TE-BENCH: A UNIFIED BENCHMARK FOR TRANSFERABILITY ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Transferability estimation (TE) predicts which pre-trained checkpoint will achieve the best post-fine-tuning performance on a downstream target task without fine-tuning any of the candidate checkpoints, turning a model search that would cost GPU hours to days of computation into one that runs in seconds to minutes. We built TE-BENCH, a unified evaluation of transferability estimation that covers 22 published TE scores, several experimental setups, and a set of baselines that are rarely compared in the literature. We found that some of these baselines outperform the published TE scores, and we introduce a simple reference baseline (TE-Ref) that significantly outperforms them in the overall setting, which covers supervised and self-supervised vision models across both CNN and ViT architectures. TE-BENCH covers more than 30 transferability estimation scores, 44 distinct checkpoints, and 11 target tasks.

1 INTRODUCTION

Choosing which pre-trained vision model in a model hub to fine-tune on a target downstream task is a costly decision in terms of both time and compute when it is done by brute-force search. Given the growing availability of pre-trained vision models, the question becomes how to select a model from the hub for a target task without carrying out this expensive search. *Transferability estimation* (TE) methods were introduced to address this problem at a far lower cost. These methods estimate the post-fine-tuning performance of a pre-trained model on a target task using forward and backward passes over the target data, and then rank the candidate models by their estimated performance, saving substantial GPU time. The quality of a TE method is evaluated by the agreement between its estimated scores and the ground truth, namely the actual post-fine-tuning performances, measured with the weighted Kendall correlation (Vigna, 2015), a rank-based measure that emphasizes agreement at the top of the ranking.

Over the past several years, a range of TE estimators has appeared, most of them measuring class separability, likelihood or evidence, or adaptation dynamics, such as LEEP (Nguyen et al., 2020), LogME (You et al., 2021), SFDA (Shao et al., 2022), ETran (Gholami et al., 2023), DISCO (Zhang et al., 2025), and ITM (Zheng et al., 2025). In practice, the setup consists of a model hub and a target hub, and an estimator is asked to rank the models in the model hub for a given target task. These estimators are hard to compare: evaluations differ in targets, model hubs, architectures, and ground truth. For the target hub, most papers use a set of eleven downstream tasks (Section 4), which we also adopt in TE-Bench. However, some methods do not evaluate on all of these targets, such as ITM (Zheng et al., 2025) and LEAD (Hu et al., 2024), and others use a different set of tasks, such as the k-NN estimator (Sorkhei et al., 2025). The model hub varies as well, and some works, such as ITM (Zheng et al., 2025), are evaluated on an entirely different model hub. Vision transformers are not evaluated in most existing TE methods, with only a few exceptions such as ITM (Zheng et al., 2025) and SFDA (Shao et al., 2022). Some scores are also not compatible with self-supervised checkpoints, such as ETran (Gholami et al., 2023) and LEEP (Nguyen et al., 2020). As for the ground truth, some methods use their own, such as SFDA (Shao et al., 2022), ITM (Zheng et al., 2025), and the k-NN estimator (Sorkhei et al., 2025), while others rely on published ground truth obtained from the same checkpoints, such as DISCO (Zhang et al., 2025). Finally, these scores are rarely compared against simple controls, which we investigate in TE-Bench.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **Capacity × generalisation.** τ_w against the published ground truth, averaged over the 11 targets. TE-Ref sums a capacity signal (S_{params}) and a generalisation signal (S_{lda}); Δ is its gap to the best published estimator and p the two-sided Wilcoxon signed-rank p-value of that gap across targets.

Hub	S_{params} capacity	S_{lda} generalisation	TE-Ref **combined**	best published estimator	Δ	p
Supervised CNNs (11)	0.561	0.510	**0.686**	ITM 0.674	+0.011	0.64
Self-Supervised CNNs (10)	–	0.734	**0.734**	PACTran 0.729	+0.005	0.67
Vision Transformers (10)	0.678	0.405	0.682	H-Score 0.691	−0.009	0.70
Mixed hub (20)	0.478	0.295	**0.572**	ITM 0.490	+0.083	0.24
Average	0.572	0.486	**0.669**	NCTI 0.567	**+0.102**	**0.003**

We address these challenges with TE-BENCH, a benchmarking platform that gathers TE methods and other baselines and evaluates them under different experimental setups and analysis perspectives of a TE evaluation. TE-BENCH covers more than 30 baselines, comprising published TE methods together with trivial and rarely evaluated controls, 44 distinct pre-trained checkpoints that span supervised and self-supervised CNNs and ViTs, and 11 target datasets, and we release the platform to support unified evaluation and further study of transferability-estimation methods. We then re-evaluate the literature under a single, shared protocol. Section 5 evaluates the published scores across the three model hubs (supervised CNNs, self-supervised CNNs, and ViTs) and, from that analysis, introduces our reference baseline TE-Ref. Section 6 then studies the scores more closely: their behaviour under distribution shift between the target and the pre-training domain, their reliability under target data scarcity, weight-based metrics that require no target data at all, and a combined hub of 20 models covering different architectures. Further investigations are in the Appendix.

Our results show that a simple baseline, which we call **TE-Ref** (TE-Reference), where TE-Ref $= S_{\text{lda}} \oplus S_{\text{params}}$, matches or exceeds the best published TE estimator on two of the three architecture hubs (the supervised and self-supervised CNNs) and on the mixed hub, and significantly outperforms it on the four-hub average, where it improves the weighted Kendall correlation by roughly 18% (Table 1). Here S_{lda} is a generalisation score, the mean true-class probability of a linear discriminant (LDA) with a shared identity covariance on the frozen target features, that is, a nearest-class-mean classifier (Fisher, 1936), and S_{params} a capacity score, the model's parameter count.

We make three contributions. First, we release TE-BENCH, an open-source benchmark for the unified and comprehensive evaluation of TE methods. Second, we re-evaluate the published TE methods under a single shared protocol, together with baselines rarely evaluated in the standard TE setup, and we examine the robustness of all scores under different scenarios, including target data scarcity, distribution shift, and model-architecture diversity. Third, we introduce TE-Ref (TE-Ref), a simple baseline that most published estimators fail to beat.

2 TRANSFERABILITY ESTIMATION

Transferability estimation. Transferability estimation grew out of a handful of likelihood scores around 2019, such as NCE (Tran et al., 2019) and LEEP (Nguyen et al., 2020), and has since expanded into a broad family of estimators built on many different signals. Almost all of them share the same inputs: they read the target features and the target labels produced by a single forward pass of the frozen candidate over the target data, and then compute a cheap statistic from them, as in LogME (You et al., 2021), SFDA (Shao et al., 2022) and H-Score (Bao et al., 2019). They differ in a few requirements that decide where they can be applied. a) Some also read the candidate's pre-training classifier head or its logits, such as LEEP (Nguyen et al., 2020) and ETran (Gholami et al., 2023), and therefore cannot be used on self-supervised checkpoints, which have no such head. b) A smaller, dynamics-based group does not read the frozen state alone but runs a few backward passes with light tuning, as in ITM (Zheng et al., 2025), or simulates a short adaptation of a head, as in LEAD (Hu et al., 2024). c) Most scores use only target data, while a few, such as OTCE (Tan et al., 2021) and PED (Li et al., 2023), additionally require source-domain data.

TE published methods. By what they measure, the published estimators fall into five groups. a) Separability scores quantify how well the frozen features already separate the target classes: H-Score (Bao et al., 2019) and its shrinkage variant H-Score$_{\text{reg}}$ (Ibrahim et al., 2022), the self-challenging Fisher discriminant SFDA (Shao et al., 2022), the neural-collapse geometry of NCTI (Wang et al., 2023), the Gaussian class overlap GBC (Pándy et al., 2022), the distance-correlation PARC (Bolya et al., 2021), the spectral-component score DISCO (Zhang et al., 2025), the feature-perturbation score SA (Khoba et al., 2025), and the k-NN classifier accuracy (Sorkhei et al., 2025). b) Likelihood scores relate the source head's predictions to the target labels: LEEP (Nguyen et al., 2020), NCE (Tran et al., 2019) and $\mathcal{N}$LEEP (Li et al., 2021b). c) Evidence scores place a probabilistic model over the same features and report its marginal likelihood: LogME (You et al., 2021), the PAC-Bayes bound PACTran (Ding et al., 2022), and the energy-augmented ETran (Gholami et al., 2023). d) Transport and information-theoretic scores measure the target task through optimal transport or coding rate: OTCE (Tan et al., 2021), TransRate (Huang et al., 2022) and the intra-class variance measure TMI (Xu & Kang, 2023). e) Dynamics scores estimate adaptation rather than the frozen state, simulating a short fine-tune of a head: PED (Li et al., 2023), LEAD (Hu et al., 2024) and ITM (Zheng et al., 2025). A further confidence estimator, ATC (Garg et al., 2022), thresholds the source head's softmax and, like the likelihood family, applies only on hubs where every candidate has a classifier.

Baselines evaluated in this paper. Against the published methods we evaluate a set of cheap baselines that any estimator should beat, several of which have not previously been used as transferability signals. a) Capacity heuristics read no target data at all, namely the model's parameter count, its penultimate-layer feature dimension, and its ImageNet accuracy (INB). b) The spectral baseline RankMe (Garrido et al., 2023) summarises the singular-value spectrum of the feature matrix Z and needs neither labels nor a head,

$$\mathrm{RankMe}(Z) \;=\; \exp\Big(-\sum_{i} p_i \log p_i\Big), \qquad p_i = \frac{\sigma_i(Z)}{\sum_j \sigma_j(Z)}, \tag{1}$$

a smooth effective rank that is high when the features spread across many dimensions. c) The weight-only baseline reads the trained weights and touches no data. The scale-weighted heavy-tailed self-regularisation exponent α-weighted (Martin et al., 2021) averages the per-layer power-law exponents of each layer's correlation matrix $W^\top W$,

$$\hat{\alpha} \;=\; \frac{1}{L}\sum_{l=1}^{L} \alpha_l \, \log_{10} \lambda_{\max,l}, \tag{2}$$

where $\lambda_{\max,l}$ is the largest eigenvalue of layer l and a smaller $\hat{\alpha}$ indicates a better-trained model. d) Uncertainty statistics read only the source logits on the target data; the maximum softmax probability is the mean top-1 confidence of the source head,

$$\mathrm{MSP} \;=\; \frac{1}{N}\sum_{i=1}^{N} \max_{c} p_{ic}, \qquad p_i = \mathrm{softmax}(z_i), \tag{3}$$

with the softmax entropy and negative log-likelihood as related statistics; like the head-dependent methods they are undefined on self-supervised checkpoints. e) Three feature-geometry readouts round out the set: the clustering agreement k-means NMI, and the representation variance RepVar and label-feature correlation LFC.

A separate line of work does not score the frozen candidate but instead learns the estimator over a meta-set of (model, task) pairs, such as Model-Spider (Zhang et al., 2023), EMMS (Meng et al., 2023) and Task2Vec (Achille et al., 2019). Because these are tuned on ground-truth transfer results and rely on artefacts that a forward-pass platform cannot produce, we do not implement them.

3 Problem Setting

Let $\mathcal{M} = \{m_1, \ldots, m_M\}$ be a *hub* of pre-trained checkpoints and let $\mathcal{T}$ be a target task with data $D = \{(x_i, y_i)\}_{i=1}^{N}$, $y_i \in \{1, \ldots, C\}$. Fine-tuning m on D under a fixed protocol $\mathcal{F}$ yields an accuracy $a(m, \mathcal{T}) = \mathrm{Acc}(\mathcal{F}(m, D)) \in [0, 1]$, and the practitioner wants the best checkpoint $m^\star = \arg\max_m a(m, \mathcal{T})$. Obtaining it exactly costs M fine-tuning runs, which on the hubs we

study is 10 to 20 full training runs per target, the very expense that transfer learning exists to avoid. A transferability estimator replaces this with a function

$$s : (m, D) \longmapsto \mathbb{R}, \tag{4}$$

evaluated *without* running $\mathcal{F}$, whose induced ranking of $\mathcal{M}$ should agree with the ranking induced by $a(\cdot, \mathcal{T})$. Only the ranking matters: any strictly monotone transformation of s gives the same estimator, so we measure agreement with a rank correlation, following the field's convention of the weighted Kendall correlation

$$\tau_w \;=\; \tau_w\Big(\{s(m_j, D)\}_{j=1}^{M},\; \{a(m_j, \mathcal{T})\}_{j=1}^{M}\Big), \tag{5}$$

whose hyperbolic weighting emphasises agreement at the top of the ranking, since that is the part that model selection actually uses. We report τ_w per target and average over targets, and we never pool two hubs, because their ground-truth tables were produced under different protocols $\mathcal{F}$.

