How Far Pre-trained Models Are from Neural Collapse on the Target Dataset Informs their Transferability

Zijian Wang[1] Yadan Luo[1] Liang Zheng[2] Zi Huang[1] Mahsa Baktashmotlagh[1]
[1]The University of Queensland [2]Australian National University
{zijian.wang, y.luo, helen.huang, m.baktashmotlagh}@uq.edu.au, liang.zheng@anu.edu.au

Abstract

This paper focuses on model transferability estimation, i.e., assessing the performance of pre-trained models on a downstream task without performing fine-tuning. Motivated by neural collapse (NC) [25] that reveals particular feature geometry at the terminal stage of training, we consider model transferability as how far the target activations obtained by pre-trained models are from their hypothetical state in the terminal phase of the model fine-tuned on the target domain. We propose a metric that measures this proximity based on three phenomena of NC: within-class variability collapse, simplex encoded label interpolation geometry structure is formed, and the nearest center classifier becomes optimal on training data. Through experiments on 11 datasets, we confirm none of the three NC proxies are dispensable, which allows us to obtain very competitive transferability estimation accuracy with approximately 10× wall-clock time speed up compared to state-of-the-art approaches.

1. Introduction

The past decade has witnessed a surge in the availability of off-the-shelf pre-trained models in public repositories. They are adapted and re-tuned to facilitate new tasks, which has become a standard practice [12]. Nevertheless, this approach raises an important practical question of how to effectively select the best pre-trained model from a large model pool to be carried onto a downstream task.

While precise ranking of these pre-trained models on a new task can be obtained by a greedy approach of fine-turning each model and comparing the test accuracy, it is often prohibitive as it requires a vast amount of computing resources. Therefore, an early line of research has been developed referred to as *transferability estimation* [22, 41, 24, 29], where the aim is to design efficient methods for ranking the performances of pre-trained models on a downstream task without fine-tuning. Existing

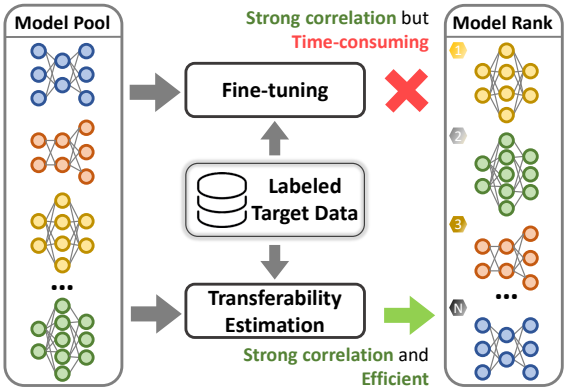


Figure 1: Illustration of the transferability estimation problem. Given a pool of pre-trained models and labeled target data, the objective of transferability estimation is to predict which models can achieve higher performance on the target data after fine-tuning. This task is much more time efficient than fine-tuning the models to obtain ground truth rankings.

approaches in transferability estimation typically consider the current status of the pre-trained models (*w.r.t* the target task) and can be generally categorized into two streams of probabilistic-based [22, 36, 41, 1] and feature distribution-based techniques [24, 29]. Approaches from the former category model the expected conditional probability of target labels given the extracted target features [41, 20] or predictions [36, 22]. The latter category models transferability as the class-wise separation of features, in which Pandey *et al.* [24] employed the Bhattacharyya coefficient to approximate the overlap between class distributions.

From a different viewpoint for transferability estimation, this paper measures the *difference between the current status and the hypothetical terminal status of pre-trained models after fine-tuning*. The latter status refers to neural collapse (NC) [25, 33] on the target domain, which commonly occurs when the training loss approaches zero. Specifically, there are three distinct characteristics of NC: 1) the within-class feature variability drops towards zero; 2) class means gradually converge to a simplex encoded label interpolation

(SELI) geometry (def. in Sec.3); and 3) the behavior of the final classifier layer is similar to the nearest centroid classifier. Meanwhile, it is empirically demonstrated in [25] that stopping training at the terminal phase achieves higher test performance compared to the models that stop at an earlier stage (*i.e.*, zero prediction error).

Motivated by these interesting findings, we formalize a transferability metric based on the observations of NC discussed above. The metric measures how close the geometry of the target features is to their hypothetical state in the terminal stage of the fine-tuned model. Specifically, we analyze the spectral property of the class-wise feature matrix to understand to what extent the current status is from zero within-class variation. Secondly, we assess the gap between the features extracted by the pre-trained model and SELI geometry with the rank of the feature matrix. Thirdly, we calculate the difference between the hypothetical classifier layer and the nearest neighbor classifier w.r.t. the empirical class-conditional distribution. Collectively, these three different measurements allow us to understand how far the pre-trained models are from NC on the target domain. On a series of standard benchmarks, we demonstrate that the metric exhibits a strong correlation with the transferability of supervised and self-supervised pre-trained models and thus is a very useful transferability estimator. Compared with the state-of-the-art, our method gives stronger correlations while being computationally efficient. We summarize our main contributions below.

- We propose the neural collapse transferability index (NCTI) for the transferability estimation. NCTI approximately measures the model *w.r.t* the three prominent characteristics of NC, which can be efficiently computed.
- Experimental results on 11 datasets demonstrate strong correlations between NCTI and the transferability of models to the target dataset, which outperforms existing transferability estimation methods.
- We give interesting insights when the NC components are computed on a subset of the pre-training dataset. Our findings suggest a potential for an unsupervised transferability estimation metric, solely based on measuring within-class variability collapse of the source features.

