAR-100 super-classes; all these 40 classes were not included in the 40-class target task (see details in Appendix B). This source task is relatively related to the target task due to having the same balance of 2-classes per superclass. Unless otherwise specified, the source DNN was trained using all the 20K training samples of the corresponding 40 classes in the CIFAR-100 dataset.

<sup>1</sup><https://www.cs.toronto.edu/~kriz/cifar.html>

- (b) Weakly related 40-class source task #1: This source task is less related to the target task due to having a different balance of classes per superclass. Specifically, from the first 10 superclasses of CIFAR-100 we take three classes each, and from the other 10 superclasses we take only one class each. All these 40 classes were not included in the 40-class target task, and the corresponding training dataset includes 20K training samples from CIFAR-100.
- (c) Weakly related 40-class source task #2: This source task is even less related to the target task than the previous weakly related task. The weaker relation is due to not including classes from all the 20 superclasses of CIFAR-100. Specifically, here, the 40 classes are taken only from the first 14 superclasses of CIFAR-100. From the first 13 superclasses of CIFAR-100 we take three classes each, and from the 14th superclass we take only one class. All these 40 classes were not included in the 40-class target task, and the corresponding training dataset includes 20K training samples from CIFAR-100.

All the source DNNs were trained on datasets without any (artificially added) label noise and for 4000 training epochs with the Adam optimizer, batch size 128, learning rate 0.0001 for source tasks with 10 or less classes, and learning rate 0.00005 for source tasks of 40 classes.

In contrast to the relatively large training dataset of the source tasks, the target tasks have limited data for training (a challenge that often motivates using transfer learning in practice). We will consider a variety of dataset sizes, ranging from the few-shot learning setting (e.g., less than 10 training samples per class) and up to larger datasets (e.g., with 100 samples per class) that are still considerably smaller than datasets such as the complete CIFAR-10 or CIFAR-100. Due to the relatively small size of the target dataset, we define the batch size to be twice the number of training samples per class, but not more than 128 samples or 25% of the entire dataset. The training is for 8000 epochs using the Adam optimizer with a constant learning rate of 0.00001. Similar to the studies of double descent in the learning of DNNs from scratch [14, 18], we use long training and low learning rate in order to evaluate the “natural” generalization behavior in a smooth manner; e.g., in contrast to using learning rate schedulers that may affect the epoch-wise error curves and interfere with our goal of studying the existence or lack of the double descent phenomenon.

We will give particular attention to generalization trends that stem by using different number of frozen layers (as defined in Section 4) and different similarity levels between

the source and target tasks.

### 3. When Does Double Descent Emerge in Transfer Learning?

Label noise is known as a promoter of significant double descent phenomena in DNNs [14]. Accordingly, in learning the target DNN from scratch, the test error follows a double descent trend along the training epochs when the target dataset has label noise, but usually not when it is noiseless (Fig. 3 and the additional results for the ViT architecture and the 10-class problem in Appendix C.1).

In this section we consider the effect of sizes of the target and source datasets on the emergence of the double descent phenomenon. For a start, we consider transfer learning where the entire target DNN is trained from its initialization with the pre-trained source parameters (in Section 4 we will also examine transfer learning with frozen layers).

#### 3.1. Double Descent Can Emerge in Transfer Learning, but is Less Likely in Few-Shot Learning

The test error in transfer learning is more likely to follow an epoch-wise double descent if the target dataset is *sufficiently large* and has label noise (see Fig. 3 and the corresponding results for the ViT architecture and the 10-class problem in Appendix C.1). Transfer learning is often motivated by limited availability of data for training and, therefore, the lack of double descent in the case of a small target dataset is of great importance. Specifically, in few-shot learning the target dataset is extremely small and, thus, the occurrence of double descent is even less probable. The lack of a double descent peak of the test error implies that the generalization performance is less sensitive to the exact duration of training and stopping criterion.

As noise label in the target dataset is crucial for having a double descent phenomenon, from this point and on we will consider noisy target datasets unless otherwise specified.

#### 3.2. Larger Source Datasets Can Delay Interpolation and Double Descent in the Target DNN

Now we turn to examine the possible effect of the size of the *source* training dataset on the evolution of the errors in the transfer learning of the *target* DNN. Recall that the source DNN is trained to interpolate its (noiseless) source dataset, and then it is utilized as an initialization for the target DNN training from its (target) dataset. Here the experiment is designed as follows: For a given target training dataset, we examine several transfer learning options for the target DNN — the difference among these options is in the size of the source dataset that the pre-trained source model was trained on. Our results (Figs. 1b, 4, and the additional results in Appendix C.2) demonstrate that a larger source dataset can delay (i) the arrival of the target DNN to interpolation of its own dataset, (ii) the double descent peak of



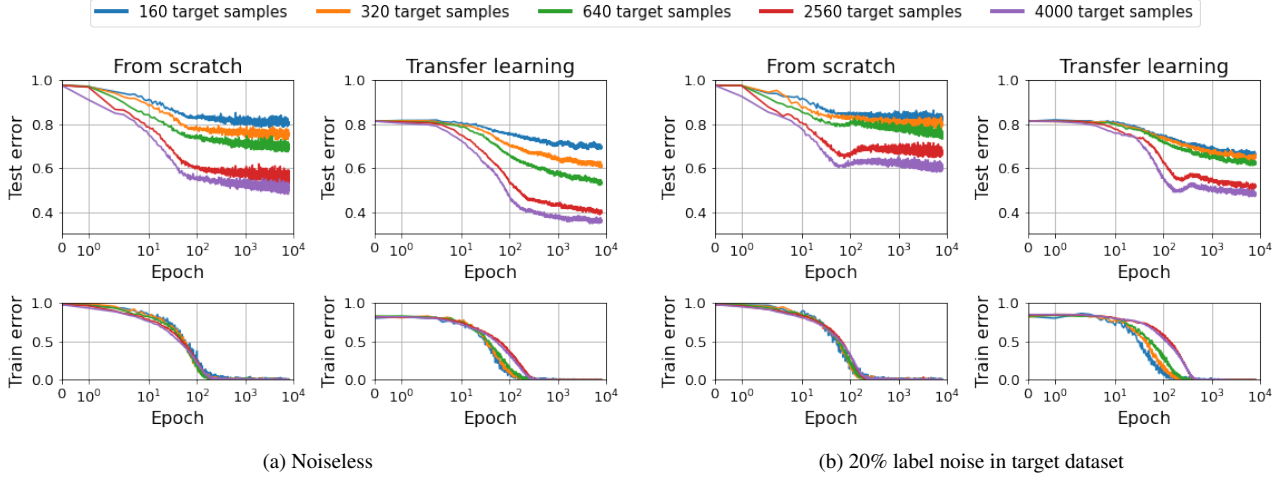


Figure 3. Evaluation of ResNet-18 training for the 40-class target task. The transfer learning is from the strongly related source task of 40 classes. The definitions of the 40-class source and target tasks are provided in Section 2. Each curve color corresponds to another size of the target dataset. Note that the double descent emerges in transfer learning only for noisy (target) datasets that are sufficiently large.

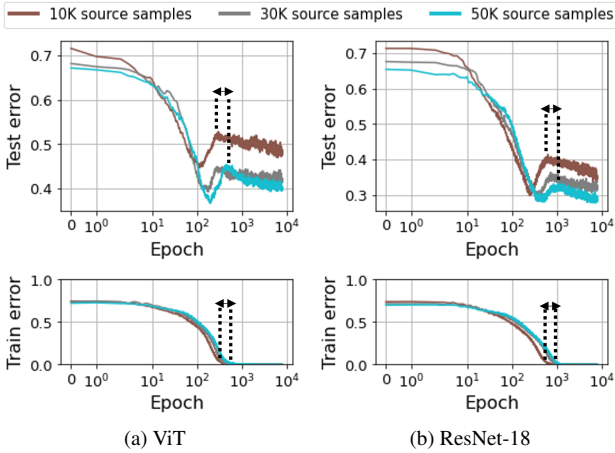


Figure 4. The effect of the source training dataset size on the training evolution of the errors of the target model. This figure considers transfer learning from the 10-class strongly related problem to the target 10-class problem. The target training dataset includes 2560 samples. See results for other dataset sizes in Appendix C.2.

the corresponding double descent. This interesting behavior seems to stem from the extra training time that is required for reverting the accurate interpolation of a larger source dataset in the target DNN initialization, which is necessary for the interpolation of the target training dataset<sup>2</sup>. The de-

<sup>2</sup>Although a smaller (noiseless) source dataset implies worse performance on the source task and, thus, probably also a higher test error in the initialization of the target DNN (i.e., errors in epoch 0), we claim that these differences in the errors at the initialization are not the cause for the delay in the interpolation and double descent peak in the target training. This

lay in the double descent peak may (but not necessarily) induce scenarios where using a pre-trained model that was trained on a smaller source dataset is more beneficial (see, e.g., in Fig. 4a where the double descent peak of the cyan curve is above the gray curve for a short interval of epochs).