Assumptions. We assume white-box but frozen access: we may read a forward pass and the parameters but update no weights, so almost every score s is a function of the activations cached by a single forward pass over D. Concretely, this pass caches a *probe*

$$P(m, \mathcal{T}) \;=\; \big(\, F \in \mathbb{R}^{N \times d},\;\; Z \in \mathbb{R}^{N \times C_s},\;\; y \in \{1, \ldots, C\}^N \,\big), \tag{6}$$

where F are the penultimate features of width d, Z the source logits over the C_s pre-training classes ($Z = \varnothing$ for head-less checkpoints), and y the target labels. Almost every score is a function of P alone; a few source-dependent scores additionally read a source-domain probe $P(m, \mathcal{S})$, and a few weight-based scores read the checkpoint's weight matrices. The ground truth $a(m, \mathcal{T})$ is always taken from a published table rather than from our own runs.

4 The TE-Bench Platform

TE-Bench is our benchmark for transferability estimation. It puts the score, the model hub, the sample budget and the ground-truth table behind one interface, so each can be varied while the rest stays fixed:

`https://anonymous.4open.science/r/random-8D3C`

We describe the model hubs, targets and scores below, and give the full details in Appendix A.1: the name and pre-trained weight source of every checkpoint, the ground-truth tables, the target datasets and their statistics.

Model hub. A hub is a set of checkpoints sharing one ground-truth table, and we study four. **Supervised CNNs** (11): ResNet-34/50/101/152 (He et al., 2016), DenseNet-121/169/201 (Huang et al., 2017), GoogLeNet (Szegedy et al., 2015), Inception-v3 (Szegedy et al., 2016), MnasNet-1.0 (Tan et al., 2019) and MobileNet-v2 (Sandler et al., 2018). **Self-Supervised CNNs** (10), all ResNet-50 and differing only in pre-training objective: InsDis (Wu et al., 2018), MoCo-v1 (He et al., 2020), MoCo-v2 (Chen et al., 2020), InfoMin (Tian et al., 2020), PCL-v1/v2 (Li et al., 2021a), SeLa-v2 (Asano et al., 2020), SwAV (Caron et al., 2020), DeepCluster-v2 (Caron et al., 2018) and BYOL (Grill et al., 2020). **Vision Transformers** (10): ViT-T/S/B-16 (Dosovitskiy et al., 2021; Steiner et al., 2022), PVT-Tiny/Small/Medium (Wang et al., 2021), PVTv2-B2 (Wang et al., 2022), Swin-T (Liu et al., 2021), and the self-supervised DINO ViT-S/16 (Caron et al., 2021) and MoCo-v3 ViT-S/16 (Chen et al., 2021). **Mixed hub** (20): ResNet-18/34/50/101/152, DenseNet-121/161/169/201, GoogLeNet, Inception-v3, MnasNet-1.0, MobileNet-v2, EfficientNet-B0 (Tan & Le, 2019), and the transformers DINO ViT-S/8, DINO ViT-B/16, MoCo-v3 ViT-B/16, MAE ViT-B/16 (He et al., 2022), MAE ViT-L/16 and SimMIM ViT-B/16 (Xie et al., 2022).

Target hub. All four model hubs are evaluated on the 11 target tasks: Aircraft (Maji et al., 2013), Caltech-101 (Fei-Fei et al., 2004), Cars (Krause et al., 2013), CIFAR-10/100 (Krizhevsky, 2009), DTD (Cimpoi et al., 2014), Flowers (Nilsback & Zisserman, 2008), Food-101 (Bossard et al., 2014), Pets (Parkhi et al., 2012), SUN397 (Xiao et al., 2010) and VOC2007 (Everingham et al., 2010), spanning 10 to 397 classes and two orders of magnitude in size.

Table 2: Per-target weighted Kendall τ_w at the full sample budget for the five best scores of each hub and the reference TE-Ref; the shaded column is the per-hub average, and the final block gives SFDA, NCTI, and TE-Ref averaged over the three hubs.

Hub	Score	Airc.	Cal101	Cars	C10	C100	DTD	Flwr	Food	Pets	SUN	VOC	Avg
Sup. CNNs	α-weighted	0.64	0.77	0.64	0.71	0.75	0.76	0.72	0.93	0.71	0.78	0.52	0.721
	ITM	0.32	0.80	0.42	0.72	0.75	0.67	0.94	0.76	0.82	0.69	0.53	0.674
	NCTI	0.44	0.74	0.59	0.86	0.88	0.65	0.46	0.79	0.54	0.73	0.70	0.669
	SFDA	0.61	0.74	0.47	0.93	0.85	0.56	0.52	0.83	0.66	0.68	0.42	0.659
	INB	0.73	0.63	0.73	0.78	0.76	0.74	0.25	0.54	0.73	0.67	0.65	0.654
	TE-Ref	0.79	0.61	0.80	0.75	0.82	0.81	0.18	0.54	0.82	0.68	0.76	0.686
SSL CNNs	LDA	0.06	0.77	0.80	0.75	0.84	0.89	0.87	0.92	0.62	0.92	0.64	0.734
	PACTran	0.06	0.77	0.80	0.85	0.74	0.89	0.85	0.84	0.62	0.92	0.69	0.729
	SA	0.06	0.77	0.80	0.75	0.78	0.89	0.87	0.92	0.62	0.92	0.64	0.729
	TransRate	0.08	0.78	0.70	0.89	0.76	0.89	0.83	0.86	0.61	0.87	0.70	0.724
	SFDA	0.23	0.65	0.68	0.79	0.79	0.90	0.81	0.84	0.60	0.83	0.62	0.705
	TE-Ref	0.06	0.77	0.80	0.75	0.83	0.89	0.87	0.92	0.62	0.92	0.64	0.734
ViTs	H-Score	0.46	0.81	0.67	0.64	0.75	0.75	0.58	0.62	0.57	0.82	0.93	0.691
	LEAD	0.46	0.73	0.67	0.64	0.75	0.75	0.58	0.62	0.57	0.82	0.91	0.682
	#Params	0.52	0.62	0.54	0.65	0.77	0.78	0.71	0.56	0.74	0.80	0.76	0.678
	LogME	0.26	0.79	0.46	0.59	0.75	0.73	0.60	0.62	0.51	0.77	0.74	0.620
	RankMe	0.47	0.30	0.64	0.23	0.44	0.64	0.85	0.77	0.89	0.58	0.73	0.594
	TE-Ref	0.27	0.65	0.57	0.61	0.76	0.85	0.74	0.57	0.82	0.83	0.84	0.682
SFDA, avg. over hubs		0.37	0.62	0.52	0.77	0.78	0.77	0.63	0.76	0.64	0.68	0.61	0.649
NCTI, avg. over hubs		0.21	0.78	0.32	0.73	0.78	0.65	0.50	0.74	0.60	0.72	0.63	0.607
TE-Ref, avg. over hubs		0.37	0.67	0.72	0.70	0.80	0.85	0.60	0.68	0.75	0.81	0.75	0.701

Score hub. The score hub contains more than 30 scores. They cover the published TE methods (Section 2) and other categories of metrics that can serve as transferability signals: capacity baselines such as the parameter count and the feature dimension, the source model's ImageNet (Deng et al., 2009) accuracy, prediction-based scores such as MSP, entropy and NLL, spectral metrics such as RankMe (Garrido et al., 2023), and weight-based metrics such as α-weighted (Martin et al., 2021).

5 TE-BENCH EVALUATION

Results. Table 2 reports the per-target τ_w (weighted Kendall correlation) of the transferability scores against the ground truth, the post-fine-tuning accuracy; higher and positive values indicate a stronger transferability estimate. For readability we list, on each hub, the five scores with the highest average over the target datasets, and give the full results in Appendix A.5. The second and third hubs contain self-supervised checkpoints, so any score that relies on a source classifier is undefined there and is omitted on those hubs. On the supervised CNN hub the strongest score is the data-free α-weighted metric (Martin et al., 2021) ($\tau_w = 0.721$), which reads only the pre-trained weights yet outperforms every dedicated transferability score. On the self-supervised hub LDA leads ($\tau_w = 0.734$), and on the ViT hub H-Score is best ($\tau_w = 0.691$).

The top performers on each hub. The best score on each hub is decided largely by what the hub holds fixed and what it varies. The supervised CNN hub spans eleven architectures from 3.5M to 60M parameters, and there the data-free α-weighted metric, a proxy for how well a backbone was trained, leads. Simple capacity baselines already order the models well on this hub, with INB, the pre-trained model's ImageNet accuracy, at $\tau_w = 0.654$, close to the best published estimator ITM (0.674), and the raw parameter count at 0.561 (Table 12). The self-supervised CNN hub instead fixes the architecture to a single ResNet-50 (25.6M) and varies only the pre-training weights, so no capacity spread remains: the simple LDA baseline now leads (0.734), and α-weighted even reverses to -0.562 (Table 13); the premise that a smaller α marks a better model was established on supervised ImageNet models and on language models (Martin et al., 2021), not on self-supervised vision backbones. On the ViT hub H-Score is strongest, but its gain over the parameter count alone is small, from 0.678 to 0.691.

TE-Ref. These observations point to two factors behind the fine-tuning order. The first is generalisation, meaning how well the pre-trained features already suit the target. When capacity is held fixed, as on the self-supervised hub, this is the only factor left, and even a linear separability measure captures it: LDA on the frozen features leads that hub ($\tau_w = 0.734$). The second is capacity. When model size varies, the frozen features do not show how much a larger backbone gains from fine-tuning, and the parameter count alone already orders the models well (0.561 on the supervised CNN hub and 0.678 on the ViT hub). We therefore define a reference baseline, TE-Ref, that combines a generalisation signal and a capacity signal,

$$\text{TE-Ref}(m,t) \;=\; S_{\text{lda}}(m,t) \;\oplus\; S_{\text{params}}(m), \tag{7}$$

where S_{lda} is the LDA separability score on the frozen features and S_{params} the model's parameter count. We min-max normalise each over the models of a hub and add them with the operator $\oplus$, weighting each element by its share of the two summed standard deviations, so a signal that varies more across the hub counts for more. Averaged over the three hubs TE-Ref reaches $\tau_w = 0.701$, above the best published estimator (SFDA, 0.649), and it stays within 0.01 of the best data-dependent score on each hub, while no published method leads on more than one hub. The weighting also makes TE-Ref adaptive: on the self-supervised hub every backbone is the same size, so S_{params} has zero standard deviation, its weight vanishes, and TE-Ref reduces to S_{lda} exactly.

6 Discussion & Analysis

Distribution shift. Averaging a score over targets hides whether it is strong on targets close to the pre-training domain, far from it, or throughout. We measure the drift of each (model, target) pair in feature space, comparing the model's target and source features with four statistics (an ℓ_2 mean gap, a cosine gap, a diagonal-Gaussian Fréchet distance, and linear CKA; see Appendix A.6). We then split the targets at the per-hub median ℓ_2 mean gap into a low-shift and a high-shift half (10.0 on supervised CNNs, 19.9 on transformers, 1.2 on the self-supervised CNNs, which stay close to the source because the hub *is* ImageNet pre-training) and re-evaluate each score within each half.

Table 3 shows that most scores degrade on the high-shift targets. On the supervised CNN hub ITM loses 0.082 of its τ_w, α-weighted 0.053 and NCTI 0.045, and SFDA is the only listed score that gains ($+0.026$). The drops are larger on the other two hubs: on the self-supervised hub LDA and TransRate lose 0.131, PACTran 0.128, SA 0.123 and SFDA 0.079, and on the ViT hub LogME loses 0.158, H-Score 0.128, RankMe 0.126 and LEAD 0.034. TE-Ref follows the same pattern, with drops of 0.034, 0.131 and 0.169 on the three hubs. The average τ_w over all targets therefore hides how a score behaves on targets far from the pre-training data. Since a practitioner may care only about high-shift targets in a real deployment, results should be reported on both halves rather than as a single average. Full per-score degradations are in Appendix A.6.