2. Related Work

Neural collapse. Initially observed empirically by Papyan *et. al* [25], the phenomenon of neural collapse has garnered significant research attention within a short time frame. Neural collapse shows that at the terminal phase of deep network training (*i.e., driving the training loss towards zero*), the last layer activations have the following four properties: 1) the variation of activation drop to neglectable; 2) the class mean of activation converge to simplex Equiangular Tight Frame (ETF); 3) the weight of linear classifier and class means converge to a dual-vector space; and 4) the final linear classifier behaves similar to a nearest class center classifier. Recent follow-up works focus on using the unconstrained feature model with cross-entropy [44, 9] or square loss [43, 35] training to theoretically analyze the NC phenomenon. [33, 8] extend the NC phenomenon from a class-balanced scenario to an imbalanced one. Particularly, Thrampoulidis *et. al* [33] demonstrated a more generalized geometry than simplex ETF, named simplex encoded labels interpolation (SELI). SELI shows the invariant property on balanced datasets, as well as imbalanced datasets. The authors also proved that SELI recovers the ETF geometry for the balanced dataset.

Transferability estimation. Early works [36, 1, 22] aim to assess the transferability of pre-trained models by calculating the expected conditional distribution of the target label space given the target prediction obtained from the pre-trained classifier. Specifically, [36] leverages negative conditional entropy to evaluate the amount of information shared by pre-training label space and the target task. [22] computes the empirical conditional distribution from the joint distribution of the target task and the pre-training task. Since the early methods heavily rely on the pre-trained predictor to obtain dummy source label distribution of target samples, they cannot directly evaluate the transferability of unsupervised or self-supervised models, where no classifier is available to predict dummy source label distributions.

Recent transferability estimation methods broaden the applicability by nullifying the dependency on the classifier of the pre-training task. In order to achieve this, NLEEP [20] substitutes the output layer with a Gaussian mixture model. LogME [41] formalizes the transferability estimation as the maximum label marginalized likelihood and adopts a directed graphical model to solve it. Huang *et. al* [15] proposed to approximate transferability with the mutual information between the pre-trained model extracted features and the corresponding label. On par with the metrics the probabilistic point of view, [29, 24] take the class separability into consideration for the transferability estimation metric. Particularly, Shao *et. al* [29] incorporates the self-challenge noise augmentation to encourage the classifier to discriminate the hard samples. However, none of the existing works model the transferability as the gap between the current target feature geometry and the hypothetical target feature geometry at the terminal stage of the fine-tuning.

3. Methodology

Transferability estimation is defined as ranking a set of M pre-trained models $\{\phi^m(\cdot)\}_{m=1}^M$ given a labeled target dataset $D = \{\mathbf{X}, \mathbf{Y}\} = \{(x_i, y_i)\}_{i=1}^N$ with N being the number of samples. An evaluation metric (accuracy, mAP, *etc.*) is associated with the dataset, which measures the ground-truth performance T^m of the m-th pre-

trained model $\phi^m(\cdot)$ after fine-tuning. The feature extracted by the m-th pre-trained model $\phi^m(\cdot)$ is denoted as $\mathbf{H}^m$, where $\mathbf{H}^m = \phi^m(\mathbf{X}) \in \mathbb{R}^{N\times d}$ and d is the feature dimension. With the objective of truly reflecting the ground truth model ranking, transferability estimation methods calculate a score S^m for each pre-trained model $\phi^m(\cdot)$. Ideally, the calculated scores $\{S^m\}_{m=1}^M$ are supposed to be highly correlated with the ground truth performance $\{T^m\}_{m=1}^M$, and able to determine the best pre-trained model to be fine-tuned on the target data.

The central idea of our proposed method is to employ neural collapse phenomena to assess the proximity of the current state of a pre-trained model $\phi^m(\cdot)$ to its terminal stage of fine-tuning on the target. Particularly, the proximity is measured from three dimensions, which are: 1) distance of within-class variability of features from zero; 2) simplicity for the pre-trained model to produce simplex encoded labels interpolation (SELI) geometry structure [33]; and 3) the applicability of the nearest centroid classifier. The final metric jointly considers these three terms, with each term contributing equally to the overall transferability score.

The subsequent sections elaborate on each of these three perspectives in detail. Here, we start with two definitions that will be used in the sections: the simplex encoded label (SEL) matrix, and the corresponding simplex encoded label interpolation (SELI) geometry structure.

Definition 3.1 (Simplex Encoded Label (SEL) Matrix [33]) *The SEL matrix $\hat{\mathbf{Z}} \in \mathbb{R}^{C\times N}$ is defined as,*

$$\hat{\mathbf{Z}}[c,i] = \begin{cases} 1-1/C & , c = y_i \\ -1/C & , c \neq y_i \end{cases}. \tag{1}$$

Here, $c \in [C]$ and $i \in [N]$ denote the c-th class and i-th observation, respectively. C is the number of classes. Note that $\hat{\mathbf{Z}}^\top \mathbf{1} = 0$.

From the definition of SEL matrix $\hat{\mathbf{Z}}$, we can observe that the matrix is almost a full rank matrix ($rank(\hat{\mathbf{Z}}) = min(C, N) - 1$).

Definition 3.2 (SEL Interpolation (SELI) Geometry [33]) *The embeddings $\mathbf{H} \in \mathbb{R}^{d\times N}$ and classifier weight matrices $\mathbf{W} \in \mathbb{R}^{d\times C}$ follow the simplex-encoded-labels interpolation geometry when for some scaling $\alpha > 0$:*

$$\mathbf{W}^\top\mathbf{W} = \alpha\mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top, \mathbf{H}^\top\mathbf{H} = \alpha\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top, and\, \mathbf{W}^\top\mathbf{H} = \alpha\hat{\mathbf{Z}}, \tag{2}$$

where $\hat{\mathbf{Z}} = \mathbf{V}\mathbf{\Lambda}\mathbf{U}^\top$ is the SEL matrix. $\mathbf{U}$ and $\mathbf{V}$ denote the left and right singular vector matrix of $\hat{\mathbf{Z}}$. $\mathbf{\Lambda}$ represents the diagonal singular value matrix.