## 4. Freezing Transferred Layers Can Induce or Eliminate Double Descent

A prevalent transfer learning strategy is to take layers from the pre-trained (source) DNN and setting them fixed (i.e., “freezing” them) in the target DNN without further adjustments or fine tuning. This approach reduces the number of learnable parameters in the target DNN and, by that, determines its *actual* parameterization level.

Popular transfer learning implementations may significantly vary in the relative portion of the DNN that is kept frozen. One typical design is to freeze the entire DNN but the last (top) layer; in contrast, another typical design is to freeze only several early layers of the DNN. In order to conduct transfer learning by partially freezing the network, we need to understand how the generalization performance and its evolution during training are affected by the freezing.

### 4.1. The Examined Freezing Modes

For the ResNet-18 model, we consider six levels of freezing in transfer learning. The levels are denoted numerically from ‘Frozen-0’ to ‘Frozen-5’. The Frozen-0 level

is supported by our results later in this paper for transfer from problems at different similarity levels but with the same size of the source training dataset, which makes the time of arrival to interpolation in the target training identical despite the difference in the target errors at the initialization, see Fig. 6.

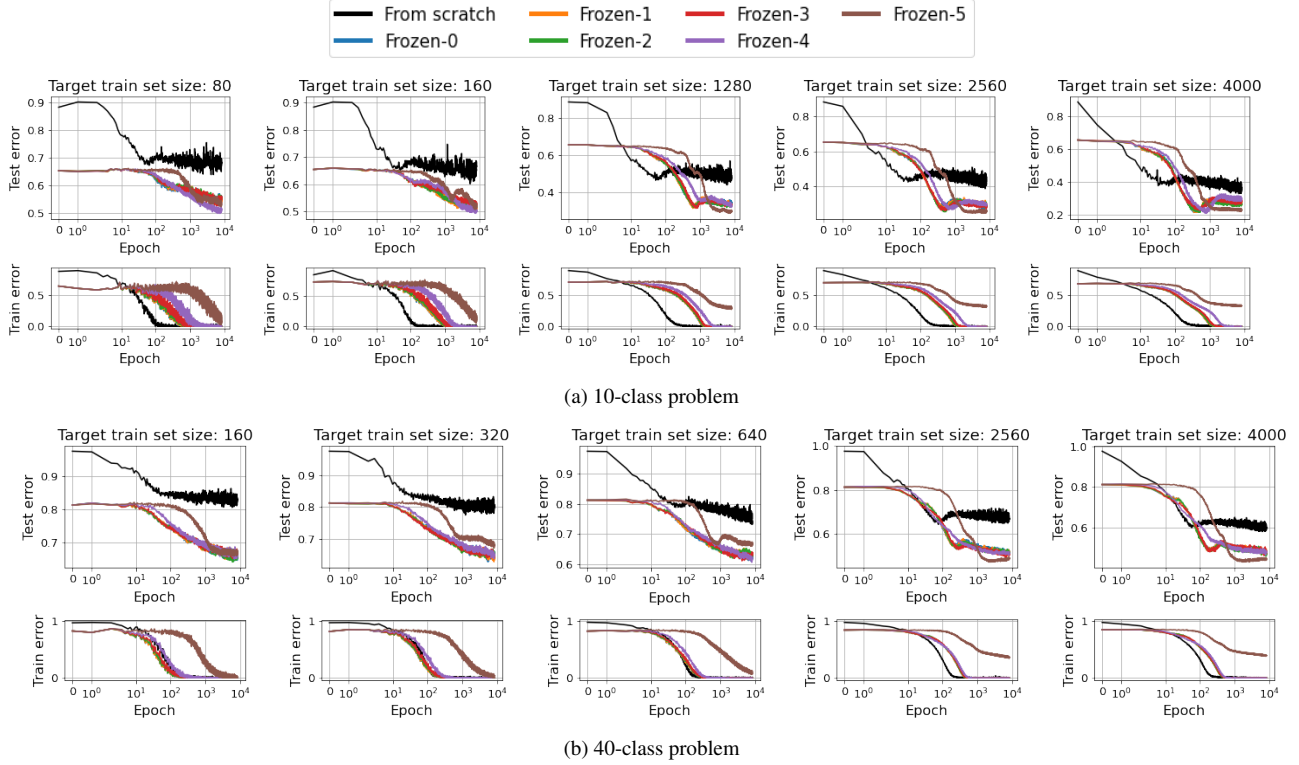


Figure 5. Evaluation of transfer learning at levels of freezing and comparison to learning from scratch. The target training datasets are with 20% label noise. Subfigures on the same row differ in the training dataset size of the target task. The model is ResNet-18. The transfer learning are from the related source tasks. Here, for better visibility, the value range of the vertical axis may differ among the subfigures.

refers to using the entire pre-trained DNN as initialization of the target DNN without freezing any of the layers. The Frozen-5 level refers to freezing the entire DNN except for the last fully-connected layer (this last layer is trained via fine tuning the values transferred from the co-located layer of the DNN). The definition of all the examined freezing levels are described in Table 1 (more details on the ResNet-18 layout are provided in Appendix A).

For the ViT architecture we consider 8 levels of freezing: from ‘Frozen-0’ (no freezing at all) to ‘Frozen-7’ (the entire DNN but the last fully-connected layer is frozen). ‘Frozen-1’ has the first (patch embedding) layer frozen. ‘Frozen-2’ to ‘Frozen-7’ correspond to the gradual freezing of the 6 transformer blocks (each of these blocks includes an attention module, etc.) A more detailed definition of the ViT freezing levels is provided in Appendix A.

#### 4.2. Freezing Layers Can Eliminate Double Descent for Large Target Datasets

In Section 3.1 we showed that, in transfer learning with no frozen layers (i.e., the ‘Frozen-0’ option), double descent emerges when the target dataset is sufficiently large. Here we consider transfer learning with frozen layers. Naturally,

freezing layers in the target DNN effectively reduces its parameterization level (simply because less parameters are learned). Accordingly, freezing many layers can make the

Level	Frozen ResNet-18 components						# frozen layers
	Conv layer 1	ResNet block 1	ResNet block 2	ResNet block 3	ResNet block 4	FC layer	
Frozen-0							0
Frozen-1	✓						1
Frozen-2	✓	✓					5
Frozen-3	✓	✓	✓				9
Frozen-4	✓	✓	✓	✓			13
Frozen-5	✓	✓	✓	✓	✓		17

Table 1. Definition of the utilized freezing levels for ResNet-18. Each row specifies the frozen ResNet components in a particular freezing level. Recall that each of the ResNet-18 blocks is consisted of 4 convolutional layers.

target DNN effectively underparameterized, namely, being far from interpolating the target training dataset. Moreover, the test error of such underparameterized DNN does not follow an epoch-wise double descent shape during training.

This elimination of a double descent behavior due to freezing many layers is evident in Fig. 5, specifically, see the error curves of the ‘Frozen-5’ setting (the brown curves) for target dataset sizes larger than 1280 samples. This behavior is further supported by the additional results for the ViT architecture in Appendix D.1.

### 4.3. Freezing Layers Can Induce Double Descent for Small Target Datasets

Now we turn to show that freezing layers can induce a double descent phenomenon when the target dataset is relatively small. Recall that the findings in Section 3.1 show that double descent is not likely to occur when the target dataset is too small and the transfer learning is without frozen layers (e.g., see the test error curves for transfer learning of a ResNet-18 DNN with target dataset sizes up to 640 samples in Fig. 3b). Now we reconsider the same learning problems but with frozen layers in the transfer learning of the target DNN.

The results in Fig. 5 for transfer learning for the 40-class problem with target dataset size 640 or 320 show the following: For transfer learning with none or several frozen layers (specifically in this example the Frozen-0 to Frozen-4 settings) there is no double descent; in contrast, when many layers are frozen (in this example the Frozen-5 setting which is denoted by the brown curve) the double descent phenomenon emerges.

While freezing many layers can induce double descent for relatively small target datasets, we also observe that for extremely small target datasets there is no evident double descent (e.g., see the brown curves for the 160 samples dataset in the 40-class example or the 80 samples dataset in the 10-class problem in Fig. 5). This implies that in extreme cases of few-shot learning we do not expect double descent phenomena. The corresponding results for the ViT architecture (see Appendix D.2) further confirm our findings.

## 5. Overfreezing: The Less-Related Task Can be Better to Transfer From

The similarity between the source and target tasks naturally affects the generalization performance in transfer learning. In this section we examine whether transfer from a more related task is necessarily better than transfer from a less related task.

Our results for the ResNet-18 in Fig. 6 (and the corresponding results in Appendix E for ViT) show that the test error in transfer learning from a more related task usually outperforms the corresponding test error in transfer learning from a less related task; this observation is aligned with

the basic intuition that it is better to transfer from a more similar task. Yet, a closer look at our results reveals cases where **transfer from a less related task can generalize on par or better than a transfer from a more related task**.