Data scarcity. Table 4 sweeps the target sample budget from three samples per class to the full target set. Most scores degrade as the budget shrinks. On supervised CNNs ITM falls from 0.674 at the full budget to 0.310 at three samples per class, NCTI from 0.669 to 0.194 and SFDA from 0.659 to 0.201; on the self-supervised hub SA falls from 0.729 to 0.561 and NCTI from 0.702 to 0.475; and on the ViT hub H-Score falls from 0.691 to 0.296 and LEAD from 0.682 to 0.508. A few scores hold their level: LDA (0.718 at three samples, 0.734 at the full budget) and TransRate (0.711 and 0.724) on the self-supervised hub, LogME on the ViT hub, which is higher at three samples per class (0.691) than at the full budget (0.620), and the data-free scores (α-weighted, INB and the parameter count), which do not read the target at all. Most published TE scores therefore depend on the target sample budget, which limits their use in real scenarios where target data is scarce. The most stable data-dependent scores are MSP and NLL, which read the source classifier's predictions on the target: on supervised CNNs their lowest τ_w over the whole sweep is 0.558 and 0.521, the two highest minima among the data-dependent scores and above every published TE score (Table 27).

Computation time. Full per-target timings for every hub are in Appendix Tables 8 to 11. Data-free scores such as the parameter count and INB need no computation on the target. Every other score pays two costs per target: the feature extraction, on average 114 s per target on supervised CNNs (from 12 s on DTD to 329 s on CIFAR-10), and the score computation itself, which takes 14 to 18 s for cheap scores such as H-Score, ITM and LDA and 80 to 312 s for heavier ones such as

Table 3: **Data perspective: τ_w under low vs. high source-target shift.** The same top-5 scores per hub as Table 4, together with the reference TE-Ref. Targets are split at the per-hub median feature mean-gap into a low-shift half (6 targets, gap $\leq$ median) and a high-shift half (5 targets); Δ = high − low, so a negative value is a drop under shift. Red marks scores that degrade under shift, green a score that gains.

	Score	low-shift	high-shift	Δ
Sup. CNNs	α-weighted	0.745	0.692	−0.053
	ITM	0.712	0.630	−0.082
	NCTI	0.690	0.645	−0.045
	SFDA	0.647	0.673	+0.026
	INB	0.658	0.650	−0.008
	TE-Ref	0.701	0.667	−0.034
SSL CNNs	LDA	0.793	0.662	−0.131
	PACTran	0.787	0.659	−0.128
	SA	0.725	0.602	−0.123
	TransRate	0.783	0.652	−0.131
	SFDA	0.741	0.662	−0.079
	TE-Ref	0.793	0.662	−0.131
ViTs	H-Score	0.749	0.621	−0.128
	Params	0.710	0.639	−0.071
	LEAD	0.641	0.607	−0.034
	LogME	0.692	0.533	−0.158
	RankMe	0.651	0.525	−0.126
	TE-Ref	0.759	0.590	−0.169

Table 4: **τ_w versus target samples per class.** Mean over the 11 targets for the five best scores of each hub at the full budget. **Bold** marks the best score of each hub at the full budget (shaded column).

	Score	3	4	5	10	20	50	100	all
Sup. CNNs	α-weighted	0.721	0.721	0.721	0.721	0.721	0.721	0.721	**0.721**
	ITM	0.310	0.350	0.376	0.564	0.497	0.642	0.656	0.674
	NCTI	0.194	0.314	0.506	0.454	0.641	0.681	0.681	0.669
	SFDA	0.201	0.201	0.281	0.241	0.454	0.457	0.588	0.659
	INB	0.654	0.654	0.654	0.654	0.654	0.654	0.654	0.654
SSL CNNs	LDA	0.718	0.719	0.717	0.729	0.731	0.734	0.734	**0.734**
	SA	0.561	0.647	0.620	0.683	0.731	0.734	0.695	0.729
	TransRate	0.711	0.713	0.713	0.713	0.712	0.712	0.718	0.724
	SFDA	0.576	0.661	0.632	0.695	0.706	0.701	0.703	0.705
	NCTI	0.475	0.527	0.525	0.645	0.668	0.706	0.716	0.702
ViTs	H-Score	0.296	0.276	0.386	0.576	0.598	0.612	0.677	**0.691**
	LEAD	0.508	0.539	0.556	0.623	0.627	0.660	0.657	0.682
	Params	0.678	0.678	0.678	0.678	0.678	0.678	0.678	0.678
	LogME	0.691	0.680	0.693	0.711	0.699	0.629	0.625	0.620
	RankMe	0.437	0.478	0.534	0.569	0.583	0.588	0.598	0.594

PED, LEAD and PACTran. The weight-based metrics are data-free as well: they are computed once per model from its weights, independently of the target, and so add no per-target cost.

Weight-only scores: α-weighted. Weight-only scores read the trained weights alone and need no target data at all, neither images nor labels, which makes them the cheapest transferability signal available. On the supervised CNN hub one of them is also the best score overall: α-weighted (Eq. (2)) reaches $\tau_w = 0.721$, above every published estimator. On the self-supervised hub, however, its ranking reverses ($\tau_w = -0.562$). The reversal comes from the two quantities the metric multiplies, the heavy-tail exponent α_l and the log spectral norm $\log_{10} \lambda_{l,\max}$, both of which change direction between the two hubs: among supervised CNNs a smaller exponent and a smaller norm accompany better transfer ($\tau_w = 0.464$ and -0.765 for the two terms scored separately), whereas among the self-supervised ResNet-50s a larger exponent and a larger norm do (-0.508 and 0.672). The premise that a heavier tail marks a better-trained model was established on supervised ImageNet models and on language models (Martin et al., 2021), not on self-supervised vision backbones, and because every convolution in these ResNet-50s is followed by batch normalisation, the scale of its weights does not change the function it computes, so the norm is not tied to the features the network produces and varies instead with how each method was optimised.

Weight-only scores: effective rank. A second weight statistic is the effective rank of the layer weights. For each layer we flatten the weight into $W_l \in \mathbb{R}^{m_l \times n_l}$, normalise its singular values into $p_{l,i} = \sigma_{l,i} / \sum_j \sigma_{l,j}$, and take the maximum over the layers outside the classifier head $\mathcal{H}$:

$$\operatorname{erank}(W_l) \;=\; \exp\Big(-\sum_i p_{l,i} \log p_{l,i} \Big), \qquad S_{\mathrm{er}} \;=\; \max_{l \notin \mathcal{H}} \operatorname{erank}(W_l), \tag{8}$$

the quantity RankMe computes on features (Eq. (1)) applied to the weights, which ranges from 1 to $\min(m_l, n_l)$ and counts how many directions a layer effectively uses. It reaches $\tau_w = \mathbf{0.520}$ on supervised CNNs, $\mathbf{0.593}$ on self-supervised CNNs and $\mathbf{0.662}$ on ViTs, with no reversal between the supervised and the self-supervised hub (Appendix B.4).

Combined hub. To evaluate the scores on a wider and more diverse hub, we use a fourth hub of 20 checkpoints from 3.5M to 303M parameters with its own ground truth, which mixes CNNs and vision transformers trained with supervised and self-supervised objectives. The correlations are lower than on the other three hubs (Table 5): the best published estimator is ITM with $\tau_w = 0.490$, against 0.674 to 0.729 on the other hubs, and the parameter count alone is the second best score (0.478). TE-Ref reaches 0.572, 0.083 above the best published estimator, while each of its two signals is lower on its own (parameter count 0.478, LDA 0.295). Per-target results for every score on this hub are in Table 15.

Statistical Significance. TE papers rarely report whether the improvement of a new score is statistically significant, which is essential when a model hub holds only 10 to 20 models and the evaluation spans 11 targets. We pair TE-Ref with the strongest published score of each hub, target by target ($n = 11$, the target being the unit of replication), and with NCTI over the 44 pooled hub-target pairs, and test each gap with a two-sided Wilcoxon signed-rank test. On the individual hubs the differences are small and none is significant: $\Delta\tau_w = +0.011$ against ITM on supervised CNNs ($p = 0.64$), $+0.005$ against PACTran on self-supervised CNNs ($p = 0.67$), -0.009 against H-Score on ViTs ($p = 0.70$), and $+0.083$ against ITM on the combined hub ($p = 0.24$), where eleven targets cannot resolve the larger gap. Pooled across all four hubs the margin over NCTI is $+0.102$ and significant ($p = 0.003$), so TE-Ref, a simple baseline, significantly outperforms the current transferability scores across these four hubs.

Fine-tuning protocols. A transferability score is evaluated against a fine-tuning ground-truth, yet that ground-truth is itself a protocol-dependent measurement, since the accuracy a checkpoint attains depends on the optimiser, schedule and hyper-parameter search used to fine-tune it. Two of our model-hubs share seven checkpoints under identical weights but are paired with ground-truths obtained under different protocols: the SFDA ground-truth (Shao et al., 2022) grid-searches the learning rate and weight decay per (model, target) and retains the best configuration, thereby approximating a maximum over training recipes, whereas the ITM ground-truth (Zheng et al., 2025) adopts a single fixed AdamW schedule of 100 epochs and reports the accuracy of one recipe. Using one ground-truth as a transferability estimator for the other, an estimator that reproduced a given protocol exactly would still be bounded by the agreement between the two, which across the seven shared checkpoints is $\tau_w = 0.812$ and varies substantially across targets, from 0.960 on CIFAR-100

Table 5: **Combined hub.** Per-target weighted Kendall τ_w at the full sample budget for the five best scores of the combined hub and the reference TE-Ref; the shaded column is the average over the 11 targets.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flwr	Food	Pets	SUN	VOC	Avg
ITM	0.09	0.59	0.29	0.54	0.51	0.64	0.56	0.43	0.41	0.68	0.64	0.490
#Params	0.13	0.49	0.41	0.64	0.77	0.53	0.30	0.52	0.39	0.61	0.47	0.478
NCTI	0.22	0.43	0.18	0.45	0.46	0.79	0.37	0.42	0.68	0.53	0.41	0.448
DISCO	0.24	0.43	0.05	0.36	0.23	0.59	0.60	0.55	0.41	0.38	0.16	0.363
$\mathcal{N}$LEEP	0.07	0.31	0.08	0.48	0.40	0.42	0.46	0.47	0.35	0.47	0.32	0.349
TE-Ref	0.21	0.54	0.64	0.52	0.71	0.64	0.21	0.73	0.72	0.74	0.66	**0.572**

down to 0.551 on VOC2007. The fine-tuning accuracy obtained under one recipe therefore does not fully determine the ordering of the models under another, so part of the gap between a score and a single ground truth reflects the fine-tuning protocol rather than the score. This analysis is limited to seven checkpoints and a single pair of protocols, and could be strengthened with additional shared checkpoints and further ground-truths obtained under different protocols.

Transferability estimation in practice. Before using these scores, a practitioner should consider the target dataset and the model hub. The first question is how far the target is from the pre-training data: most TE scores degrade on high-shift targets (Table 3), so their average τ_w overstates their quality there. The second is whether target labels are available. Almost every published TE score needs them; the label-free alternatives are prediction-based scores such as MSP and entropy, which require the source classifier head and so cannot be applied to self-supervised checkpoints, and RankMe, which reads only the target features (Table 28). The third is how much target data is available: with few samples per class many TE scores degrade, while the prediction-based scores MSP and NLL remain the most stable (Table 27). When no target data is available at all, the two weight-based metrics discussed above, α-weighted and the weight effective rank, remain applicable because they read only the pre-trained weights, although α-weighted reverses on the self-supervised hub. Finally, the composition of the hub matters: when model capacity varies, a capacity signal such as the parameter count carries much of the ranking, whereas in a hub of fixed capacity, such as our self-supervised hub, a generalisation signal such as LDA is the one that helps. This is why we introduced TE-Ref, which combines both signals and can therefore be used on diverse hubs as well, including the combined hub of Table 5.

7 Conclusion

We introduced TE-Bench, a unified platform for benchmarking transferability estimation that covers different model hubs, from supervised and self-supervised CNNs to vision transformers, and different target tasks. Using it, we analysed the current transferability scores through several lenses, such as distribution shift and target data scarcity, and summarised the results as guidance for using TE in practice. We also studied weight-based metrics, which have received little attention as transferability estimators and require no target data at all. Finally, we introduced TE-Ref, a simple reference baseline that significantly outperforms the current TE scores in the overall setting. The main limitation is the ground truth: our conclusions rest on a limited set of published ground-truths, and richer benchmarks with more checkpoints and additional fine-tuning protocols would let them be examined further.