3.1. Within-class Variability Collapse

As described in the neural collapse phenomenon, at the terminal stage of training, the variability of within-class features becomes neglectable and the features collapse towards the corresponding class mean. Motivated by this, we develop a score function S_{vc} to track the within-class variability of features, which indicates how close a pre-trained model is to the neural collapse state. Specifically, the within-class covariance $\mathbf{\Sigma}_c$ of c-th class is denoted as:

$$\mathbf{\Sigma}_c = \frac{1}{n_c}(\mathbf{H_c} - \mu_c)^\top(\mathbf{H_c} - \mu_c), \tag{3}$$

where $\mathbf{\Sigma}_c \in \mathbb{R}^{d\times d}$, and $\mathbf{H_c} \in \mathbb{R}^{N_c\times d}$ consists N_c features that belong to c-th class. μ_c denotes the mean of c-th class. By applying singular value decomposition of $(\mathbf{H_c} - \mu_c) = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$, the covariance matrix can be written as:

$$\mathbf{\Sigma}_c = \mathbf{V}\mathbf{\Lambda}\mathbf{U}^\top\mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top = \mathbf{V}\mathbf{\Lambda}^2\mathbf{V}. \tag{4}$$

Here, $\mathbf{U}$ and $\mathbf{V}$ represent the left and right singular vector matrices. $\mathbf{\Lambda}$ denotes the diagonal singular value matrix.

From Eq. 4, we observe that larger singular values mean higher within-class variability. Since SVD is computationally prohibitive on large matrices, it motivates us to use the nuclear norm, which calculates the sum of singular values of a matrix with lower time complexity. Meanwhile, as feature space is usually with high dimensionality, there might exist noises that are easily mitigated by fine-tuning but negatively affects variability calculation. Therefore, we compute the feature variability by substituting $\mathbf{H}_c^m$ into class-wise logits $\mathbf{Z}_c^m$. We define the within-class variability score of S_{vc} using the nuclear norm as follows:

$$S_{vc}^m(\mathbf{H}^m) = -\sum_{c=1}^{C} ||\mathbf{Z}_c^m||_*, \tag{5}$$

where $\mathbf{Z}_c^m$ denotes the logits of c-th class extracted by the m-th model. We defer the formulation of $z_{i,c}$ to section 3.3. Since S_{vc}^m directly reflects the within-class variability, when a pre-trained model $\phi^m(\cdot)$ is close to the terminal stage of the fine-tuning, a larger S_{vc}^m should be achieved.

3.2. Simplex Encoded Labels Interpolation (SELI)

Thrampoulidis *et al.* [33] proposed SELI as a generalized geometry structure version of the simplex equiangular tight frame (ETF) that is defined in the neural collapse phenomenon. While the simplex ETF geometry observed in neural collapse applies only to balanced datasets, SELI can be established in both balanced and imbalanced datasets at the neural collapse state. As such, we propose a score function S_{seli}^m to assess the SELI geometry structure of the target features extracted from the m-th pre-training model. In the

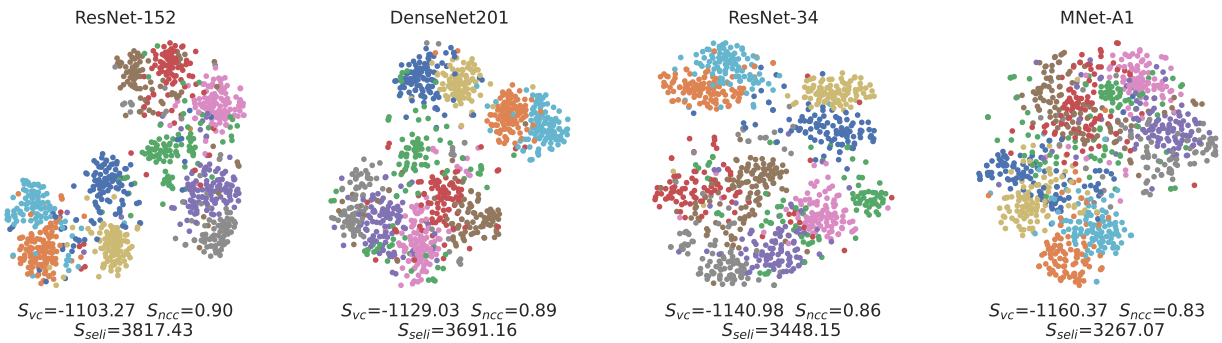


Figure 2: TSNE visualization of feature distributions produced by four pre-trained models with CIFAR-10 as the downstream task. Different colors mean different classes in CIFAR-10. From left to right, target recognition accuracy (or transferability) decreases. At the same time, the proposed three scores, i.e., S_{vc}, S_{seli}, and S_{ncc} also decrease.

rest of this section, we first review the role of SELI geometry in cross-entropy minimization, followed by discussing our SELI-based metric.

Under the unconstrained feature (UF) model [44], Thrampoulidis *et al.* shows that minimizing the cross-entropy loss converges to a KKT point of the SVM classifier, where the global minimizer of this optimization problem is the SEL matrix.