In general, a particular learning setting whose test error follows a double descent shape has a degraded generalization performance in the interval of epochs around the double descent peak of its test error. This may cause the transfer learning from a more related task to be less beneficial than a transfer learning from a less related task (see, e.g., Figs. 2 and the additional examples in Appendix E).

We define *overfreezing* as a case where the transfer from a more related task is unbeneficial (in generalization performance) due to freezing too many of the transferred layers. Figures 2, 7a, 7b show examples for overfreezing. Namely, transfer from the more related task is always more beneficial when there are no frozen layers; but when too many layers are frozen the transfer from a less related task can be better or on par with the transfer from the more related task. Put differently, in the case of overfreezing we do not achieve the performance gain that we expect from using a pre-trained model from a more related task. The overfreezing in Fig. 2 is associated with the performance degradation due to the double descent peak of the transfer from the more related task, but the overfreezing example in Fig. 7b occurs without any double descent phenomenon.

Interestingly, despite the gap in generalization performance, the task similarity level does not noticeably affect the number of epochs required for interpolation of the target dataset. This further supports our finding from Section 3.2 that the interpolation threshold location in the target DNN training depends on the size of the source training dataset; specifically, here we show that the task similarity and target errors at epoch 0 do not affect the location of the interpolation threshold of the target task.

## 6. Conclusion

In this work we have developed and validated a new approach to research of the generalization performance of transfer learning. We have explicitly considered overparameterization concepts such as the double descent phenomenon and interpolation of the training data. Our findings elucidate how the generalization behavior of transfer learning is affected by the source and target dataset sizes, the number of frozen layers, and the similarity between the source and target tasks. Specifically, we have identified cases where transfer from a related task is not better than transfer from a less related task, and showed that these cases stem from *overfreezing* — namely, freezing too many transferred layers in the target DNN. We believe that our perspective and insights open a new direction to explore generalization in transfer learning in various problems and settings.

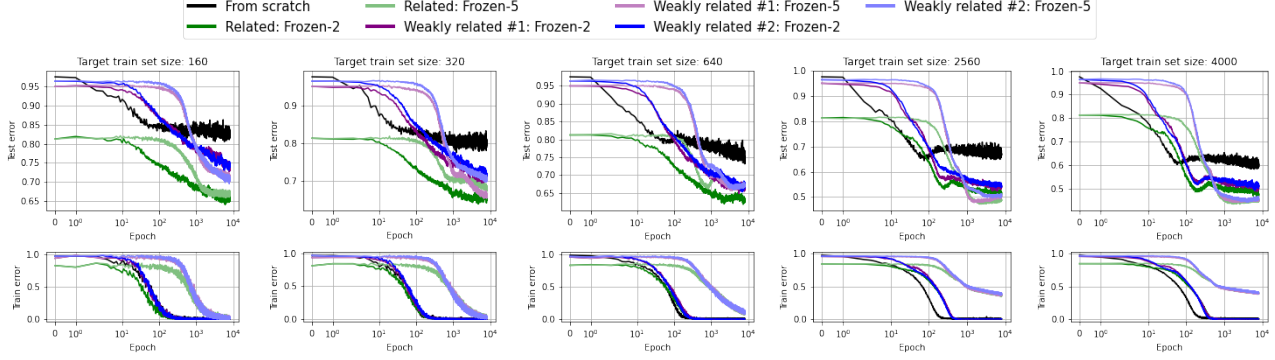
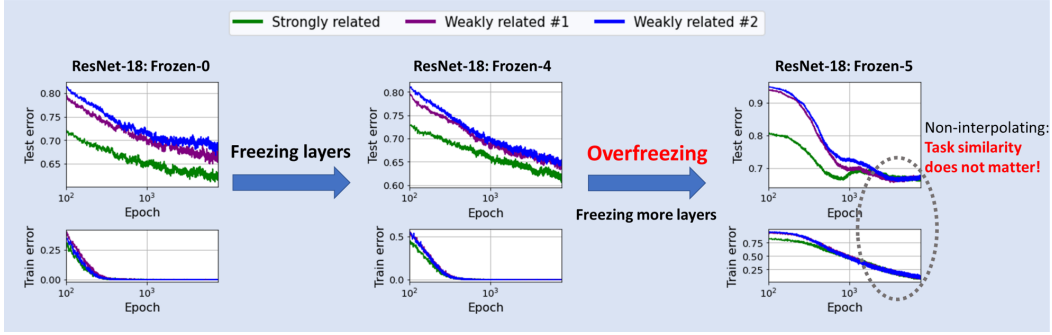
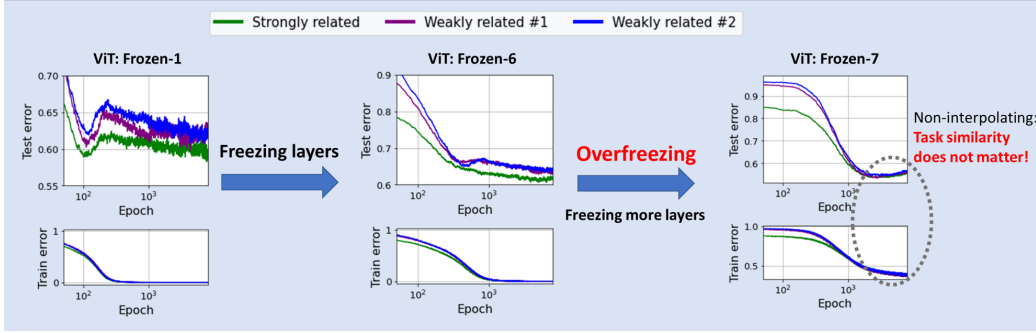


Figure 6. Generalization performance in the 40-class target task using transfer learning from a related and from weakly related source tasks (all are 40-class problems). The target training datasets are with 20% label noise. The subfigures on the same row differ in the training dataset size of the target task. The model is ResNet-18. Here, the value range of the vertical axis may differ among the subfigures.



(a) ResNet-18 and a target dataset of 640 samples



(b) ViT and a target dataset of 4000 samples

Figure 7. Examples for *overfreezing* in transfer learning for classification of 40 classes from CIFAR-100. The sizes of the source datasets is 20000 samples. The classification tasks are defined in Section 2.

## Acknowledgments

This work was supported by NSF grants CCF-1911094, IIS-1838177, and IIS-1730574; ONR grants N00014-18-12571, N00014-20-1-2534, and MURI N00014-20-1-2787; AFOSR grant FA9550-22-1-0060; and a Vannevar Bush Faculty Fellowship, ONR grant N00014-18-1-2047.

## References

- [1] M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019. 1
- [2] Y. Bengio. Deep learning of representations for unsupervised and transfer learning. In *ICML workshop on unsupervised and transfer learning*, pages 17–36, 2012. 1
- [3] Y. Dar and R. G. Baraniuk. Double double descent: On gen-



- eralization errors in transfer learning between linear regression tasks. *arXiv preprint arXiv:2006.07002*, 2020. **2**
- [4] Y. Dar, D. LeJeune, and R. G. Baraniuk. The common intuition to transfer learning can win or lose: Case studies for linear regression. *arXiv preprint arXiv:2103.05621*, 2021. **2**
- [5] Y. Dar, V. Muthukumar, and R. G. Baraniuk. A farewell to the bias-variance tradeoff? An overview of the theory of overparameterized machine learning. *arXiv preprint arXiv:2109.02355*, 2021. **1**
- [6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. **3**
- [7] L. Ericsson, H. Gouk, and T. M. Hospedales. How well do self-supervised models transfer? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5414–5423, 2021. **1, 3**
- [8] F. Gerace, L. Saglietti, S. S. Mannelli, A. Saxe, and L. Zdeborová. Probing transfer learning with a model of synthetic correlated datasets. *Machine Learning: Science and Technology*, 3(1), 2022. **2**
- [9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. **3**
- [10] S. Kornblith, J. Shlens, and Q. V. Le. Do better imagenet models transfer better? In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2661–2671, 2019. **1, 3**
- [11] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. 2009. **3, 10**
- [12] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning (ICML)*, pages 2208–2217, 2017. **1**
- [13] T. Mensink, J. Uijlings, A. Kuznetsova, M. Gygli, and V. Ferrari. Factors of influence for transfer learning across diverse appearance domains and task types. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. **1, 3**
- [14] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, (12), 2021. **1, 2, 4, 9, 10**
- [15] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. **1**
- [16] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio. Transfusion: Understanding transfer learning for medical imaging. In *Advances in neural information processing systems*, pages 3347–3357, 2019. **1, 3**
- [17] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016. **1**
- [18] G. Somepalli, L. Fowl, A. Bansal, P. Yeh-Chiang, Y. Dar, R. Baraniuk, M. Goldblum, and T. Goldstein. Can neural nets learn the same model twice? investigating reproducibility and double descent from the decision boundary perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13699–13708, 2022. **4, 10**
- [19] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3), 2020. **1, 2**
- [20] A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese. Taskonomy: Disentangling task transfer learning. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3712–3722, 2018. **1, 3**
- [21] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. In *ICLR*, 2017. **1**
- [22] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021. **1**

## Appendices

These appendices support the main paper as follows. Appendix **A** provides additional details on the DNN architectures that we examine in this paper. Appendix **B** provides additional details on the classification problems that we consider in this paper. Appendices **C**, **D**, **E** include additional experimental results that further support Sections **3**, **4**, **5** in the main text.