Reproducibility Statement

The code of TE-Bench is available at `https://anonymous.4open.science/r/random-8D3C`. The implementation details of every score, the model and target hubs, and the full results are provided in Part A of the appendix (Appendix A).

REFERENCES

Alessandro Achille, Michael Lam, Rahul Tewari, Avinash Ravichandran, Subhransu Maji, Charless Fowlkes, Stefano Soatto, and Pietro Perona. Task2Vec: Task embedding for meta-learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations (ICLR)*, 2020.

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 *IEEE International Conference on Image Processing (ICIP)*, 2019.

Daniel Bolya, Rohit Mittapalli, and Judy Hoffman. Scalable diverse model selection for accessible transfer learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, 2014.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision (ECCV)*, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

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

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Nan Ding, Xi Chen, Tomer Levinboim, Soravit Changpinyo, and Radu Soricut. PACTran: PAC-Bayesian metrics for estimating the transferability of pretrained models to classification tasks. In *European Conference on Computer Vision (ECCV)*, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2), 2010.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR Workshop on Generative-Model Based Vision*, 2004.

Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7 (2), 1936.

Saurabh Garg, Sivaraman Balakrishnan, Zachary C. Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. In *International Conference on Learning Representations (ICLR)*, 2022.

Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann LeCun. RankMe: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International Conference on Machine Learning (ICML)*, 2023.

Mohsen Gholami, Mohammad Akbari, Xinglu Wang, Behnam Kamranian, and Yong Zhang. ETran: Energy-based transferability estimation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Zixuan Hu, Xiaotong Li, Shixiang Tang, Jun Liu, Yichun Hu, and Ling-Yu Duan. LEAD: Exploring logit space evolution for model selection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Long-Kai Huang, Junzhou Huang, Yu Rong, Qiang Yang, and Ying Wei. Frustratingly easy transferability estimation. In *International Conference on Machine Learning (ICML)*, 2022.

Shibal Ibrahim, Natalia Ponomareva, and Rahul Mazumder. Newer is not always better: Rethinking transferability metrics, their peculiarities, stability and performance. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2022.

Prafful Kumar Khoba, Zijian Wang, Chetan Arora, and Mahsa Baktashmotlagh. Feature space perturbation: A panacea to enhanced transferability estimation. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *IEEE International Conference on Computer Vision Workshops*, 2013.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations (ICLR)*, 2021a.

Xiaotong Li, Zixuan Hu, Yixiao Ge, Ying Shan, and Ling-Yu Duan. Exploring model transferability through the lens of potential energy. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Yandong Li, Xuhui Jia, Ruoxin Sang, Yukun Zhu, Bradley Green, Liqiang Wang, and Boqing Gong. Ranking neural checkpoints. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021b.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

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

Charles H. Martin, Tongsu Peng, and Michael W. Mahoney. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nature Communications*, 12 (1), 2021.

Fanqing Meng, Wenqi Shao, Zhanglin Peng, Chonghe Jiang, Kaipeng Zhang, Yu Qiao, and Ping Luo. Foundation model is efficient multimodal multitask model selector. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Cuong V. Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. LEEP: A new measure to evaluate transferability of learned representations. In *International Conference on Machine Learning (ICML)*, 2020.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

Michal Pándy, Andrea Agostinelli, Jasper Uijlings, Vittorio Ferrari, and Thomas Mensink. Transferability estimation using Bhattacharyya class separability. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

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 *European Conference on Computer Vision (ECCV)*, 2022.

Moein Sorkhei, Christos Matsoukas, Johan Fredin Haslum, Emir Konuk, and Kevin Smith. k-NN as a simple and effective estimator of transferability. *Transactions on Machine Learning Research (TMLR)*, 2025.

Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your ViT? data, augmentation, and regularization in vision transformers. *Transactions on Machine Learning Research (TMLR)*, 2022.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, 2019.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Yang Tan, Yang Li, and Shao-Lun Huang. OTCE: A transferability metric for cross-domain cross-task representations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Anh T. Tran, Cuong V. Nguyen, and Tal Hassner. Transferability and hardness of supervised classification tasks. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Weijie Tu, Weijian Deng, Dylan Campbell, Yu Yao, Jiyang Zheng, Tom Gedeon, and Tongliang Liu. Ranked from within: Ranking large multimodal models without labels. In *International Conference on Machine Learning (ICML)*, 2025.

Sebastiano Vigna. A weighted correlation index for rankings with ties. In *Proceedings of the 24th International Conference on World Wide Web (WWW)*, 2015.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. PVT v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3), 2022.

Zijian Wang, Yadan Luo, Liang Zheng, Zi Huang, and Mahsa Baktashmotlagh. How far pre-trained models are from neural collapse on the target dataset informs their transferability. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. SimMIM: A simple framework for masked image modeling. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Huiwen Xu and U Kang. Fast and accurate transferability measurement by evaluating intra-class feature variance. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

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 (ICML)*, 2021.

Tengxue Zhang, Yang Shu, Xinyang Chen, Yifei Long, Chenjuan Guo, and Bin Yang. Assessing pre-trained models for transfer learning through distribution of spectral components. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2025.

Yi-Kai Zhang, Ting-Ji Huang, Yao-Xiang Ding, De-Chuan Zhan, and Han-Jia Ye. Model spider: Learning to rank pre-trained models efficiently. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Yaoyan Zheng, Huiqun Wang, Nan Zhou, and Di Huang. Implicit modeling for transferability estimation of vision foundation models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Appendix contents. The appendix has two parts.

- **Part A: Full Details and Full Results**: platform setup (model hub A.1, target hub A.2), score implementations, computation time (A.4), full TE-Bench evaluation (A.5), distribution shift (A.6), data scarcity and sweep robustness (A.7), label-free scores, and leave-four-out robustness (A.9).
- **Part B: Further Analysis**: alternative metrics (B.1), absolute transferability (B.2), the feature-dimension confound (B.3), weight-only scores (B.4), robustness to pre-trained weights (B.5), the language modality (B.6), score redundancy (B.7), and variance reduction by cross-fitting (B.8).

AI USE STATEMENT

In this work, we used generative AI tools to help re-implement some of the transferability estimation scores that have no official code release; these implementations were carefully reviewed and refined by the authors. Additionally, we used generative AI tools for polishing and editing some parts of the text, and for converting the benchmark's result files (JSON) into the LaTeX tables of the paper; all generated tables were carefully reviewed and re-checked by the authors against the benchmark outputs. We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

A Full Details and Full Results

A.1 Model hub: checkpoints and source accuracy

Table 6 lists every checkpoint in the four hubs with the pre-trained weights it loads, the published ImageNet-1k top-1 accuracy of its source classifier — the `source_top1` signal, which also serves as the trivial *source top-1* baseline and feeds the capacity half of TE-Ref — and its fine-tuning ground-truth top-1 on all 11 targets (the quantity every score is trying to rank), with the per-checkpoint average. The supervised checkpoints load the pinned `IMAGENET1K_V1` torchvision weights or the corresponding `timm` tags; the self-supervised backbones (all ResNet-50 for the SFDA hub) come from the SFDA/PARC release and the self-supervised transformers from `timm` or local checkpoints. One subtlety: the mixed (ITM) hub reuses several architectures from the supervised-CNN hub but under torchvision's `DEFAULT` weights, which are `IMAGENET1K_V2` for ResNet-50/101/152 and MobileNet-v2; those four rows therefore carry the V2 weights and the higher V2 source top-1, and their ground-truth differs from the V1 rows of the supervised-CNN hub. A self-supervised backbone has no classifier head, so it has no source top-1 (dash) and every head-dependent score is undefined on it (Section 5).

A.2 Target statistics

Table 7 lists the 11 targets with their class counts, the number of samples N used for feature extraction and inference, and the feature-extraction wall-clock (seconds, summed over all models of the hub). The per-sample cost of a score on a target is its total wall-clock divided by N; the table's cost columns average this over targets.

A.3 Score implementations

All scores in TE-Bench read a shared cache built by one forward pass of each frozen checkpoint over the target data, which stores the penultimate features, the source-head logits when the checkpoint has an ImageNet classifier, and the target labels. A few scores also read the source domain (the same forward pass over $5{,}000$ ImageNet validation images (Deng et al., 2009)), the trained weights, or model metadata (parameter count, feature dimension and published ImageNet top-1). For the published methods we adapted the code released by the authors whenever it is public. When no code was released, we re-implemented the method from the equations and details given in the paper. Every score is oriented so that a higher value predicts better transfer, so costs, entropies and variances are negated. Scores with several terms store the raw terms and combine them after a min-max normalisation over the models of the hub, using the combination given by the original authors.

Separability scores. H-Score (Bao et al., 2019) is $\mathrm{tr}\big(\mathrm{Cov}(F)^{+}\,\mathrm{Cov}(\mathbb{E}[F \mid Y])\big)$, where F are the target features, Y the labels, both covariances are population covariances over the target samples, $\mathbb{E}[F \mid Y]$ replaces each sample by the mean of its class, and $^{+}$ is the pseudo-inverse with relative cutoff 10^{-15}. H-Score$_{\text{reg}}$ (Ibrahim et al., 2022) replaces $\mathrm{Cov}(F)$ with $(1-a)\,\mathrm{Cov}(F) + a\,\frac{\mathrm{tr}(\mathrm{Cov}(F))}{d}\,I$, where d is the feature dimension and a is the Ledoit-Wolf shrinkage intensity of the centred features, clipped to $[10^{-6}, 1]$. SFDA (Shao et al., 2022) has two stages. With π_c and μ_c the share and mean of class c and $S_w = \sum_c \pi_c\,\mathrm{Cov}_c$ the within-class covariance, the first stage shrinks S_w to $\tilde{S}_w = (1-a)\,S_w + a\,\frac{\mathrm{tr}(S_w)}{d}\,I$ with $a = \max\big(e^{-5\lambda_{\max}(S_w)}, 10^{-10}\big)$, where $\lambda_{\max}(S_w)$ is estimated by three steps of power iteration, and classifies each sample with the discriminant logits $z_{ic} = x_i^{\top}\tilde{S}_w^{-1}\mu_c - \frac{1}{2}\mu_c^{\top}\tilde{S}_w^{-1}\mu_c + \log\pi_c$. The ConfMix step then moves every sample towards the average of the other class means, $x_i \leftarrow q_i\,x_i + (1-q_i)\,\bar{\mu}_{-y_i}$, where q_i is the true-class entry of a second softmax applied to the first-stage probabilities, as in the released code. The second stage fits the same discriminant on the moved features with the same a, and the score is its mean true-class softmax probability. NCTI (Wang et al., 2023) reduces the features to 64 dimensions by PCA, fits a shrinkage LDA whose intensity is set by the leading PCA variances, and combines its three terms as cls_conf + feature_nuc − log class_pred_nuc. GBC (Pándy et al., 2022) sums the negative Bhattacharyya coefficients between class Gaussians with diagonal covariances on 64 PCA dimensions. PARC (Bolya et al., 2021) takes the Spearman correlation between the feature and label dissimilarity matrices on 32 PCA dimensions, with at most $5{,}000$ samples. DISCO (Zhang et al., 2025) weights each of the top 64 spectral components of the centred features by its share of the spectral energy and

Table 6: **Model hub: checkpoint provenance, source accuracy and fine-tuning ground-truth.** Each checkpoint of the four hubs: the pre-trained weights it loads, the published ImageNet-1k top-1 of its source classifier (*src top-1*), and its fine-tuning ground-truth top-1 on the 11 targets with the per-checkpoint average. Within a hub, rows are ordered by source top-1; self-supervised backbones have no classifier and hence no source top-1. The mixed hub loads torchvision `DEFAULT` (V2) weights for ResNet-50/101/152 and MobileNet-v2, so those rows differ from their V1 counterparts in the supervised-CNN hub.