Theorem 1 *(Structure of the UF-SVM minimizers) [33] Let* $(\hat{\mathbf{W}}, \hat{\mathbf{H}})$ *be any solution and* p^* *be the optimal cost of the following unconstrained feature SVM (UF-SVM) classifier:*

$$(\hat{\mathbf{W}}, \hat{\mathbf{H}}) \in \arg\min_{\mathbf{W},\mathbf{H}} ||\mathbf{W}||_F^2/2 + ||\mathbf{H}||_F^2/2,$$
$$s.t.\ (w_{y_i} - w_c)^T h_i \geq 1, i \in [n], c \neq y_i.$$

Since $\hat{\mathbf{Z}} = \hat{\mathbf{W}}^\top \hat{\mathbf{H}}$ *holds for optimal logits, the optimal cost can be obtained as* $p^* = ||\hat{\mathbf{Z}}||_* = ||\hat{\mathbf{H}}||_F^2 = ||\hat{\mathbf{W}}||_F^2$.

A straightforward way to assess the SELI geometry structure of the target features is to compute the difference between logits $\mathbf{Z}^m$ extracted from m-th pre-trained model and the optimal logits $\hat{\mathbf{Z}}$. However, since the source and target data might not share similar classes, we cannot directly obtain the logits $\mathbf{Z}^m$ without the time-consuming fine-tuning process on the target dataset. Therefore, we extract features $\mathbf{H}^m$ of the model and measure its discrepancy to form the SELI structure.

Specifically, we use the proven fact that, at the terminal stage of the fine-tuning where the model achieves the optimal cost p^*, we have $||\hat{\mathbf{Z}}||_* = ||\hat{\mathbf{H}}||_F^2$ [33]. Therefore, we approximate the complexity of achieving the optimal logits $\hat{\mathbf{Z}}$ via features $\mathbf{H}^m$ extracted from the pre-trained model $\phi_m(\mathbf{X})$ as:

$$S^m_{seli}(\mathbf{H}^m) = ||\mathbf{H}^m||_*. \tag{6}$$

Since the nuclear norm is the upper bound of $||\cdot||_F^2$ and is the convex envelope for $rank()$, a higher S^m_{seli} indicates a higher rank of the feature matrix $\mathbf{H}_m$, which makes $\mathbf{Z}$ closer to a full rank matrix.

3.3. Simplicity to Nearest Centroid Classifier

Based on the neural collapse phenomenon shown in [25], at the terminal stage of training, the linear classifier behaves similarly to the nearest centroid classifier. Therefore, a score function S^m_{ncc} is proposed to measure the applicability of the nearest centroid classifier on the m-th pre-trained model. We start with defining the nearest centroid classifier as [28]:

$$\hat{y} = \arg\max_{c\in[C]} \cos(\mu_c, x), \tag{7}$$

where $\hat{y}$ is the assigned class label to a sample x, and cos indicates the cosine similarity.

In the neural collapse state, the nearest centroid classifier can achieve the optimal classification performance on the training set. However, in transferability estimation, where there is no actual fine-tuning on the target dataset, directly comparing the performance of applying the nearest centroid classifier to the extracted feature with that of the optimal solution can lead to inaccurate estimation results. To mitigate this issue, we consider two relaxations to the nearest centroid classifier: Instead of using cosine similarity which only involves the mean of the classes, we adopt Mahalanobis distance as the nearest centroid classifier that further involves the correlation between the classes. Moreover, we leverage the expected conditional distribution of labels given input to replace the hard label. Under a mild assumption, the class distribution follows a multivariate Gaussian distribution, thus, we can obtain the following posterior by applying Bayes' Rules:

$$\log P(y=c|h) = \frac{1}{2}(h_i-\mu_c)^\top \mathbf{\Sigma}(h_j-\mu_c) + \log P(y=c). \tag{8}$$

Here, $\log P(y=c)$ denotes the prior probability of class c, which can be computed as N_c/N. N_c represents the number of samples belonging to class c.

Compared with the discrete output of *argmax* function in the nearest centroid classifier formulation, we adopt the

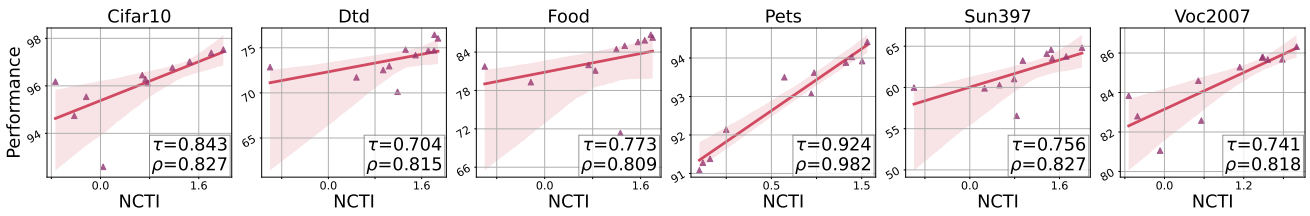


Figure 3: Strong correlation between the proposed NCTI score and model recognition performance (%) on the target dataset after fine-tuning. Each marker represents a different pre-trained model. The target domains are (from left to right): Cifar-100 [17], DTD [6], Food101 [2], Pets [26], SUN397 [39], and VOC2007 [7]. We show Spearman's coefficient ρ and weighted Kendall's τ here. More correlation results are presented in supplementary materials.

softmax function as it preserves the relative order information of the prediction. Therefore, even if the corresponding class of a sample is not assigned with the highest probability, the probability value still reflects the distance of the sample to that class. The c-th class normalized posterior probability of i-th sample can be calculated as follows:

$$z_{i,c}^m = \frac{\exp(\log P(y=c|h_i^m))}{\sum_{k=1}^{C}\exp(\log P(y=k|h_i^m))}, \tag{9}$$

As such, we have the score S_{ncc}^m to describe the discrepancy between the current state of the m-th pre-trained model $\phi_m(X)$ and the optimal nearest centroid classifier as follows:

$$S_{ncc}^m(\mathbf{H}^m) = \frac{1}{N}\sum_{i=1}^{N} z_i^m y_i, \tag{10}$$

where y_i denotes the corresponding one-hot ground truth label of x_i and z_i^m is the C dimensional logits of x_i, predicted by m-th model. A greater S_{ncc}^m indicates a smaller discrepancy towards an optimal nearest centroid classifier, which translates to greater transferability on the target dataset.