The indexing of figures in the Appendices below is prefixed with the letter of the corresponding Appendix. Other references correspond to the main paper.

## A. Additional Details on the Examined DNN Architectures

In this paper we examine two DNN architectures: ResNet-18 and ViT. In this Appendix we describe the relevant details of the examined architectures, in addition to the details already provided in Sections **2** and **4** (specifically, note that the technical details of the training optimization are provided in Section **2** and not here).

### A.1. ResNet-18

The examined ResNet-18 has the common structure and dimensions according to the number of classes in the addressed classification problem. We use the same ResNet-18 form as in [14] that is based on the implementation from <https://github.com/kuangliu/pytorch-cifar>. Namely, this is the Preactivation ResNet-18 with the following structure:

- convolutional layer

- 4 ResNet-18 blocks, each of these ResNet blocks includes a sequence of batch norm, ReLu, convolution, batch norm, ReLu, convolution.
- average pooling
- fully connected layer

The dimensions of the layers are as in [14] for the standard width parameter  $k = 64$ .

There is a total of 18 layers with learnable parameters; specifically, the first and the last layers, and four learnable layers in each of the ResNet blocks. Accordingly, as explained in Section 4.1, we consider 6 levels of freezing in ResNet-18:

- Frozen-0: No frozen layers at all, i.e., the entire target DNN can be trained from its initialization using the pre-trained source DNN.
- Frozen-1: Only the first convolutional layer is frozen with its pre-trained parameters in the target DNN training.
- Frozen-2: Only the first convolutional layer and the first Resnet-18 block are frozen.
- Frozen-3: Only the first convolutional layer and the first two Resnet-18 blocks are frozen.
- Frozen-4: Only the first convolutional layer and the first three Resnet-18 blocks are frozen.
- Frozen-5: Only the first convolutional layer and the four Resnet-18 blocks are frozen. Namely, only the last fully-connected layer in the target DNN can be trained.

## A.2. Vision Transformer (ViT)

We consider a ViT architecture in the same form and dimensions as in [18], namely, a patch size of  $4 \times 4$  pixels, 6 transformer layers (each of these is actually a block of several layers), and 8 heads. This ViT architecture allows us to extensively examine its generalization behavior at various settings. The implementation of the ViT is based on the code from <https://github.com/lucidrains/vit-pytorch>.

The structures and layers of the examined ViT can be described as follows:

- Patch embedding
- 6 transformer layers, each of these layers is actually a block of layers that includes norm layers, a multi-head attention sub-block, and an MLP
- an MLP head.

As explained in Section 4.1, we consider 8 levels of freezing in ViT:

- Frozen-0: No frozen layers at all, i.e., the entire target DNN can be trained from its initialization using the pre-trained source DNN.
- Frozen-1: Only the patch embedding layer is frozen with its pre-trained parameters in the target DNN training.
- Frozen-2: Only the patch embedding layer and the first transformer layer are frozen.
- Frozen- $r$  for  $r \in \{3, \dots, 6\}$ : Only the patch embedding layer and the first  $r - 1$  transformer layers are frozen.
- Frozen-7: Only the patch embedding layer and the six transformer layers are frozen. Namely, only the MLP head at the end of the target DNN can be trained.

## B. Additional Details on the Examined Classification Problems

In this paper we consider 10-class and 40-class classification problems, these problems are defined based on the CIFAR-10 and CIFAR-100 datasets [11] as explained in Section 2. In this Appendix we provide additional details that were not included in Section 2 due to the limited space.

### B.1. The 10-Class Problem

The CIFAR-10 dataset includes the following 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck. These are the 10 classes that define the 10-class *source* task that we examine.

As explained in Section 2, we consider a 10-class *target* problem based on the following classes, which are taken from the CIFAR-100 dataset: cloud, bus, butterfly, raccoon, cattle, fox, crab, camel, sea, pickup truck. These classes were chosen due to their relative similarity (relative w.r.t. the other classes in CIFAR-100) to the CIFAR-10 classes. Accordingly, we consider the source and target 10-class problems as being strongly related.

### B.2. The 40-Class Problem

As explained in Section 2, in this paper we consider several 40-class classification problems that are defined based on classes from the CIFAR-100 dataset. The CIFAR-100 dataset includes 100 classes that are grouped into 20 superclasses as described in Table B.1.

The 40 classes in the 40-class *target* task are composed by two classes from each of the 20 CIFAR-100 superclasses, those are the first two classes in each of the superclasses (that are denoted in bold text) in Table B.1.

The strongly related 40-class *source* task is composed by two classes from each of the 20 CIFAR-100 superclasses, those are the third and fourth classes in each of the superclasses in Table B.1. It is considered as a strongly related task due to having the same balance of 2 classes per each of the 20 superclasses, as in the 40-class target task (although corresponding to the same superclasses, the classes themselves are different).

We also consider two weakly related 40-class source tasks, where the weaker relation to the target task is due to the different balance of classes per superclass than in the target task:

- Weakly related 40-class source task #1: From each of the first 10 superclasses of CIFAR-100 in Table B.1 we take the third, fourth and fifth classes; and from each of the other 10 superclasses we take only the third class. All these 40 classes were not included in the 40-class target task.
- Weakly related 40-class source task #2: The task has a weaker relation to the target task due to not including classes from all the 20 superclasses of CIFAR-100. Specifically, here, the 40 classes are taken only from the first 14 superclasses of CIFAR-100. From each of the first 13 superclasses of CIFAR-100 in Table B.1 we take the third, fourth and fifth classes; and from the 14th superclass we take only the third class. All these 40 classes were not included in the 40-class target task.

## C. Additional Results for Section 3

### C.1. Additional Results for Section 3.1

In Section 3.1 we have examined the evolution of the error curves in the learning of from scratch and transfer learning for a variety of *target* dataset sizes. Fig. 3 shows the results for learning a ResNet-18 for the 40-class problem. Here, Fig. C.1 shows the results for a ResNet-18 for the 10-class problem. Figs. C.2-C.3 show the results for the ViT. These additional results further support our finding that double descent is more likely to emerge in transfer learning as the target dataset is larger and noisy.

### C.2. Additional Results for Section 3.2

In Section 3.2 we have considered the effect of the *source* dataset size on the evolution of the error curves in the transfer learning of the *target* DNN. Fig. 1b shows results for the transfer learning of a ResNet-18 for the 40-class problem using a target dataset of 4000 samples. Fig. 4 shows results for the transfer learning of a ViT and ResNet-18 for the 10-class problem using a target dataset of 2560 samples. Here we provide additional results for additional target dataset sizes, see Fig. C.4 for the ResNet-18 architecture

and Fig. C.5 for the ViT architecture. Specifically, these results show that a larger source dataset can delay the arrival of the target DNN to interpolation at a later training epoch; this is observable in Figs. C.4,D.1a here, as well as in Figs. 1b, 4 in the main text.

The results in Figs. 1b, 4, C.4, C.5 consider noiseless source datasets and noisy target datasets (with 20% label noise). In Fig. C.6 we show results for settings where the target dataset is noiseless; while there are no double descent phenomena in these settings, these results demonstrate that a larger source dataset can delay the arrival to interpolation of the target DNN.

## D. Additional Results for Section 4

### D.1. Additional Results for Section 4.2

In Section 4 we examine the error evolution during training for transfer learning with various levels of freezing (i.e., different number of frozen layers). Fig. 5 shows results for ResNet-18 DNNs that are trained for the 10-class and 40-class problems. Here, we provide in Fig. D.1 the corresponding results for transfer learning of a ViT. In the examined ViT architecture we have 8 levels of freezing (recall the definitions of Frozen-0 to Frozen-7 in ViT, as described in Appendix A.2), among which in Fig. D.1 we show 6 freezing levels (Frozen-0,1,3,5,6,7) for better visibility.

As discussed in Section 4.2 and shown in Fig. 5 for ResNet-18, the results for ViT in Fig. D.1 also demonstrate that freezing layers can eliminate the double descent phenomenon when the transfer learning is with a sufficiently large target dataset.

### D.2. Additional Results for Section 4.3

The results for ViT in Fig. D.1 and the more detailed example in Fig. D.2 are also useful for observing if freezing layers can induce double descent when the transfer learning is with a small (but not too small) target dataset. Indeed, the results in Fig. D.2 for the 10-class problem with target dataset of 120 samples show that double descent emerges when several layers are frozen. But for the smaller dataset of 80 samples (that corresponds to a few-shot learning setting) in Fig. D.1, freezing does not appear to induce double descent. These results are conceptually aligned with the discussion in Section 4.3 and the results for the ResNet-18 in Fig. 5.