Checkpoint	Pre-trained weights	src	Fine-tuning ground-truth top-1 (%)											
		top-1	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Supervised CNNs (SFDA)														
ResNet-152	IN-1k (tv V1)	78.3	86.3	93.1	89.9	97.5	85.7	76.4	96.9	86.3	94.4	64.8	86.3	87.1
ResNet-101	IN-1k (tv V1)	77.4	85.5	92.4	89.5	97.4	84.9	74.8	96.5	85.6	93.9	63.8	85.7	86.4
Inception-v3	IN-1k (tv V1)	77.3	80.2	92.8	87.7	96.2	81.5	72.8	95.7	81.8	92.1	60.0	83.8	84.1
DenseNet-201	IN-1k (tv V1)	76.9	85.4	93.1	89.4	97.0	84.9	76.0	97.1	86.7	94.0	64.6	85.7	86.7
ResNet-50	IN-1k (tv V1)	76.1	84.6	92.0	89.1	96.3	82.8	74.7	96.3	84.5	93.9	63.5	85.8	85.8
DenseNet-169	IN-1k (tv V1)	75.6	84.2	92.5	89.0	96.8	84.3	74.7	97.3	85.8	93.6	64.1	85.8	86.2
DenseNet-121	IN-1k (tv V1)	74.4	84.7	91.5	89.3	96.5	82.8	74.2	97.0	85.0	93.1	63.3	85.3	85.7
MNASNet-1.0	IN-1k (tv V1)	73.5	66.5	89.3	72.6	92.6	72.0	70.1	95.4	71.3	91.1	56.6	81.1	78.1
ResNet-34	IN-1k (tv V1)	73.3	84.1	91.2	88.6	96.1	81.9	73.0	95.2	82.0	93.5	61.0	84.6	84.7
MobileNet-v2	IN-1k (tv V1)	71.9	79.7	88.6	86.4	94.7	78.1	71.7	96.2	81.1	91.3	60.3	82.8	82.8
GoogLeNet	IN-1k (tv V1)	69.8	80.3	90.8	87.8	95.5	79.8	72.5	95.8	79.3	91.4	59.9	82.6	83.3
Self-supervised CNNs (SFDA); all ResNet-50 backbones														
BYOL	IN-1k self-sup. (SFDA)	–	82.1	91.9	89.8	97.0	83.9	76.4	96.8	85.4	91.5	63.7	85.1	85.8
DeepCluster-v2	IN-1k self-sup. (SFDA)	–	82.4	91.2	90.2	97.2	84.8	77.3	97.0	87.2	90.9	66.5	85.4	86.4
InfoMin	IN-1k self-sup. (SFDA)	–	83.8	80.9	86.9	96.7	70.9	73.5	95.8	78.8	90.9	57.7	81.4	81.6
InsDis	IN-1k self-sup. (SFDA)	–	79.7	77.2	80.2	93.1	69.1	66.4	93.6	76.5	84.6	51.6	76.3	77.1
MoCo-v1	IN-1k self-sup. (SFDA)	–	81.8	79.7	82.2	94.2	71.2	67.4	94.3	77.2	85.3	53.8	77.9	78.6
MoCo-v2	IN-1k self-sup. (SFDA)	–	83.7	82.8	85.5	96.5	71.3	72.6	95.1	77.2	89.1	56.3	78.3	80.8
PCL-v1	IN-1k self-sup. (SFDA)	–	82.2	88.6	87.2	96.4	79.4	73.3	95.6	77.7	88.9	58.4	81.9	82.7
PCL-v2	IN-1k self-sup. (SFDA)	–	83.0	87.5	85.6	96.5	79.8	69.3	95.9	80.3	88.7	58.8	81.8	82.5
SeLa-v2	IN-1k self-sup. (SFDA)	–	85.4	90.5	89.8	96.8	84.4	76.0	96.2	86.4	89.6	65.7	85.5	86.0
SwAV	IN-1k self-sup. (SFDA)	–	83.0	89.5	89.8	96.8	83.8	76.7	97.1	87.2	90.6	66.1	85.1	86.0
Vision transformers (SFDA)														
ViT-B/16	IN-21k→1k (timm)	84.5	78.4	93.5	89.3	98.6	90.0	77.7	98.0	89.0	94.6	68.6	87.9	87.8
PVTv2-B2	IN-1k (timm)	82.0	84.1	93.1	90.6	98.0	88.2	77.2	97.9	88.7	93.9	66.4	86.4	87.7
ViT-S/16	IN-21k→1k (timm)	81.4	73.1	92.7	86.7	97.7	86.6	75.1	96.8	86.3	94.0	64.8	86.6	85.5
PVT-Medium	IN-1k (timm)	81.2	76.7	93.8	87.7	97.9	87.4	77.1	97.4	85.6	94.5	67.2	87.4	86.6
Swin-T	IN-1k (timm)	81.2	81.9	91.9	88.9	97.3	86.0	77.0	97.4	86.7	94.5	65.5	87.5	86.8
PVT-Small	IN-1k (timm)	78.7	75.2	93.0	87.6	97.3	86.2	75.8	97.3	87.0	94.1	65.8	86.6	86.0
ViT-T/16	IN-21k→1k (timm)	75.5	71.3	89.4	82.1	96.5	81.6	71.9	95.5	82.0	91.4	58.4	83.1	82.1
PVT-Tiny	IN-1k (timm)	70.5	69.8	90.0	84.1	94.9	75.3	72.9	95.8	83.8	91.5	61.9	84.6	82.2
DINO ViT-S/16	IN-1k self-sup. (timm/ckpt)	–	72.2	86.8	79.8	98.0	85.7	76.0	96.0	85.7	92.6	64.1	84.8	83.8
MoCo-v3 ViT-S/16	IN-1k self-sup. (timm/ckpt)	–	76.0	89.8	82.2	97.9	85.8	71.9	93.9	82.8	90.4	60.6	81.8	83.0
Mixed hub (ITM)														
ResNet-152	IN-1k (tv V2)	82.3	81.6	97.6	89.1	98.0	88.6	71.4	89.2	88.3	94.8	76.2	84.7	87.2
ResNet-101	IN-1k (tv V2)	81.9	85.8	97.5	88.7	98.1	87.6	71.2	90.9	88.0	94.0	77.2	83.8	87.5
ResNet-50	IN-1k (tv V2)	80.9	84.5	97.4	88.9	97.7	84.6	70.8	91.4	87.1	93.7	74.8	84.2	86.8
EfficientNet-B0	IN-1k (eff._pytorch)	77.7	81.6	97.9	87.2	97.9	87.0	68.9	89.7	85.8	90.7	73.9	81.1	85.6
Inception-v3	IN-1k (tv V1)	77.3	79.6	97.0	84.7	96.9	83.4	66.8	87.1	81.8	92.9	68.1	79.5	83.4
DenseNet-161	IN-1k (tv V1)	77.1	88.2	98.2	89.8	97.8	87.0	72.6	92.0	86.7	93.2	74.1	82.3	87.5
DenseNet-201	IN-1k (tv V1)	76.9	84.0	97.5	89.5	97.7	86.4	73.0	91.1	86.1	93.1	74.1	80.9	86.7
DenseNet-169	IN-1k (tv V1)	75.6	83.8	97.3	88.8	97.8	86.2	70.9	91.6	85.6	92.9	73.8	81.8	86.4
DenseNet-121	IN-1k (tv V1)	74.4	83.9	97.2	88.4	97.4	85.7	69.5	90.4	85.0	91.8	72.5	82.8	85.9
MNASNet-1.0	IN-1k (tv V1)	73.5	75.0	95.6	81.1	96.8	83.4	61.0	76.5	82.9	90.7	70.0	78.8	81.1
ResNet-34	IN-1k (tv V1)	73.3	77.5	96.5	86.1	97.3	85.3	67.5	90.0	85.2	92.7	70.8	81.6	84.6
MobileNet-v2	IN-1k (tv V2)	72.2	76.4	96.0	85.7	96.5	82.2	67.3	90.0	83.7	89.0	71.2	82.5	83.7
GoogLeNet	IN-1k (tv V1)	69.8	83.3	96.5	86.5	97.0	83.4	69.7	88.0	81.4	90.6	69.7	80.5	84.2
ResNet-18	IN-1k (tv V1)	69.8	77.6	95.7	83.6	96.6	82.7	65.6	88.5	79.9	88.5	68.7	76.7	82.2
DINO ViT-S/8	IN-1k self-sup. (timm/ckpt)	–	80.0	97.5	89.3	98.7	90.2	72.0	90.8	90.8	93.1	77.5	85.6	87.8
DINO ViT-B/16	IN-1k self-sup. (timm/ckpt)	–	78.4	98.2	88.4	98.7	90.5	74.3	93.2	87.9	93.0	77.0	84.5	87.6
MAE ViT-B/16	IN-1k self-sup. (timm/ckpt)	–	72.2	97.5	88.2	98.4	87.5	69.0	85.8	87.3	89.8	75.3	82.9	84.9
MAE ViT-L/16	IN-1k self-sup. (timm/ckpt)	–	85.3	98.0	91.2	98.5	91.3	74.3	90.7	90.8	94.7	79.2	88.0	89.3
MoCo-v3 ViT-B/16	IN-1k self-sup. (timm/ckpt)	–	75.7	97.7	88.5	98.7	90.7	72.1	93.6	87.2	89.8	75.9	80.4	86.4
SimMIM ViT-B/16	IN-1k self-sup. (timm/ckpt)	–	68.3	96.2	86.0	98.6	88.8	66.2	83.9	87.9	87.2	74.5	79.6	83.4

Table 7: Per-target statistics: class count, sample count N, and feature-extraction wall-clock (seconds, all models) for each hub.

Target	classes	N	extract Sup. CNN	extract SSL CNN	extract ViT	extract Mixed
Aircraft	100	1,000	25.6	19.5	21.8	51.6
Caltech-101	101	8,677	18.1	15.5	16.0	42.6
Cars	196	16,185	39.1	32.2	33.9	83.2
CIFAR-10	10	60,000	328.7	105.3	106.6	402.6
CIFAR-100	100	60,000	122.2	104.7	106.7	289.5
DTD	47	5,640	12.2	10.3	10.9	28.5
Flowers-102	102	8,189	17.4	14.7	15.4	40.7
Food-101	101	101,000	203.3	174.4	178.9	484.2
Pets	37	7,349	15.6	13.2	13.9	36.6
SUN397	397	39,700	187.8	110.1	106.2	251.9
VOC2007	20	9,963	286.0	21.8	22.6	190.2
mean			114.2	56.5	57.5	172.9

by the ratio of its between-class to within-class variance. SA (Khoba et al., 2025) perturbs the features by spreading each class and attracting overlapping classes, applies the perturbation only when the adaptive shrinkage reaches its floor, as in the original code, and reports SFDA on the perturbed features. The k-NN estimator (Sorkhei et al., 2025) is the 3-fold cross-validated accuracy of a k-NN classifier with $k = 20$.

Likelihood and evidence scores. LEEP (Nguyen et al., 2020) builds the empirical conditional $p(y \mid z)$ between target labels and source-head predictions and reports the mean log-likelihood of the true labels. NCE (Tran et al., 2019) computes the negative conditional entropy $-H(Y \mid Z)$, with Z the source head's top-1 prediction. $\mathcal{N}$LEEP (Li et al., 2021b) replaces the source head with a Gaussian mixture on PCA features that keep 80% of the energy; the paper uses $5C$ components for C classes, which we cap at 200 so that the fit on SUN397 terminates. LogME (You et al., 2021) fits, for each class c, a Bayesian linear regression of the one-vs-rest indicator of c on the raw target features, with prior precision α and noise precision β updated by the fixed-point rule of the paper from $\alpha = \beta = 1$ for at most 11 iterations (relative tolerance 10^{-3} on α/β). It computes the log marginal evidence per sample of each regression, and the score is the average over the classes. PACTran (Ding et al., 2022) reports the PAC-Bayes bound with the Gaussian prior on 64 PCA dimensions, negated. ETran (Gholami et al., 2023) combines its energy term (the mean log-sum-exp of the source logits) and an LDA term as 0.15 energy $+ 0.85$ LDA after min-max normalisation.

Transport and information scores. TransRate (Huang et al., 2022) is the coding-rate difference $R(F) - R(F \mid Y)$ with $\epsilon = 10^{-4}$. TMI (Xu & Kang, 2023) is the sample-weighted mean intra-class feature variance. OTCE (Tan et al., 2021) transports $1{,}500$ source features onto $1{,}500$ target features under the Euclidean cost, where source labels are the source head's predictions on ImageNet validation images. The full OTCE regresses its two terms on auxiliary tasks of known transfer accuracy, which a benchmark without such tasks cannot provide, so we report the auxiliary-free form, the negative conditional entropy of the target labels given the source labels under the transport plan.