3.4. Overall Score Function

In this work, we model the transferability of a pre-trained model as how similar the current state of the network is to its terminal phase of fine-tuning. Specifically, we expect the terminal phase of fine-tuning to hold the so-called three properties of within-class variability, SELI geometry, and the nearest centroid classifier. Correspondingly, we design three score functions, which are $S_{vc}(\cdot)$, $S_{seli}(\cdot)$, and $S_{ncc}(\cdot)$ to measure the proximity of the state of the pre-trained network applied on the target data versus its optimal state at the terminal phase. However, since the values of score functions have different scales, summing up the scores may hurt the transferability estimation performance (*e.g.* $S_{seli}(\cdot)$) can numerically dominate over $S_{ncc}(\cdot)$). Instead of having manually defined hyper-parameters to balance the contribution of each score, we propose to normalize each score into a unit range (0 to 1) as follows:

$$S_{seli}^m \leftarrow \frac{S_{seli}^m - \min(S_{seli})}{\max(S_{seli}) - \min(S_{seli})}. \tag{11}$$

Similarly, we can have the normalized score $S_{vc}(H^m)$ and $S_{ncc}(H^m)$ for the m-th pretrained model. By summing the normalized scores, we have the final transferability estimation metric:

$$S_{total}^m = S_{vc}^m(\mathbf{H}^m) + S_{seli}^m(\mathbf{H}^m) + S_{ncc}^m(\mathbf{H}^m), \tag{12}$$

where the m-th pre-trained model with a higher overall score S_{total}^m indicates a better transferability in the model pool for the target dataset D.

4. Experiment

4.1. Datasets and Evaluation Metrics

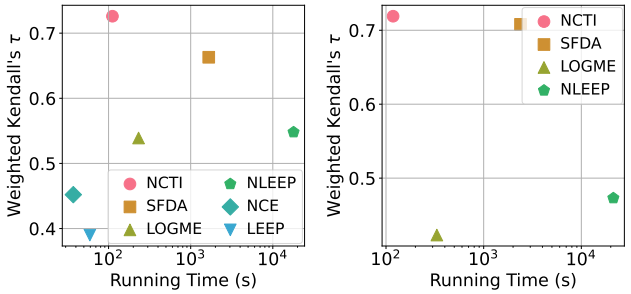
Datasets. We consider a wide range of classification benchmark datasets, including five fine-grained classifications (FGVC Aircraft [21], Standford Cars [16], Food-101 [2], Oxford-IIIT Pets [26], Oxford-102 Flowers [23]), four coarse-grained classifications (Caltech101 [18], CIFAR-10 [17], CIFAR-100 [17], VOC2007 [7]), one scene classification (SUN397 [39]), and one texture classification (DTD [6]). In total, we adopt 11 benchmark datasets that are broadly used in transfer learning. A detailed description of each dataset can be found in the supplementary material. All models have undergone pre-training on the ImageNet dataset, and the fine-tuned performance of the model on target datasets is obtained from [29]. Without affecting the fine-tuning performance, we note that only the training and validation splits of the dataset are used for transferability estimation, and we hold the test split out of the estimation process.

Correlation measurement. Following the previous work [41, 24, 29], we adopt weighted Kendall's τ [37] to measure the correlation between the estimated model ranking and the ground-truth performance ranking. Each pairwise comparison between items in two rankings is assigned a weight based on the distance between their ranks, whereas top-ranked items will be assigned with higher weight.

Implementation details. In order to replicate the fine-tuning process, our proposed method aims to find a projection matrix W that maximizes the discriminative power of the extracted features $\phi^m(X)$ from the m-th pre-trained

Table 1: Method comparison of their correlation strength with target accuracy for **supervised** models. Weighted Kendall's τ on 11 target test sets and their average are shown. For each column, the best, and second-best results are in bold, and underlined, respectively. Our method achieves the best overall weighted Kendall's τ.

	Aircraft	Caltech	Cars	Cifar10	Cifar100	DTD	Flowers	Food	Pets	SUN	VOC	Average
NCE [36]	-0.161	0.465	**0.685**	0.709	0.723	0.302	-0.482	0.627	0.772	0.760	0.571	0.452
LEEP [22]	-0.277	0.605	0.367	0.824	0.677	0.486	-0.291	0.434	0.389	0.658	0.413	0.390
LogME [41]	0.439	0.463	0.605	0.852	0.725	0.700	0.147	0.385	0.411	0.511	0.695	0.539
NLEEP [20]	-0.531	0.614	0.489	0.825	0.731	**0.820**	0.054	0.529	**0.955**	**0.848**	0.699	0.548
SFDA [29]	**0.614**	**0.696**	0.518	**0.949**	0.866	0.575	0.514	**0.815**	0.522	0.558	0.671	0.663
NCTI	0.496	0.679	0.647	0.843	**0.879**	0.704	**0.541**	0.773	0.924	0.756	**0.741**	**0.726**



(a) Supervised models (b) Self-supervised models

Figure 4: Method comparison *w.r.t* running time (seconds) vs. average Weighted Kendall's τ on 11 datasets on (a) supervised and (b) self-supervised models. Our method obtains the highest ranking correlation within a small time budget.