## E. Additional Results for Section 5

In Section 5 we examine transfer learning from source tasks at different similarity levels. In Fig. 6 we show the detailed error curves in the learning of a ResNet-18 in the Frozen-0 and Frozen-5 settings. Here, we provide in Fig. E.1 the detailed error curves in the learning of a ViT in the Frozen-0 vs. Frozen-7 settings (Fig. E.1a) and the

Superclass	Classes
aquatic mammals	<b>whale, dolphin</b> , seal, beaver, otter
fish	<b>ray, trout</b> , shark, flatfish, aquarium fish
flowers	<b>sunflower, orchid</b> , tulip, rose, poppy
food containers	<b>bottle, bowl</b> , can, plate, cup
fruit and vegetables	<b>orange, pear</b> , mushroom, apple, sweet pepper
household electrical devices	<b>clock, television</b> , lamp, keyboard, telephone
household furniture	<b>wardrobe, table</b> , chair, couch, bed
insects	<b>bee, beetle</b> , butterfly, cockroach, caterpillar
large carnivores	<b>leopard, bear</b> , lion, wolf, tiger
large man-made outdoor things	<b>skyscraper, bridge</b> , house, castle, road
large natural outdoor scenes	<b>sea, cloud</b> , mountain, forest, plain
large omnivores and herbivores	<b>cattle, elephant</b> , camel, chimpanzee, kangaroo
medium-sized mammals	<b>raccoon, possum</b> , skunk, porcupine, fox
non-insect invertebrates	<b>lobster, worm</b> , snail, crab, spider
people	<b>girl, man</b> , boy, baby, woman
reptiles	<b>crocodile, snake</b> , dinosaur, turtle, lizard
small mammals	<b>shrew, hamster</b> , rabbit, mouse, squirrel
trees	<b>maple tree, pine tree</b> , palm tree, oak tree, willow tree
vehicles 1	<b>train, bus</b> , bicycle, pickup truck, motorcycle
vehicles 2	<b>lawn mower, tractor</b> , streetcar, rocket, tank

Table B.1. The CIFAR-100 superclasses and classes. The classes in bold text compose the *target* 40-class problem in this paper, the other classes are utilized for composing the various 40-class source tasks.

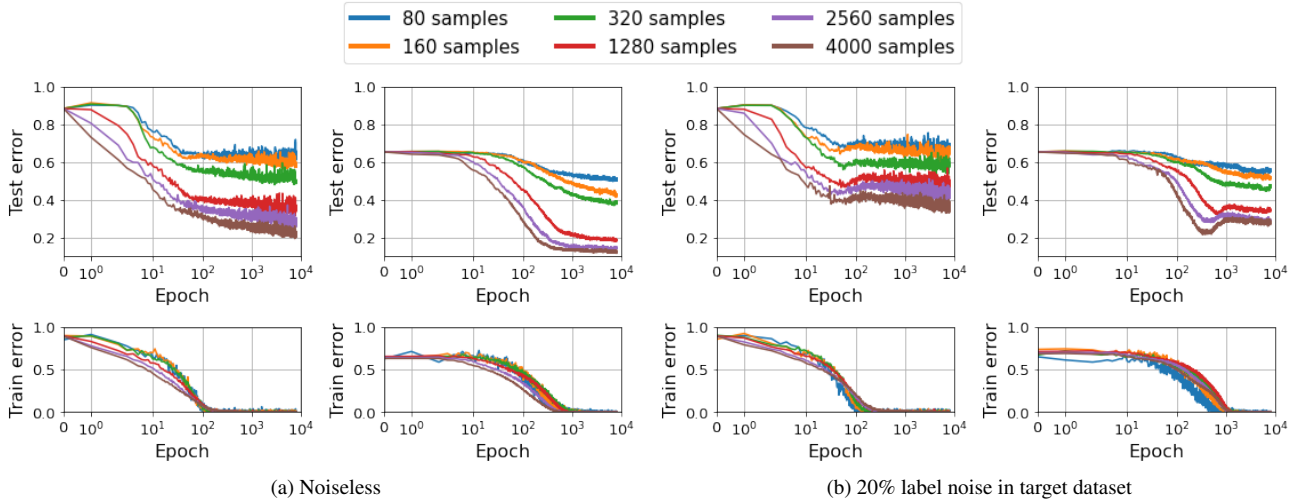


Figure C.1. Evaluation of **ResNet-18** training for the **10-class** target task. The transfer learning is from the strongly related source task of 10 classes and without frozen layers in the training. The definitions of the 10-class source and target tasks are provided in Section 2. Each curve color corresponds to another size of the target dataset. Recall that, in all of the experiments in this paper, the source dataset is noiseless. Note that significant double descent emerges in transfer learning only for noisy target datasets that are sufficiently large.

Frozen-1 vs. Frozen-6 settings (Fig. E.1b). When the target dataset is sufficiently large (see, e.g., the subfigures for the target dataset size of 2560 and 4000 samples in Fig. E.1a), the Frozen-7 setting in ViT shows overfreezing effects (i.e., transfer from a more related task is not necessarily more

beneficial).

In Fig. E.2 we provide a focused presentation of the overfreezing effect in the transfer learning of ResNet-18 with target datasets of 2560 and 4000 samples. These focused demonstrations are in addition to those already given in the

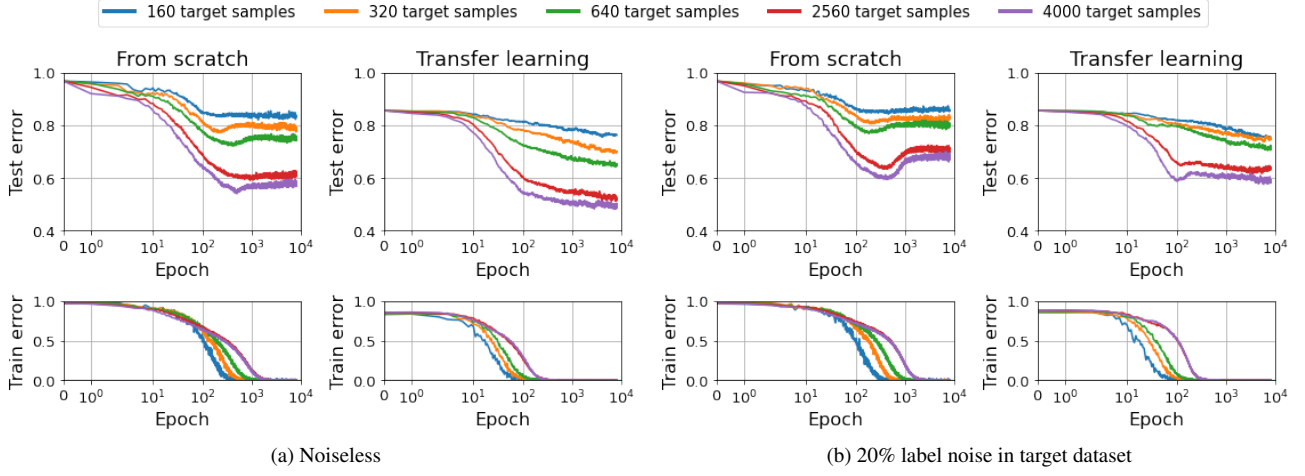


Figure C.2. Evaluation of ViT training for the **40-class** target task. The transfer learning is from the strongly related source task of 40 classes and without frozen layers in the training. The definitions of the 40-class source and target tasks are provided in Section 2. Each curve color corresponds to another size of the target dataset. Recall that, in all of the experiments in this paper, the source dataset is noiseless. Note that significant double descent emerges in transfer learning only for noisy target datasets that are sufficiently large.