Dynamics scores. PED (Li et al., 2023) applies a potential-energy descent that pushes overlapping class clusters apart for at most six steps, with the features standardised by ImageNet feature statistics, and reports SFDA on the remapped features. ITM (Zheng et al., 2025) standardises each feature dimension, maps the features with a linear layer to C outputs and L2-normalises them. For a batch of b samples with normalised outputs $Z \in \mathbb{R}^{b \times C}$ and fixed, L2-normalised one-hot label embeddings E, it applies one step of the implicit update $\tilde{Y} = \big(I_b - \frac{1}{2b} Z Z^\top\big)(Z - E) + E$, where $ZZ^\top$ is computed with gradients stopped, and a second linear layer maps $\tilde{Y}$ to the logits. Training minimises the cross-entropy with AdamW (learning rate 0.005, weight decay 0.01, batch 256, learning rate decaying linearly over a horizon of $10{,}000$ iterations) for 500 iterations on a random 80% of the target samples. Every 100 iterations it measures the accuracy on the remaining 20% with the same forward pass, which also receives the label embeddings, and the score is the best of these five accuracies. LEAD (Hu et al., 2024) follows Algorithm 1 of the paper. It centres the features and divides them by their mean row norm, and takes as initial logits those of a ridge classifier on the

one-hot labels Y, $F_0 = XW$ with $W = (X^\top X + \lambda I)^{-1} X^\top Y$ and $\lambda = 10^{-3}$. A randomly initialised head with one hidden layer of 64 tanh units (Gaussian weights with variances $1/d$ and $1/64$, zero biases) defines, for up to 8 samples of each class c, the empirical neural tangent kernel $\Phi_c = GG^\top$, where each row of G is the gradient of the sum of the C output logits with respect to the head parameters. The logits of the samples of class c then relax towards their labels in closed form, $F_c = \left(1 - e^{-\bar{\lambda}_c t}\right) Y_c + e^{-\bar{\lambda}_c t} F_{0,c}$ with $t = 10$, where $\bar{\lambda}_c$ is the mean eigenvalue of Φ_c divided by the sum of its eigenvalues. The score is the negative mean cross-entropy of $\mathrm{softmax}(F)$ against the labels. ATC (Garg et al., 2022) calibrates a threshold on the source head's negative entropy over ImageNet validation images so that the fraction below it equals the published ImageNet error, and reports the fraction of target samples above that threshold.

Baselines. The parameter count, feature dimension and INB, the published ImageNet top-1 accuracy, are read from model metadata. LDA, the separability half of TE-Ref, is the mean true-class probability of a linear discriminant with a shared identity covariance (Fisher, 1936), fitted on the target features. With μ_c the mean and π_c the share of class c, each sample receives the logits $z_{ic} = x_i^\top \mu_c - \frac{1}{2}\|\mu_c\|^2 + \log \pi_c$, which equal $-\frac{1}{2}\|x_i - \mu_c\|^2 + \log \pi_c$ up to a term shared by all classes, and the score is $\frac{1}{N}\sum_i \mathrm{softmax}(z_i)_{y_i}$, evaluated on the same samples that define the class means. The implementation first estimates a shrunk within-class scatter $\tilde{S}_w$ and a between-class scatter $\tilde{S}_b$, but it takes the discriminant directions from a symmetric eigensolver applied to the non-symmetric matrix $\tilde{S}_w^{-1}\tilde{S}_b$, which returns an orthonormal basis V. Since $VV^\top = I$, the directions cancel and the score is exactly the nearest-class-mean rule above. It reads only target features and labels, so it is defined on every hub. MSP takes, for each target sample, the maximum softmax probability of the source head and averages it over the target samples (Eq. (3)), Entropy the negated mean Shannon entropy of its softmax, and NLL the mean log-probability of the predicted class (Tu et al., 2025), so none of the three reads the target labels. RankMe (Garrido et al., 2023) is the effective rank of the target feature matrix (Eq. (1)). RepVar is the negated sum of the per-dimension variances of the row-normalised features and logits, and reduces to the feature term on checkpoints without a head. LFC is the correlation between the feature Gram matrix and the label agreement matrix. k-means NMI is the normalised mutual information between a k-means clustering with one cluster per class and the true labels. α-weighted (Martin et al., 2021) reshapes every convolutional and linear weight into an (output $\times$ fan-in) matrix, fits a power law to the tail of the eigenvalues of $W^\top W$ by maximum likelihood with the lower cut-off chosen by the Kolmogorov-Smirnov distance, and averages $\alpha_l \log_{10} \lambda_{l,\max}$ over the layers (Eq. (2)), negated so that a smaller value ranks higher.

A.4 Computation time

Tables 8–11 give the absolute wall-clock cost of the platform, one table per hub, in seconds per target. The first row is the shared feature extraction (GPU), paid once per (model, target) and amortised across every feature-based score; the remaining rows are each individual score's own computation time at the full budget, grouped into TE methods, weight-only metrics and trivial baselines, and the last row is their sum. The weight-only and dataset-agnostic scores read no features and so cost effectively nothing; the cost is dominated by the separability and adaptation solvers — PACTran, PED, SFDA, SA, TransRate, ITM and LEAD — and by the largest targets (Food, SUN).

A.5 Full TE-Bench evaluation results

Tables 12–15 give τ_w per target and averaged, at the full sample budget, for every score defined on the complete hub. Scores not defined on every model of a hub are omitted, as in the main text; the weight-only family is set off in its own block, and the trivial baselines below it.

A.6 Distribution shift: metrics and full results

Definitions. For a model m and target $\mathcal{T}$ we compare the target features F_t with the model's source-domain features F_s, both taken from the same backbone so that the two live in the same

Table 8: **Supervised CNNs: wall-clock time per score (seconds, full budget).** Shared feature extraction, then each complete score's own computation time per target, and their sum.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Feature extraction	25.6	18.1	39.1	328.7	122.2	12.2	17.4	203.3	15.6	187.8	286.0	114.2
TE published methods												
PACTran	120.8	113.3	332.6	244.5	528.7	59.2	107.4	846.5	65.1	941.2	68.0	311.6
LEAD	25.7	23.4	75.7	42.2	155.6	9.5	21.9	263.4	10.2	369.3	8.8	91.4
PED	49.4	49.1	71.7	108.8	109.1	38.1	50.3	158.4	43.2	161.0	43.7	80.2
TransRate	36.5	37.0	72.1	15.3	49.8	17.2	36.9	56.8	15.1	149.2	9.3	45.0
SFDA	29.0	30.8	44.6	42.8	54.8	19.5	28.5	76.9	19.0	92.1	16.8	41.3
H-Score$_{\text{reg}}$	18.5	16.8	25.9	76.6	75.8	13.0	16.5	122.1	15.6	52.8	18.5	41.1
PARC	29.4	31.6	32.4	32.7	31.4	31.4	31.4	32.0	31.5	31.8	31.5	31.6
SA	21.5	20.3	25.9	40.2	42.8	19.5	20.4	57.5	19.9	42.2	20.7	30.1
LogME	15.0	14.1	19.7	23.6	33.6	12.2	13.5	48.4	12.7	47.0	13.4	23.0
NCTI	15.4	15.3	18.8	29.1	31.6	13.9	15.2	41.4	15.0	31.5	15.3	22.0
ETran	12.7	12.6	21.6	22.6	29.0	8.1	12.3	41.5	8.0	45.4	7.6	20.1
DISCO	16.8	15.7	22.0	23.1	24.3	12.6	15.3	24.2	14.2	26.8	16.7	19.2
GBC	14.0	13.2	15.9	25.0	26.8	12.0	11.7	33.4	11.9	26.0	13.4	18.5
ITM	15.0	13.5	20.8	13.4	19.1	11.4	13.7	22.4	11.5	40.8	10.3	17.5
H-Score	10.1	9.4	11.4	21.3	21.1	8.9	9.6	30.0	9.3	16.9	9.9	14.4
k-NN	5.7	5.1	14.0	20.6	20.9	2.5	4.6	20.7	3.2	21.2	5.7	11.3
ATC	6.2	6.2	6.8	10.7	10.8	6.0	6.1	14.4	6.1	9.0	6.3	8.1
LEEP	0.7	0.6	1.1	3.7	3.9	0.4	0.6	6.2	0.6	3.3	0.7	2.0
OTCE	1.9	1.9	2.0	1.9	1.9	1.9	2.0	2.0	2.0	2.0	1.9	1.9
TMI	0.2	0.2	0.3	2.1	1.3	0.1	0.2	2.1	0.1	0.9	0.2	0.7
NCE	0.0	0.0	0.1	0.2	0.2	0.0	0.0	0.4	0.0	0.2	0.0	0.1
Baselines evaluated in this paper												
LDA	11.8	11.4	19.1	18.5	23.4	7.9	11.3	42.1	7.0	38.8	6.6	18.0
RankMe	8.6	7.8	10.3	23.7	24.0	6.7	8.3	38.5	7.4	16.9	8.6	14.6
LFC	7.1	7.1	7.3	7.0	7.2	3.6	7.2	7.2	6.0	7.7	7.0	6.8
NLL	0.7	0.6	1.0	3.1	3.1	0.4	0.5	5.2	0.6	2.0	0.6	1.6
Entropy	0.5	0.4	0.8	2.9	2.9	0.3	0.4	4.9	0.4	1.9	0.5	1.4
RepVar	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
α-weighted	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
k-means NMI	28.5	22.3	41.3	9.5	29.6	12.0	24.9	30.4	11.3	64.0	11.9	26.0
MSP	0.6	0.6	0.9	3.1	3.1	0.4	0.6	5.2	0.5	2.1	0.6	1.6
INB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Feature dim	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
#Params	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Score computation	562.6	534.2	1,043.8	949.1	1,496.8	360.8	523.4	2,201.5	382.6	2,527.8	389.9	997.5

Table 9: **Self-supervised CNNs: wall-clock time per score (seconds, full budget).**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Feature extraction	19.4	15.5	32.2	105.2	104.7	10.3	14.7	174.4	13.2	110.0	21.8	56.5
TE published methods												
SA	91.5	83.2	137.2	122.4	159.1	64.1	89.1	240.4	63.7	276.7	63.3	126.4
PED	73.7	74.8	106.5	143.5	147.7	57.4	74.1	213.2	58.3	236.4	59.6	113.2
PACTran	40.6	47.8	101.3	84.9	205.8	25.7	37.8	301.3	30.0	307.1	31.1	110.3
LEAD	29.5	26.3	85.6	51.5	178.1	11.0	25.0	302.6	11.3	418.5	10.6	104.5
TransRate	59.8	61.2	119.1	22.0	76.0	29.4	60.5	86.5	24.0	242.7	14.5	72.3
H-Score$_{\text{reg}}$	32.2	27.6	41.2	119.4	117.6	21.1	26.7	195.0	24.7	83.2	30.4	65.4
SFDA	41.5	41.8	69.4	57.4	75.2	26.9	40.9	104.7	24.9	138.0	23.6	58.6
LogME	22.1	21.2	27.3	34.4	48.0	18.6	20.6	67.0	19.9	62.2	19.9	32.8
PARC	28.1	29.0	28.8	28.5	28.3	29.0	27.6	29.7	30.3	28.7	29.2	28.8
DISCO	24.7	22.5	30.1	32.7	33.6	18.4	22.2	33.2	20.5	35.8	24.1	27.1
NCTI	16.9	15.8	20.1	32.2	34.0	14.6	16.0	44.1	15.3	34.2	16.3	23.6
H-Score	16.1	15.5	17.6	31.2	30.7	14.4	15.2	43.1	14.9	25.1	16.0	21.8
GBC	14.9	14.9	17.9	29.8	29.6	14.0	14.8	38.5	14.0	27.7	14.4	20.9
ITM	15.7	13.3	21.3	12.5	18.0	11.1	13.5	22.1	10.6	36.8	10.4	16.8
k-NN	6.9	6.0	17.2	25.3	25.2	2.9	5.7	25.1	3.8	25.3	6.7	13.6
TMI	0.2	0.2	0.4	2.3	1.4	0.1	0.2	2.4	0.2	1.0	0.3	0.8
Baselines evaluated in this paper												
LDA	20.3	20.3	34.0	25.5	36.6	12.7	20.0	49.5	12.1	67.0	10.6	28.1
RankMe	12.8	11.8	14.9	33.0	33.0	9.6	11.9	52.9	11.0	24.1	12.3	20.7
LFC	7.3	7.3	7.4	7.2	7.3	3.7	7.4	7.4	6.2	7.8	7.3	6.9
RepVar	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
α-weighted	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
k-means NMI	33.7	27.5	53.5	12.9	35.7	14.6	32.0	38.0	15.4	85.0	14.4	33.0
Feature dim	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
#Params	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Score computation	678.1	651.8	1,133.2	1,014.4	1,497.4	447.4	642.0	2,117.2	463.8	2,560.2	465.1	1,061.0