model. To achieve this, we employ linear discriminant analysis (LDA) to maximize the separation between the class means of the projected data. However, LDA can be prone to overfitting when applied to high-dimensional feature spaces or when the covariance matrix is singular [40, 42]. To mitigate the overfitting issue on high-dimensional feature space, we follow the previous transferability estimation methods [20, 24] which incorporate principal component analysis (PCA) into the framework to project the original features onto a lower-dimensional subspace. We further address the singularity issue by shrinking the covariance matrix toward a full-rank identity matrix as follows:

$$\mathbf{\Sigma}'_w = (1-\alpha)\mathbf{\Sigma}_w + \alpha\mathbf{I}, \text{where } \alpha = exp(\sum_{i}^{d^*} \lambda_i), \quad (13)$$

Here, λ_i denotes the i-th largest eigenvalue of the PCA projected feature matrix $PCA(\phi_m(X))$. d^* indicates the number of eigenvalues to be considered in the calculation. We set the dimension of the PCA projected subspace as 64 and set d^* as 32 throughout all the experiments. α controls the trade-off between the covariance and the identity matrices.

4.2. Evaluation on Supervised Models

Overview. To initiate our evaluation of transferability estimation metrics for ranking supervised models, we assess the performance of these metrics on 11 widely adopted models, which are ResNet-34 [13], ResNet-50, ResNet-101, ResNet-151, DenseNet-121 [14], DenseNet-169, DenseNet-201, MNet-A1 [32], MobileNet-v2 [27], GoogleNet [30], and Inception-v3 [31].

Performance comparison. We compare our proposed NCTI with existing transferability estimation metrics, including NCE [36], LEEP [22], LogME [41], NLEEP [20], and SFDA [29]. The experimental results are shown in Tab. 1. From the table, we can observe that empirical conditional probability-based methods (*i.e,* NCE, LEEP, and LogME) generally achieve lower ranking correlations than our method and SFDA, indicating the importance of modeling the feature space in transferability estimation. NCTI achieves the best or the second best average weighted Kendall's ranking correlation on nine datasets and obtains a 6.9% averaged performance gain compared to the second-best method SFDA. We argue that while the neural collapse explicitly reflects the maximum class separation, a feature space with a high class separation score can not necessarily leads to the neural collapse phenomenon. Therefore, our method can achieve superior performance than class separation-based methods.

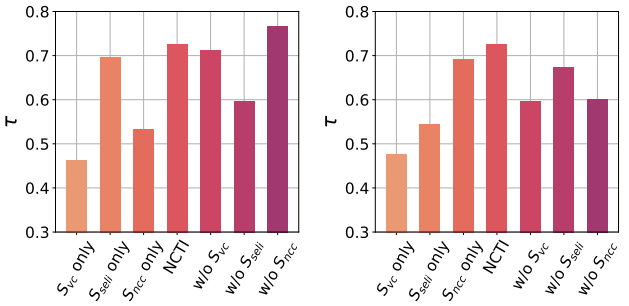
4.3. Evaluation on Self-supervised Models

Overview. We further evaluate the effectiveness of NCTI in assessing the transferability of models pre-trained by self-supervised learning (SSL). We construct a pool with 10 SSL pre-trained models, including MoCo-v1 [11], MoCo-v2 [5], PCL-v2 [19], SELA-V2, Deepcluster-v2 [4], BYOL [10], Infomin [34], SWAV [3], and Insdis [38].

Performance comparison. Since NCE and LEEP require the classifier on the pre-training task, it is not directly applicable to self-supervised model ranking tasks. Therefore, to evaluate the ranking performance of the self-supervised models, we compare our method with NLEEP, LogME, and SFDA in Tab. 2. Similarly, we observe that the average ranking correlation of SFDA and the proposed

Table 2: Method comparison of their correlation strength with target accuracy for **self-supervised** models. All the settings and notations are the same as Table 1. Our method achieves the best overall weighted Kendall's τ.

	Aircraft	Caltech	Cars	Cifar10	Cifar100	DTD	Flowers	Food	Pets	SUN	VOC	Average
NLEEP [20]	-0.286	0.662	0.595	0.108	0.374	0.779	0.598	0.716	**0.864**	**0.880**	-0.091	0.473
LogME [41]	0.021	0.075	0.627	0.417	0.146	0.743	0.763	0.686	0.738	0.260	0.181	0.423
SFDA [29]	**0.167**	0.674	0.683	**0.846**	0.789	**0.882**	**0.897**	0.837	0.564	0.831	**0.621**	0.708
NCTI	0.036	**0.811**	**0.796**	0.758	**0.811**	0.796	0.762	**0.945**	0.805	0.774	0.606	**0.719**



(a) Supervised models (b) Self-supervised models

Figure 5: Effectiveness of each individual component in NCTI. We use the three terms individually or remove them one at a time from the full system. Results on (a) supervised and (b) self-supervised models are shown.

method can achieve superior results than that of the conditional probability-based method. NCTI surpasses the second-best method by 1.5% on average and scores the highest overall weighted Kendall's τ, which verifies the effectiveness of the proposed method.