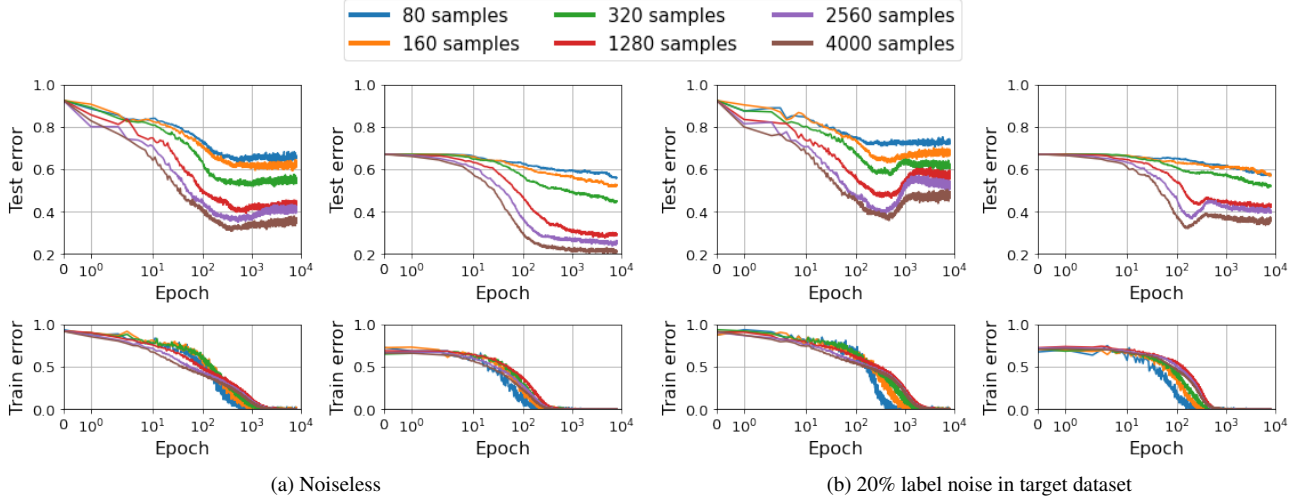


Figure C.3. Evaluation of ViT training for the **10-class** target task. The transfer learning is from the strongly related source task of 10 classes and without frozen layers in the training. The definitions of the 10-class source and target tasks are provided in Section 2. Each curve color corresponds to another size of the target dataset. Recall that, in all of the experiments in this paper, the source dataset is noiseless. Note that significant double descent emerges in transfer learning only for noisy target datasets that are sufficiently large.

main paper for ResNet-18 with target datasets of 320 and 640 samples (Figs. 2 and 7a, respectively) and for ViT with a target dataset of 4000 samples (Fig. 7b).

In Fig. E.3 we provide results for the transfer learning of a ViT with *noiseless* target dataset (i.e., in contrast to the results in Fig. E.1 for noisy target datasets). Although the noiseless target datasets induce test error curves without significant double descent shapes, we observe the effect overfreezing. Specifically, transfer from a more related task is not necessarily more beneficial, as can be observed in

the subfigures for the target dataset sizes of 2560 and 4000 samples in Fig. E.3.



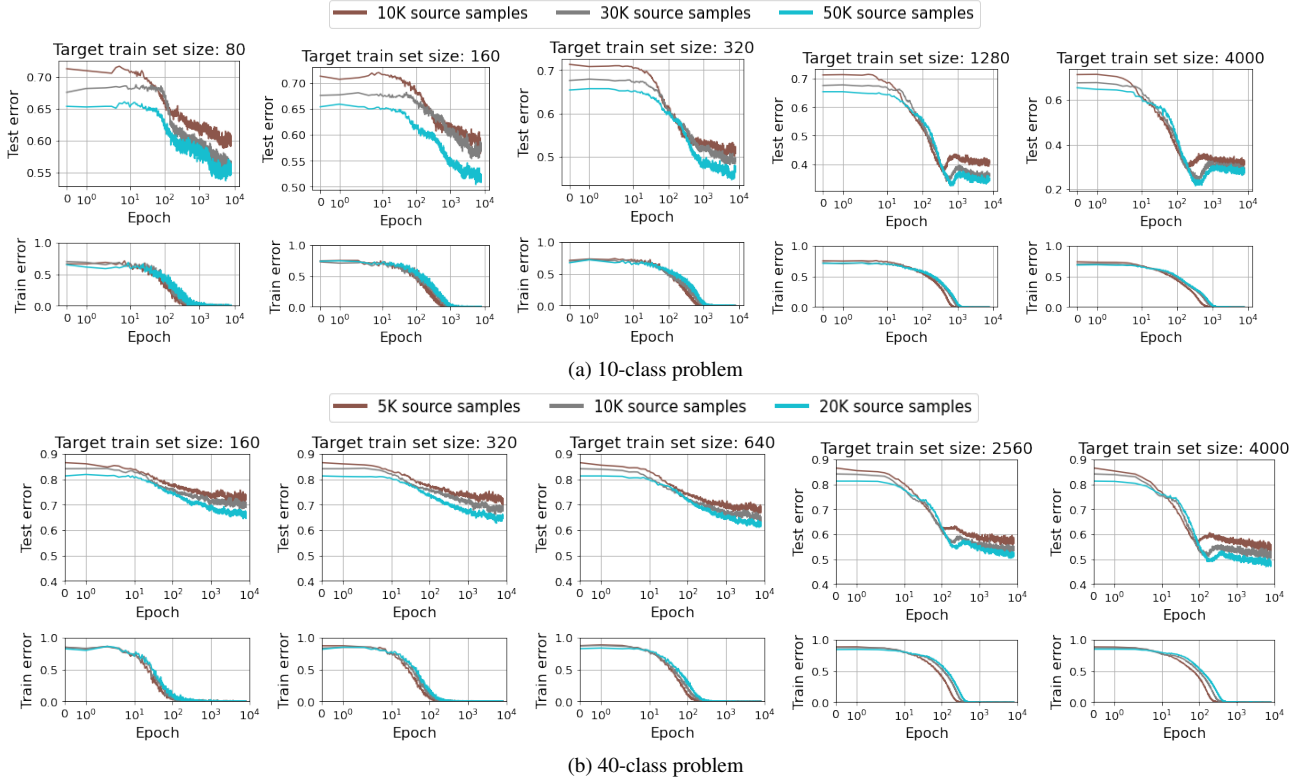


Figure C.4. The effect of the source training dataset size on the training evolution of the errors of the target model. This figure considers transfer learning of a **ResNet-18** from the strongly related problem to the target problem in the (a) 10-class setting, (b) 40-class setting. Different subfigures in the same row correspond to target training dataset of different sizes (as denoted in the subfigure titles). The source datasets are noiseless and the **target datasets are with 20% label noise**. All of these results show that the larger source dataset makes the target DNN to arrive to interpolation at a later training epoch.

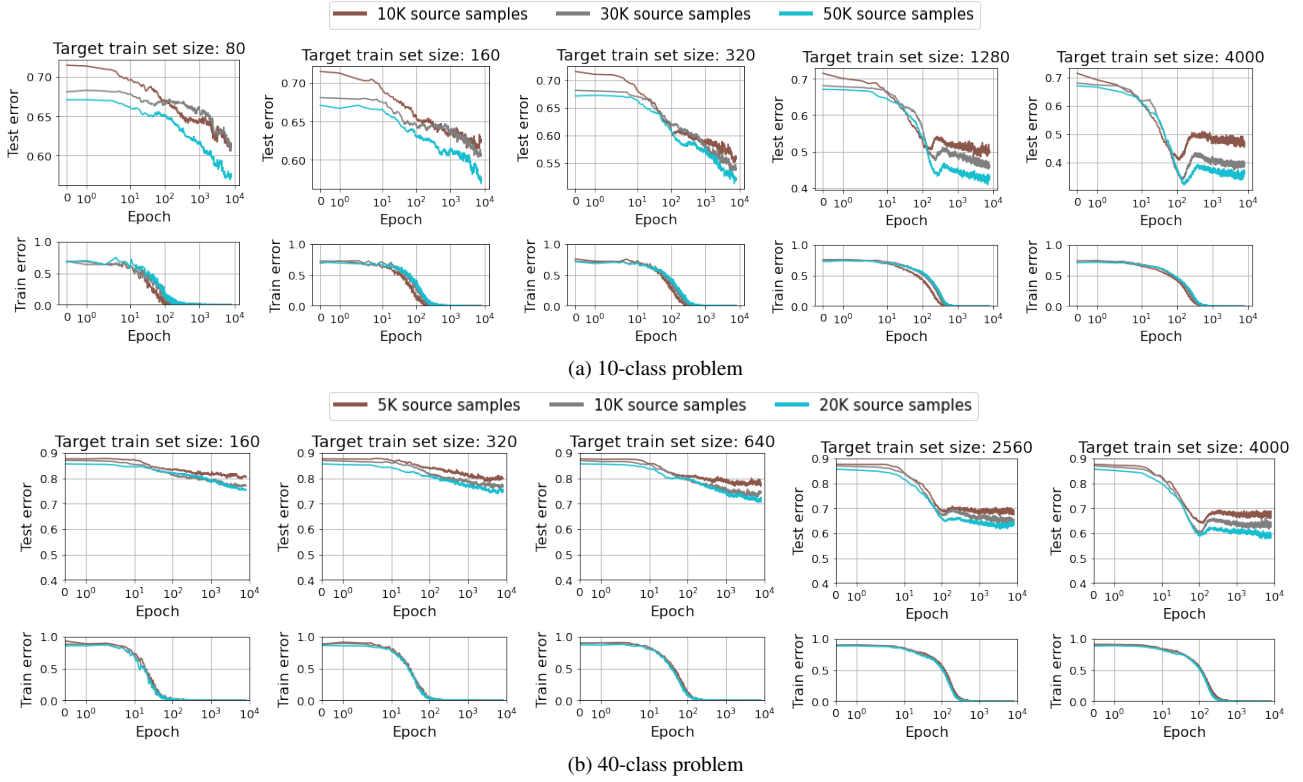


Figure C.5. The effect of the source training dataset size on the training evolution of the errors of the target model. This figure considers transfer learning of a ViT from the strongly related problem to the target problem in the (a) 10-class setting, (b) 40-class setting. Different subfigures in the same row correspond to target training dataset of different sizes (as denoted in the subfigure titles). The source datasets are noiseless and the **target datasets are with 20% label noise**. All of these results show that the larger source dataset makes the target DNN to arrive to interpolation at a later training epoch.