Table 10: **Vision transformers: wall-clock time per score (seconds, full budget).**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Feature extraction	21.8	16.0	33.9	106.6	106.7	10.9	15.4	178.8	13.9	106.2	22.6	57.5
TE published methods												
$\mathcal{N}$LEEP	56.1	42.7	96.8	900.5	1,253.4	30.2	31.1	4,264.5	23.5	561.0	34.3	663.1
PACTran	84.8	80.9	261.2	172.9	404.4	36.7	69.4	646.0	43.4	733.8	43.4	234.3
LEAD	12.5	11.1	34.6	14.4	68.9	4.6	10.7	115.1	4.7	160.3	3.2	40.0
PARC	23.7	23.1	23.4	23.8	22.9	23.3	22.7	23.1	23.9	23.5	24.0	23.4
PED	9.6	9.5	13.6	23.7	23.2	7.4	9.5	33.4	8.7	32.1	9.5	16.4
ITM	11.9	9.9	13.8	11.1	15.1	9.1	9.8	15.4	8.4	23.4	8.6	12.4
SFDA	5.3	5.3	7.9	8.8	11.5	4.3	5.4	16.0	4.2	14.9	4.2	8.0
SA	3.7	3.5	5.4	6.5	9.0	4.5	3.8	13.5	4.8	13.2	3.1	6.5
NCTI	2.4	2.1	3.9	3.3	6.0	3.5	2.3	9.0	3.5	10.1	1.7	4.3
k-NN	2.1	2.3	4.4	6.7	6.4	0.8	1.8	6.5	1.4	6.5	2.1	3.7
LogME	1.8	1.8	2.9	2.9	5.5	1.4	1.6	9.0	1.4	9.1	1.4	3.5
TransRate	2.6	2.6	5.1	2.0	4.0	1.3	2.6	5.1	1.1	10.3	0.8	3.4
DISCO	2.8	2.4	4.2	3.5	4.4	1.5	2.3	4.0	1.8	6.6	2.2	3.2
H-Score$_{\text{reg}}$	1.2	1.1	1.8	5.5	5.7	0.8	1.1	8.9	1.0	4.2	1.2	3.0
GBC	1.0	0.6	1.4	1.5	1.7	2.4	0.5	2.4	2.2	4.7	0.3	1.7
H-Score	0.8	1.0	1.1	2.7	2.9	0.6	0.7	4.4	0.6	2.4	0.7	1.6
TMI	0.1	0.1	0.1	0.5	0.5	0.0	0.1	0.7	0.0	0.5	0.1	0.2
Baselines evaluated in this paper												
LFC	4.8	4.7	4.9	4.6	4.8	2.5	4.7	4.7	4.0	5.2	4.6	4.5
RankMe	1.1	1.1	1.5	4.2	4.1	0.9	1.1	7.0	1.1	2.7	1.2	2.4
LDA	1.0	1.0	1.9	2.6	3.5	0.6	0.9	5.1	0.6	4.9	0.6	2.1
RepVar	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
α-weighted	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
k-means NMI	12.4	9.3	16.4	4.4	11.5	5.0	10.1	12.4	5.0	21.5	5.1	10.3
Feature dim	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
#Params	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Score computation	609.3	402.9	826.4	1,276.3	2,152.4	348.1	328.7	5,466.7	337.4	1,876.5	541.8	1,287.8

Table 11: **Mixed hub: wall-clock time per score (seconds, full budget).**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Feature extraction	51.6	42.6	83.2	402.6	289.5	28.5	40.7	484.2	36.6	251.9	190.2	172.9
TE published methods												
$\mathcal{N}$LEEP	150.2	139.9	177.0	2,574.1	3,035.4	289.1	127.2	10,940.1	109.2	1,215.3	244.9	1,727.5
PACTran	197.0	192.7	682.6	362.1	841.0	103.3	188.2	1,320.8	103.8	1,826.5	115.0	539.4
LEAD	41.6	37.6	119.8	63.9	246.9	14.7	35.5	417.4	15.8	585.1	13.0	144.7
PED	69.9	71.4	127.9	158.2	156.3	56.9	68.7	226.7	60.7	230.0	61.8	117.1
TransRate	48.3	48.4	246.2	21.2	65.0	22.8	48.5	76.4	19.8	197.8	12.5	73.4
SFDA	39.7	41.4	81.4	60.4	78.9	27.5	40.6	108.6	26.4	126.3	24.2	59.6
PARC	53.3	54.8	57.5	58.5	54.7	56.0	55.0	56.2	56.1	55.2	56.2	55.8
H-Score$_{\text{reg}}$	23.8	21.9	33.3	121.9	99.2	16.7	21.1	161.0	20.1	68.2	23.7	55.5
SA	39.2	31.2	88.6	58.7	64.9	31.2	33.7	97.9	29.9	79.4	30.0	53.2
NCTI	20.3	20.6	69.9	39.6	43.9	22.4	21.2	59.3	23.8	47.2	20.2	35.3
LogME	19.2	16.8	33.5	30.3	46.8	15.5	19.2	69.2	15.5	66.4	16.3	31.7
DISCO	23.4	21.4	30.4	84.9	33.6	16.7	21.0	33.2	19.0	38.1	22.4	31.3
ITM	27.1	23.1	35.6	30.9	31.9	19.8	23.4	37.9	19.6	65.4	18.8	30.3
GBC	19.3	18.1	22.2	41.5	35.1	19.8	16.9	45.1	20.3	35.8	18.5	26.6
H-Score	13.1	12.7	15.1	41.6	29.3	11.5	12.4	41.5	12.3	23.1	12.9	20.5
k-NN	8.7	7.8	21.3	31.3	31.5	3.8	7.4	31.2	5.0	31.7	8.7	17.1
TMI	0.3	0.3	0.5	3.1	2.1	0.1	0.2	3.1	0.2	1.5	0.3	1.1
Baselines evaluated in this paper												
LDA	15.9	15.1	25.8	25.8	33.4	10.4	16.2	60.3	9.2	53.4	8.9	24.9
RankMe	11.9	11.1	37.7	33.7	33.8	9.3	11.4	54.7	10.2	24.0	12.0	22.7
LFC	12.2	12.1	12.4	11.9	12.2	6.1	12.2	12.2	10.2	13.1	11.9	11.5
RepVar	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
α-weighted	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
k-means NMI	47.3	35.7	64.8	16.3	46.8	19.8	40.1	46.5	17.7	97.6	18.9	41.0
#Params	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Feature dim	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Score computation	1,531.8	1,378.6	2,847.1	4,274.6	5,740.6	1,318.2	1,184.2	14,578.9	1,055.1	5,787.4	1,806.8	3,773.0

Table 12: Supervised CNNs: full per-target τ_w at the full sample budget.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
ITM	0.321	**0.797**	0.424	0.719	0.746	0.671	**0.939**	0.758	0.818	0.693	0.534	0.674
NCTI	0.438	0.736	0.594	0.856	0.879	0.646	0.457	0.786	0.537	0.732	0.700	0.669
SFDA	0.614	0.742	0.465	0.928	0.845	0.560	0.515	0.827	0.657	0.684	0.415	0.659
SA	0.121	0.510	0.708	0.855	0.852	**0.823**	0.100	0.715	0.723	0.787	0.773	0.633
H-Score	0.532	0.536	0.701	**0.942**	0.805	0.677	0.497	0.581	0.389	0.501	0.397	0.596
k-NN	-0.416	0.468	0.247	0.901	**0.910**	0.652	0.228	0.749	**0.868**	**0.817**	**0.793**	0.565
PACTran	-0.179	0.264	0.298	0.851	0.849	0.745	0.193	0.786	0.816	0.747	0.687	0.551
DISCO	0.167	0.564	0.338	0.716	0.819	0.591	0.579	0.431	0.303	0.727	0.773	0.546
LEEP	-0.194	0.565	0.537	0.744	0.788	0.670	-0.066	0.602	0.743	0.720	0.764	0.534
PARC	-0.190	0.434	0.481	0.924	0.863	0.639	0.223	0.713	-0.130	0.740	0.761	0.496
ETran	-0.355	0.517	0.053	0.811	0.846	0.331	0.213	0.637	0.671	0.755	0.713	0.472
GBC	-0.231	0.521	0.394	0.937	0.877	0.238	0.171	0.728	0.257	0.653	0.314	0.442
LogME	0.196	0.463	0.453	0.852	0.692	0.603	0.022	0.355	0.411	0.411	0.377	0.440
LEAD	0.448	0.415	0.535	0.845	0.694	0.450	0.019	0.355	0.315	0.349	0.316	0.431
H-Score$_{\text{reg}}$	0.309	0.415	0.472	0.845	0.654	0.450	0.019	0.293	0.315	0.321	0.293	0.399
PED	0.367	0.775	0.056	0.629	0.058	0.316	0.626	0.452	0.643	0.068	0.223	0.383
OTCE	0.207	0.153	0.607	0.499	0.540	0.584	-0.254	0.237	0.582	0.494	0.480	0.375
NCE	-0.150	0.534	0.354	0.729	0.763	0.149	-0.519	0.441	0.631	0.663	0.376	0.361
TMI	-0.182	0.315	0.104	0.399	0.428	0.447	0.772	0.551	0.173	0.586	0.322	0.356
TransRate	0.432	0.269	0.434	0.514	0.173	0.196	-0.067	-0.072	0.391	0.088	0.168	0.230
ATC	-0.171	-0.163	0.214	0.546	0.629	0.015	0.054	-0.328	-0.226	0.276	-0.363	0.044
Baselines evaluated in this paper												
α-weighted	0.643	0.766	0.643	0.706	0.747	0.757	0.723	**0.931**	0.713	0.780	0.522	**0.721**
INB	**0.732**	0.630	**0.732**	0.775	0.759	0.736	0.252	0.540	0.725	0.666	0.649	0.654
MSP	0.492	0.587	0.471	0.566	0.693	0.583	0.427	0.462	0.685	0.712	0.685	0.578
NLL	0.542	0.437	0.442	0.566	0.693	0.583	0.448	0.464	0.685	0.686	0.750	0.573
#Params	0.652	0.520	0.652	0.699	0.681	0.650	0.080	0.392	0.679	0.550	0.618	0.561
k-means NMI	-0.405	0.736	0.673	0.814	0.838	0.434	0.071	0.635	0.531	0.722	0.707	0.523
LDA	-0.212	0.641	0.074	0.811	0.846	0.364	0.191	0.754	0.671	0.774	0.699	0.510
Entropy	0.460	0.229	0.518	0.626	0.628	0.563	0.154	0.606	0.353	0.686	0.488	0.483
Feature dim	0.530	0.432	0.530	0.573	0.560	0.539	0.101	0.328	0.533	0.468	0.600	0.472
LFC	-0.356	0.025	0.103	0.549	0.553	-0.088	0.584	0.244	0.142	0.155	0.423	0.212
RepVar	0.657	-0.065	0.524	0.119	-0.058	0.130	-0.136	0.167	0.313	0.313	0.314	0.207
RankMe	-0.147	0.065	-0.129	0.090	0.057	0.096	-0.375	-0.240	-0.093	-0.183	-0.021	-0.080
TE-Ref	0.790	0.610	0.800	0.750	0.820	0.810	0.180	0.540	0.820	0.680	0.760	0.686