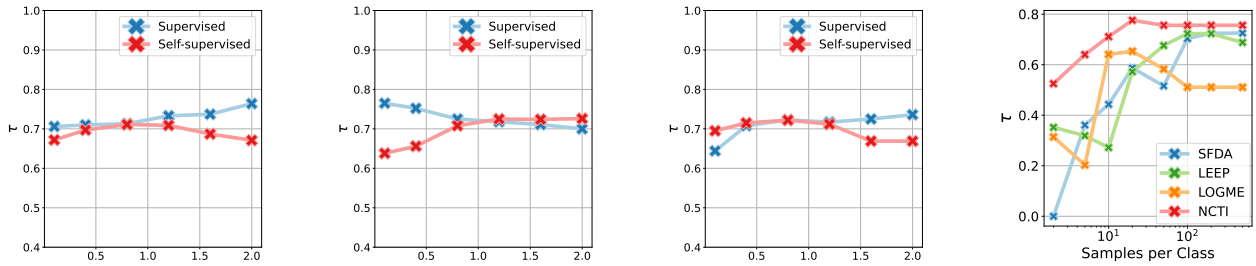
4.4. Further Analysis

Tranferability estimation with fewer target data. We test a practical but challenging transferability estimation scenario, where only limited target data are available for the estimation. We randomly sample 2 to 500 images per class from the full SUN397 and examine how limited data affects the performance of different methods. As we can see from Fig. 6(d), although fewer data generally trigger performance degradation for transferability estimation methods, our method is able to achieve a larger performance gain under the low shot scheme than using the full dataset. Particularly, we highlight that our method can achieve SOTA performance on SUN397 with only 10 samples per class, which indicates a strong potential for practical applications.

Computational efficiency comparison. We compare the time efficiency versus ranking correlation of different transferability estimation metrics in Fig. 4. The x-axis is the total time cost for estimating transferability over 11 datasets, while the y-axis indicates the average ranking correlation on 11 datasets. We adopt a logarithmic scale for the x-axis. The figure shows that within a similar time budget, NCTI can achieve significantly higher ranking correlations compared with empirical conditional probability-based methods, such as NCE, LEEP, and LogME. SFDA can achieve a comparable ranking correlation with NCTI on self-supervised model ranking tasks. However, it consumes more than $10\times$ more time to estimate the transferability, showing that it is less efficient than our proposed metric. Meanwhile, we can observe from the figure that our method scores the highest ranking correlation to the running time ratio under both settings, emphasizing the effectiveness and efficiency of our method.

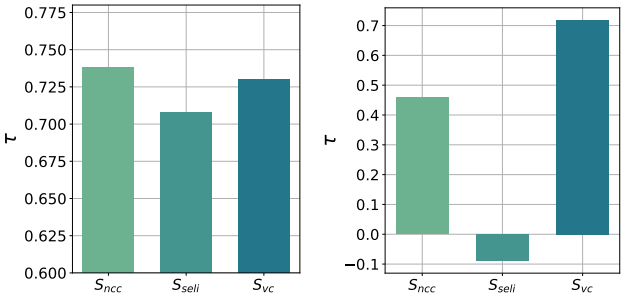
Ablation study. The NCTI metric proposed in this work is composed of three terms, each term is designed to describe a specific aspect of the gap between the current feature geometry and the geometry of neural collapse. In order to illustrate the impact of each individual term in NCTI, we conduct ablation studies to show how each individual term contributes to the overall metric. The ablation studies are conducted on both supervised and self-supervision scenarios, and the results are shown in Fig. 5. Among these three individual terms of NCTI, although S_{vc} can attain a positive averaged ranking correlation, the contribution towards the full metric is less than the other two terms. S_{seli} delivers the highest average ranking correlation on supervised model estimation, while S_{ncc} attains the best-averaged performance on self-supervised model estimation. We infer this is due to the difference in the supervision signal. As the label information is not available for self-supervised training, the discriminative power of the pre-trained feature is more critical for the downstream task than that in supervised models. We also validate the effectiveness of two-term combinations. As we can see from the figure, the combination between S_{seli} and S_{vc} always achieves higher correlations than its individual component, confirming the terms are complementary. Although NCTI obtains a slightly lower correlation than w/o S_{ncc} on the supervised models, it achieves an absolute 4.2% higher averaged correlation of both tasks, which verifies the importance of S_{ncc}.

Hyperparameter analysis. NCTI simplifies the hyperparameter tuning process by assigning equal weights to each term. In order to validate this weight assignment, we investigate the impact of weight changes on the ranking cor-



(a) Weight of S_{vc} (b) Weight of S_{ncc} (c) Weight of S_{seli} (d) SUN397

Figure 6: (a)-(c) Hyperparameter analysis of three components in NCTI. We report correlation strength τ computed on 11 downstream datasets against weight values. Impact on both supervised and self-supervised pre-trained models are drawn. We study each weight by fixing the rest as 1. In this paper, by default, we set the weight to 1 for all components. (d) Influence of target data size for estimation on the correlation strength. We vary the data size from 2 to 500 samples per class and observe that our method can achieve a larger performance margin compared with existing methods under the low-shot scenario.



(a) Supervised models (b) Self-supervised models

Figure 7: Effectiveness of the three components computed on **a subset of the pre-training dataset**. Average correlation strength (τ) on 11 target datasets reported. We assess (a) supervised models, and (b) self-supervised models.

relation. Specifically, we vary the weight of the term of interest within the range of $\{0.1, 0.4, 0.8, 1.2, 1.6, 2\}$ while keeping the weight of the remaining terms fixed at 1. The experimental results of the hyperparameter analysis are reported in Fig.6 (a)-(c). Our findings indicate that NCTI achieves high averaged correlations ($\tau \geq 0.7$) with the most weight configurations. While we observe different monotonicity between the correlations of supervised and self-supervised tasks, the results demonstrate that harmonic points (i.e., high mean and low variance) are consistently achieved when all weights are close to 1, thereby confirming the appropriateness of the equal weight setting.