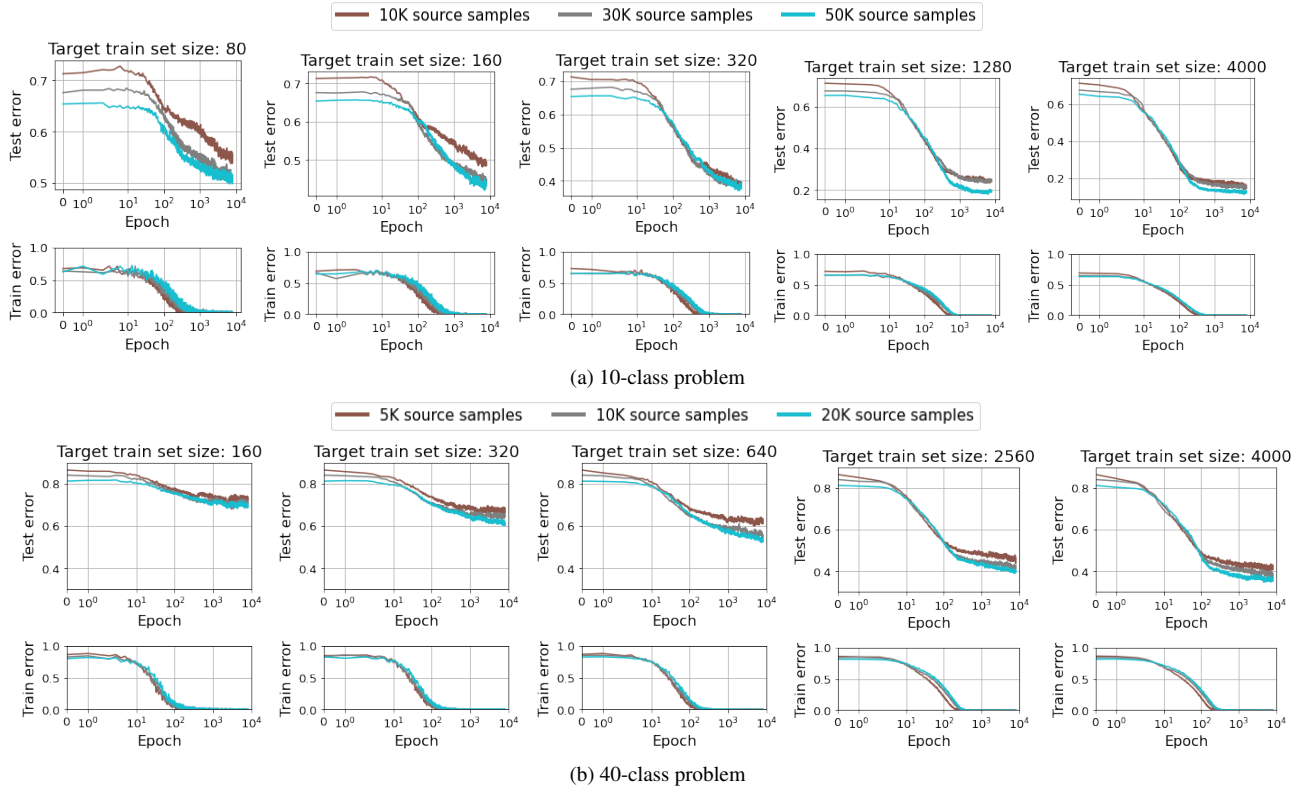


Figure C.6. The effect of the source training dataset size on the training evolution of the errors of the target model. This figure considers transfer learning of a **ResNet-18** from the strongly related problem to the target problem in the (a) 10-class setting, (b) 40-class setting. Different subfigures in the same row correspond to target training dataset of different sizes (as denoted in the subfigure titles). The source datasets are noiseless and the **target datasets are noiseless** as well. All of these results show that the larger source dataset makes the target DNN to arrive to interpolation at a later training epoch.

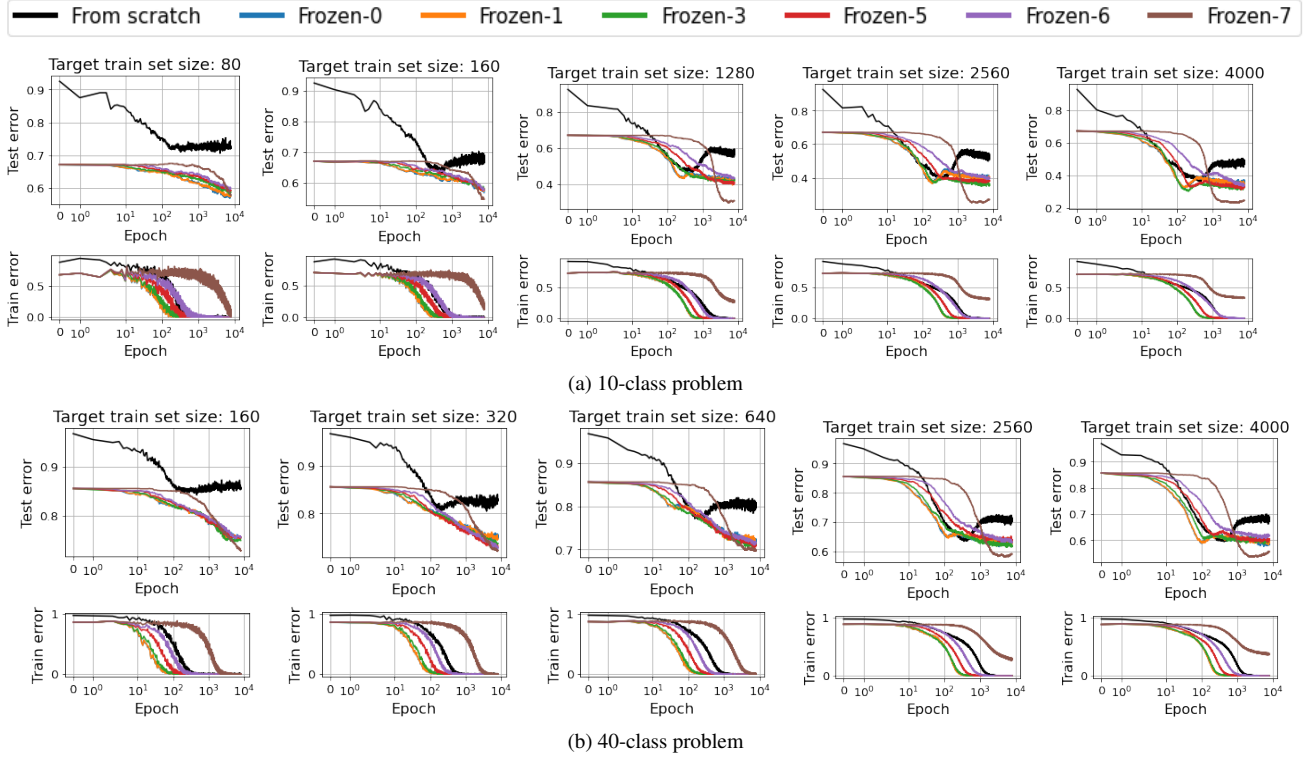


Figure D.1. Evaluation of transfer learning of a ViT at different levels of freezing and comparison to learning from scratch. The target training datasets are with 20% label noise. Subfigures on the same row differ in the training dataset size of the target task. The transfer learning are from the related source tasks. Here, for better visibility, the value range of the vertical axis may differ among the subfigures.