Analysis of NCTI computed using source features. The extensive experiments conducted so far have demonstrated the effectiveness of each individual term in the proposed NCTI metric for transferability estimation. All the previous experiments have been conducted by using the target features extracted from the pre-trained models. It remains important to investigate what the properties of neural collapse on the source (i.e., pre-training) features tell us about the transferability of a model to a downstream task. To this end, we measure each individual term in NCTI on a subset of the source dataset and report the ranking correlation of the supervised model estimation task in Fig.7(a) and (b). On average, the transferability of a supervised model to a downstream task is consistently positively related to $-S_{vc}$ and S_{ncc} on the source dataset on both supervised and self-supervised, which scores 0.725 and 0.598 weighted Kendall τ, respectively. While the figure shows S_{seli} can achieve a positive correlation on supervised tasks, no significant correlation on S_{seli} is observed in the supervised transfer estimation task. Note that, $-S_{vc}$ achieves the highest ranking correlation on average, which shows that a pre-trained model with higher within-class variability can better transfer to a new task. This indicates that a fully unsupervised metric can be designed for transferability estimation which is solely based on measuring the class-wise compactness of the source features.

5. Conclusion

How to efficiently choose the optimal pre-trained model from a vast collection of models to use in a downstream task? To rank models based on their transferability, we propose a metric named Neural Collapse informed Transferability Index (NCTI). Our method models model transferability as the gap between the current feature geometry and the geometry at the terminal stage (*i.e.*, neural collapse) after hypothetical fine-tuning on the downstream task. Specifically, we model the gap from three perspectives, including the formation of *SELI* geometry structure, the within-class variability, and the applicability of the nearest center classifier. We show that our method is light to compute and that the ranking of model transferability has a very strong correlation with the ground truth ranking and compares favorably against the state-of-the-art methods.

Acknowledgement This work was partially supported by ARC DP 230101196

References

[1] Yajie Bao, Yang Li, Shao-Lun Huang, Lin Zhang, Lizhong Zheng, Amir Zamir, and Leonidas Guibas. An information-theoretic approach to transferability in task transfer learning. In *2019 IEEE international conference on image processing (ICIP)*, pages 2309–2313. IEEE, 2019. 1, 2

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer, 2014. 5

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018. 6

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020. 6

[5] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 6

[6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014. 5

[7] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111:98–136, 2015. 5

[8] Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43):e2103091118, 2021. 2

[9] Florian Graf, Christoph Hofer, Marc Niethammer, and Roland Kwitt. Dissecting supervised contrastive learning. In *International Conference on Machine Learning*, pages 3821–3830. PMLR, 2021. 2

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 6

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 6

[12] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4918–4927, 2019. 1

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[14] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 6

[15] Long-Kai Huang, Junzhou Huang, Yu Rong, Qiang Yang, and Ying Wei. Frustratingly easy transferability estimation. In *International Conference on Machine Learning*, pages 9201–9225. PMLR, 2022. 2

[16] Jonathan Krause, Jia Deng, Michael Stark, and Li Fei-Fei. Collecting a large-scale dataset of fine-grained cars. 2013. 5

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[18] Fei-Fei Li, Marco Andreeto, Marc'Aurelio Ranzato, and Pietro Perona. Caltech 101, Apr 2022. 5

[19] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020. 6

[20] Yandong Li, Xuhui Jia, Ruoxin Sang, Yukun Zhu, Bradley Green, Liqiang Wang, and Boqing Gong. Ranking neural checkpoints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2663–2673, 2021. 1, 2, 6, 7

[21] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5

[22] Cuong Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. Leep: A new measure to evaluate transferability of learned representations. In *International Conference on Machine Learning*, pages 7294–7305. PMLR, 2020. 1, 2, 6

[23] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 5

[24] Michal Pándy, Andrea Agostinelli, Jasper Uijlings, Vittorio Ferrari, and Thomas Mensink. Transferability estimation using bhattacharyya class separability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9172–9182, 2022. 1, 2, 5, 6

[25] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020. 1, 2, 4

[26] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 5

[27] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 6

[28] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008. 4

[29] Wenqi Shao, Xun Zhao, Yixiao Ge, Zhaoyang Zhang, Lei Yang, Xiaogang Wang, Ying Shan, and Ping Luo. Not all models are equal: Predicting model transferability in a self-challenging fisher space. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIV*, pages 286–302. Springer, 2022. 1, 2, 5, 6, 7

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 6

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 6

[32] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2820–2828, 2019. 6

[33] Christos Thrampoulidis, Ganesh R Kini, Vala Vakilian, and Tina Behnia. Imbalance trouble: Revisiting neural-collapse geometry. *arXiv preprint arXiv:2208.05512*, 2022. 1, 2, 3, 4

[34] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020. 6

[35] Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning*, pages 21478–21505. PMLR, 2022. 2

[36] Anh T Tran, Cuong V Nguyen, and Tal Hassner. Transferability and hardness of supervised classification tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1395–1405, 2019. 1, 2, 6

[37] Sebastiano Vigna. A weighted correlation index for rankings with ties. In *Proceedings of the 24th international conference on World Wide Web*, pages 1166–1176, 2015. 5

[38] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018. 6

[39] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 5

[40] Jian Yang and Jing-yu Yang. Why can lda be performed in pca transformed space? *Pattern recognition*, 36(2):563–566, 2003. 6

[41] Kaichao You, Yong Liu, Jianmin Wang, and Mingsheng Long. Logme: Practical assessment of pre-trained models for transfer learning. In *International Conference on Machine Learning*, pages 12133–12143. PMLR, 2021. 1, 2, 5, 6, 7

[42] Hua Yu and Jie Yang. A direct lda algorithm for high-dimensional data—with application to face recognition. *Pattern recognition*, 34(10):2067–2070, 2001. 6

[43] Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features. In *International Conference on Machine Learning*, pages 27179–27202. PMLR, 2022. 2

[44] Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021. 2, 4