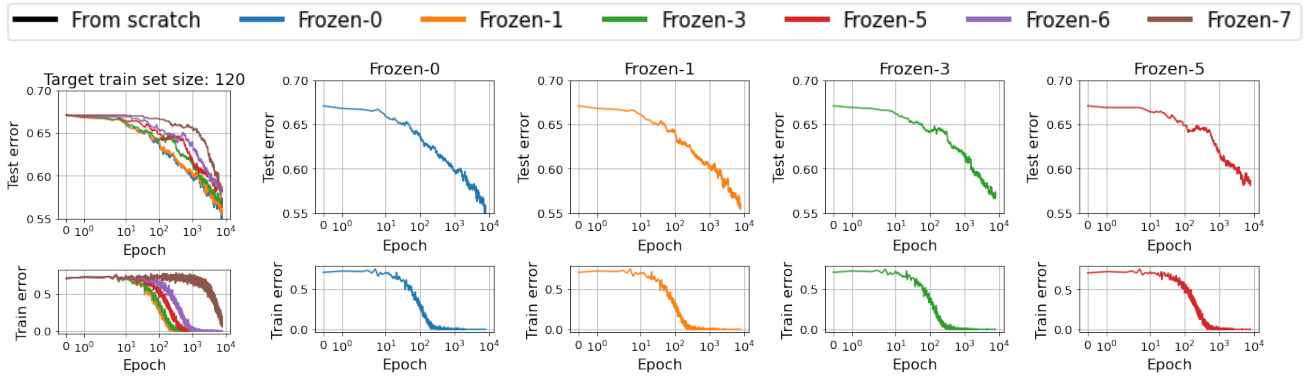
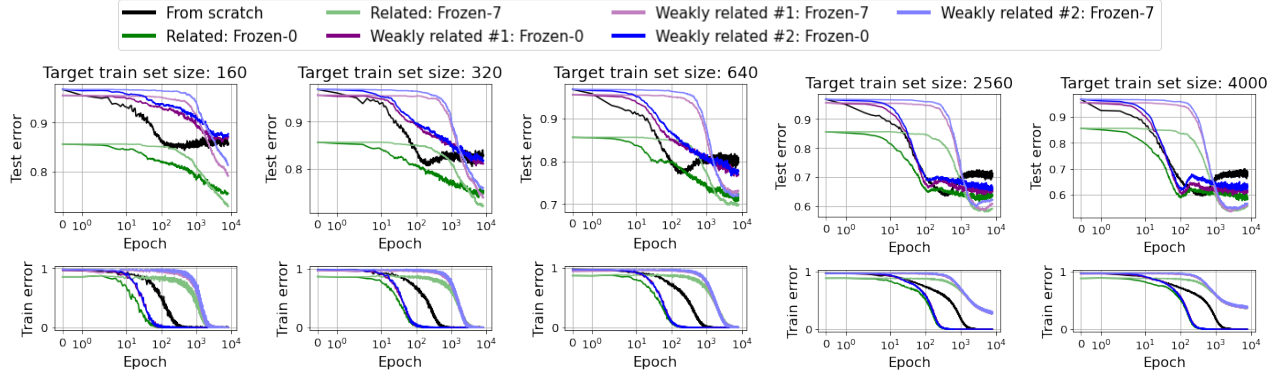
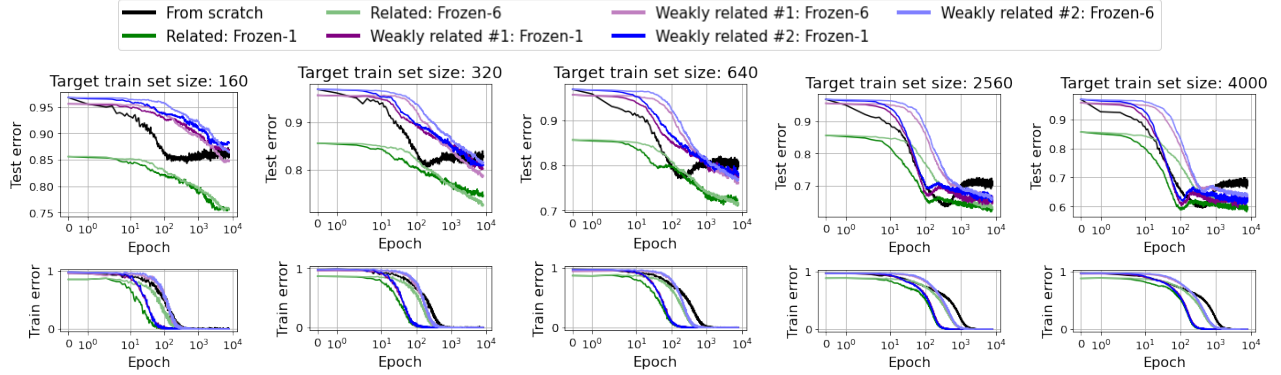


Figure D.2. Evaluation of transfer learning of a ViT at different levels of freezing. Here we consider the 10-class problem and provide evaluations for a target dataset of 120 samples (see Fig. D.1a for other dataset sizes). The left subfigure shows the error curves of 6 freezing levels, and the other subfigures show the error curves of the individual freezing levels — this visually emphasize that the double descent emerges in the Frozen-3 and Frozen-5 settings. The target training datasets are with 20% label noise. The transfer learning is from the related source task.



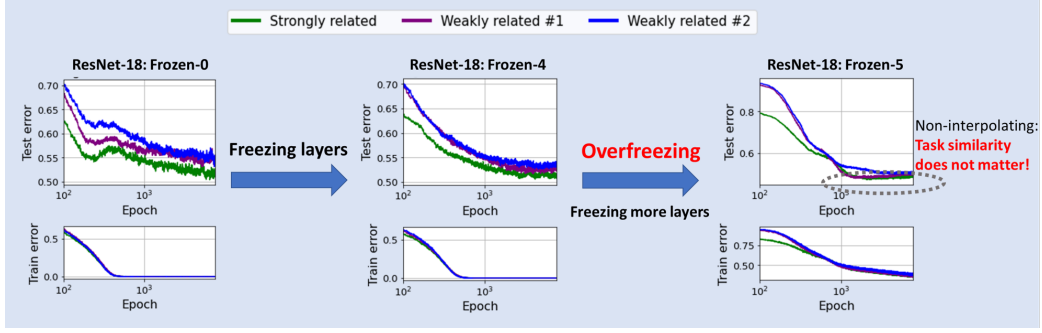
(a) Frozen-0 vs. Frozen-7



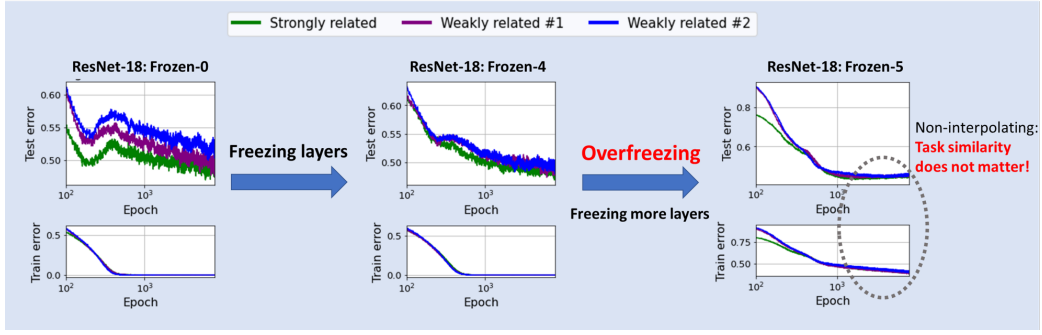
(b) Frozen-1 vs. Frozen-6

Figure E.1. Generalization performance in the 40-class target task using transfer learning of a ViT from a related and from weakly related source tasks (all are 40-class problems). **The target training datasets are with 20% label noise.** The subfigures on the same row differ in the training dataset size of the target task. Here, the value range of the vertical axis may differ among the subfigures.



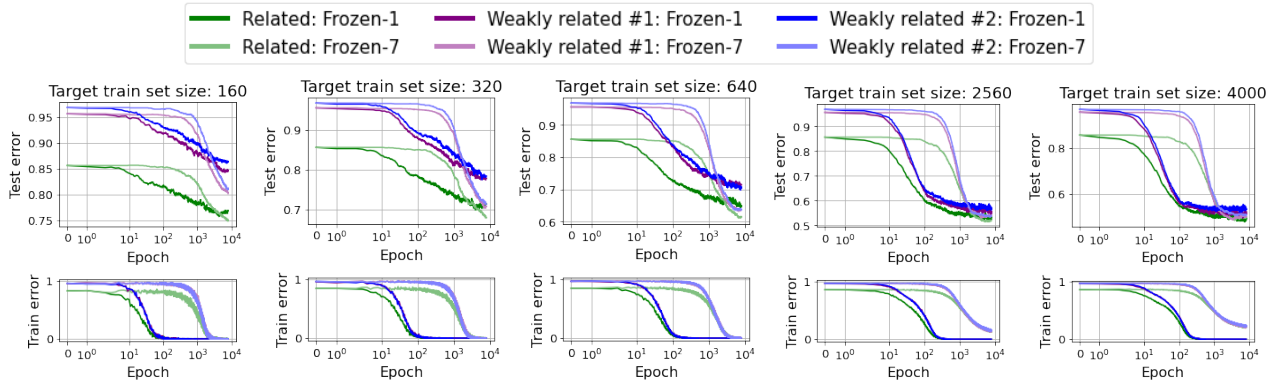


(a) Target dataset of 2560 samples



(b) Target dataset of 4000 samples

Figure E.2. Examples for *overfreezing* in transfer learning of a ResNet-18 for classification of 40 classes from CIFAR-100. The source dataset includes 20000 noiseless samples. The target datasets have 20% label noise. The classification tasks are defined in Section 2.



(a) Frozen-1 vs. Frozen-7 (noiseless target datasets)

Figure E.3. Generalization performance in the 40-class target task using transfer learning of a ViT from a related and from weakly related source tasks (all are 40-class problems). **The target training datasets are noiseless.** The subfigures on the same row differ in the training dataset size of the target task. Here, the value range of the vertical axis may differ among the subfigures.