Table 13: Self-supervised CNNs: full per-target τ_w. All head-dependent scores are absent.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
PACTran	0.063	0.770	**0.801**	0.846	0.735	0.889	0.850	0.837	0.621	**0.916**	0.690	0.729
SA	0.063	0.770	0.796	0.747	0.781	0.889	**0.870**	**0.922**	0.621	**0.916**	0.642	0.729
TransRate	0.080	0.778	0.702	**0.892**	0.757	0.886	0.832	0.855	0.605	0.873	0.703	0.724
SFDA	0.232	0.653	0.683	0.792	0.789	0.904	0.814	0.837	0.600	0.831	0.621	0.705
NCTI	0.036	**0.811**	0.648	0.742	0.757	**0.928**	0.686	0.918	0.805	0.888	0.501	0.702
TMI	0.063	0.658	0.796	0.657	0.769	0.886	**0.870**	0.898	0.491	0.873	0.650	0.692
PED	0.232	0.636	0.690	0.804	0.720	0.853	0.770	0.686	0.470	0.495	0.605	0.633
ITM	-0.118	0.764	0.543	0.787	0.668	0.737	0.740	0.791	0.645	0.827	0.371	0.614
H-Score$_{\text{reg}}$	-0.078	0.572	0.610	0.582	0.505	0.832	0.678	0.646	0.762	0.286	0.626	0.547
k-NN	-0.259	0.694	0.536	0.626	0.688	0.512	0.493	0.653	**0.817**	0.880	-0.088	0.505
LogME	-0.031	0.151	0.627	0.611	0.354	0.743	0.657	0.646	0.712	0.260	0.229	0.451
LEAD	-0.202	0.247	0.591	0.611	0.382	0.734	0.627	0.646	0.762	0.286	0.229	0.447
H-Score	-0.203	0.247	0.573	0.611	0.382	0.715	0.627	0.646	0.762	0.286	0.229	0.443
PARC	-0.290	-0.013	0.634	0.543	0.100	0.636	0.609	0.716	0.439	0.760	-0.209	0.357
GBC	-0.214	-0.066	0.699	0.118	-0.115	0.059	0.509	0.294	0.690	0.100	0.058	0.194
DISCO	-0.390	-0.014	0.323	0.223	-0.261	0.459	-0.070	-0.013	0.373	-0.000	-0.304	0.030
Baselines evaluated in this paper												
LDA	0.063	0.770	0.796	0.747	**0.835**	0.889	**0.870**	**0.922**	0.621	**0.916**	0.642	**0.734**
RepVar	**0.709**	0.401	0.626	0.527	0.497	0.578	0.444	0.508	0.199	0.599	**0.762**	0.532
k-means NMI	-0.373	0.767	0.685	0.699	0.660	0.518	0.413	0.494	0.628	0.880	-0.320	0.459
RankMe	-0.265	0.547	0.577	0.552	0.302	0.728	0.500	0.576	0.502	0.535	0.453	0.455
LFC	-0.539	0.100	0.599	-0.046	0.086	0.103	0.251	0.523	0.290	-0.358	-0.303	0.064
α-weighted	-0.021	-0.555	-0.733	-0.563	-0.716	-0.597	-0.586	-0.756	-0.208	-0.737	-0.708	-0.562
TE-Ref	0.060	0.770	0.800	0.750	0.830	0.890	0.870	0.920	0.620	0.920	0.640	0.734

Table 14: Vision transformers: full per-target τ_w at the full sample budget.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
H-Score	0.461	0.810	**0.668**	0.644	0.752	0.749	0.581	0.616	0.569	**0.821**	**0.929**	0.691
LEAD	0.461	0.734	**0.668**	0.644	0.752	0.749	0.581	0.616	0.569	**0.821**	0.905	0.682
H-Score$_{\text{reg}}$	0.461	0.810	**0.668**	0.644	0.752	0.749	0.581	0.616	0.569	**0.821**	0.814	0.681
LogME	0.263	0.793	0.462	0.593	0.752	0.729	0.597	0.616	0.506	0.769	0.738	0.620
SFDA	0.272	0.452	0.417	0.593	0.696	**0.846**	0.554	0.607	0.657	0.530	0.791	0.583
DISCO	0.584	**0.841**	0.567	0.568	**0.798**	0.301	0.393	0.557	0.392	0.591	0.713	0.573
TransRate	0.235	0.279	0.362	0.264	0.658	0.777	0.667	0.606	0.868	0.602	0.905	0.566
GBC	0.433	0.385	0.582	0.473	0.770	0.389	0.461	0.579	0.445	0.804	0.456	0.525
PED	0.347	0.343	0.042	0.515	0.490	0.786	0.621	0.441	0.657	0.381	0.767	0.490
ITM	0.179	0.586	0.108	0.593	0.696	0.438	0.554	0.572	0.423	0.544	0.635	0.484
NCTI	0.171	0.808	-0.274	0.593	0.712	0.382	0.367	0.505	0.445	0.544	0.678	0.448
$\mathcal{N}$LEEP	-0.196	0.711	0.603	0.539	0.790	0.101	0.384	0.369	0.278	0.617	0.635	0.439
PACTran	-0.183	0.203	0.028	0.621	0.736	0.274	0.405	0.572	0.543	0.617	0.850	0.424
SA	0.082	-0.202	0.603	0.593	0.736	0.072	0.405	0.572	0.570	0.591	0.548	0.416
PARC	0.016	-0.269	0.582	0.432	0.770	0.309	0.459	0.579	0.237	0.728	0.627	0.406
k-NN	-0.088	0.328	0.434	0.539	0.686	0.215	0.291	0.557	0.363	0.656	0.474	0.405
TMI	-0.285	-0.351	-0.289	0.478	0.286	0.078	0.091	0.159	0.038	0.011	0.016	0.021
Baselines evaluated in this paper												
Feature dim	**0.660**	0.514	0.650	0.502	0.666	0.750	0.816	0.724	0.856	0.736	0.845	**0.702**
#Params	0.524	0.619	0.541	**0.653**	0.772	0.776	0.705	0.560	0.738	0.804	0.762	0.678
RankMe	0.465	0.302	0.640	0.226	0.440	0.643	**0.854**	**0.768**	**0.892**	0.577	0.727	0.594
k-means NMI	-0.267	0.681	0.603	0.473	0.663	0.101	0.405	0.375	0.478	0.576	0.628	0.429
LDA	-0.109	-0.019	0.296	0.593	0.736	0.197	0.405	0.557	0.752	0.423	0.619	0.405
LFC	0.360	0.066	0.649	0.423	0.672	0.067	0.103	0.420	0.469	0.402	0.096	0.339
α-weighted	-0.082	-0.088	-0.069	0.380	0.078	0.323	0.192	0.211	0.031	0.197	0.018	0.108
RepVar	0.206	-0.236	-0.681	0.056	-0.286	-0.395	-0.325	-0.137	-0.269	-0.243	-0.216	-0.230
TE-Ref	0.270	0.650	0.570	0.610	0.760	0.850	0.740	0.570	0.820	0.830	0.840	0.682

Table 15: Mixed hub: full per-target τ_w at the full sample budget.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
ITM	0.094	**0.589**	0.289	0.540	0.507	0.639	0.559	0.434	0.413	**0.683**	**0.640**	**0.490**
NCTI	0.217	0.428	0.175	0.449	0.462	**0.788**	0.365	0.421	**0.677**	0.530	0.411	0.448
DISCO	0.243	0.425	0.047	0.362	0.226	0.592	**0.595**	0.545	0.414	0.377	0.164	0.363
$\mathcal{N}$LEEP	0.068	0.309	0.082	0.483	0.400	0.424	0.464	0.474	0.350	0.468	0.315	0.349
k-NN	0.152	0.439	0.106	0.523	0.410	0.365	0.463	0.409	0.159	0.410	0.319	0.341
PACTran	0.104	0.361	0.227	0.529	0.210	0.458	0.156	0.493	0.298	0.403	0.375	0.329
GBC	0.148	0.385	0.078	0.406	0.276	0.299	0.462	**0.559**	0.345	0.298	0.336	0.327
PARC	-0.018	0.350	0.169	0.350	0.275	0.572	0.443	0.519	0.034	0.380	0.333	0.310
LogME	0.110	0.265	0.001	0.438	0.334	0.309	0.223	0.150	0.484	0.302	0.359	0.271
SA	0.215	0.399	0.015	0.185	0.165	0.379	0.118	0.381	0.281	0.353	0.304	0.254
H-Score	0.381	0.228	0.234	0.384	0.250	0.130	0.140	0.004	0.517	0.180	0.244	0.245
H-Score$_{\text{reg}}$	0.156	0.259	0.012	0.429	0.311	0.114	-0.026	0.169	0.492	0.222	0.300	0.222
SFDA	0.276	0.170	0.146	0.374	0.260	0.002	0.166	0.246	0.550	0.017	0.229	0.221
LEAD	0.210	0.243	-0.133	0.373	0.322	0.141	0.106	0.250	0.328	0.256	0.253	0.214
TMI	0.112	0.199	0.161	0.213	-0.013	0.367	0.252	0.231	0.167	0.115	0.191	0.182
PED	0.187	0.178	0.291	0.490	0.221	0.025	0.106	0.143	0.213	-0.061	0.025	0.165
TransRate	0.075	0.059	-0.096	-0.114	-0.176	0.053	-0.153	-0.265	0.461	0.013	0.074	-0.006
Baselines evaluated in this paper												
#Params	0.132	0.490	**0.409**	**0.641**	**0.770**	0.528	0.299	0.519	0.394	0.605	0.468	0.478
LFC	0.187	0.295	-0.017	0.517	0.219	0.410	0.515	0.418	0.391	0.366	0.372	0.334
k-means NMI	0.087	0.357	0.072	0.434	0.385	0.382	0.385	0.481	0.314	0.466	-0.047	0.302
LDA	0.290	0.467	0.209	0.222	0.168	0.561	0.148	0.320	0.277	0.289	0.289	0.295
Feature dim	**0.649**	0.338	0.332	-0.229	-0.106	0.113	0.139	-0.078	0.496	-0.032	0.020	0.149
α-weighted	0.381	0.126	0.243	-0.158	-0.222	0.255	0.142	-0.013	0.180	-0.153	-0.116	0.060
RankMe	0.271	-0.004	0.044	-0.090	-0.138	0.016	-0.117	-0.101	0.503	-0.037	0.026	0.034
RepVar	-0.071	-0.015	0.176	0.074	0.348	-0.029	-0.034	0.079	-0.247	0.118	-0.040	0.033
TE-Ref	0.210	0.540	0.640	0.520	0.710	0.640	0.210	0.730	0.720	0.740	0.660	0.572

space. With μ, σ the per-dimension mean and standard deviation of a subsample of each,

$$\text{mean-gap} = \|\mu_t - \mu_s\|_2, \qquad \text{cos-gap} = 1 - \frac{\mu_t^\top \mu_s}{\|\mu_t\| \, \|\mu_s\|}, \tag{9}$$

$$\text{FID} = \|\mu_t - \mu_s\|_2^2 + \|\sigma_t - \sigma_s\|_2^2, \qquad \text{CKA} = \frac{\|F_t^\top F_s\|_F^2}{\|F_t^\top F_t\|_F \, \|F_s^\top F_s\|_F}, \tag{10}$$

the last two being a diagonal-Gaussian Fréchet distance and linear centred kernel alignment (higher CKA means *less* shift). All four are computed per (model, target) pair; the split into low- and high-shift halves in Section 6 uses the mean-gap at the per-hub median.

Table 16: Mean shift statistics per (hub, target), averaged over each hub's models — the quantity used to split targets into low- and high-shift halves.

Hub	Target	mean-gap	cos-gap	FID	CKA
Supervised CNNs	aircraft	14.10	0.263	344.2	0.176
	caltech101	5.38	0.048	61.0	0.268
	cars	11.22	0.173	231.8	0.196
	cifar10	13.76	0.233	261.7	0.238
	cifar100	12.72	0.224	225.2	0.260
	dtd	7.13	0.083	112.0	0.288
	flowers	12.39	0.217	262.7	0.218
	food	10.02	0.137	175.1	0.272
	pets	5.65	0.048	79.4	0.217
	sun397	5.84	0.051	76.1	0.268
	voc2007	4.01	0.024	46.7	0.293
Self-Supervised CNNs	aircraft	2.02	0.387	122.3	0.085
	caltech101	0.72	0.140	3.1	0.233
	cars	1.39	0.328	7.2	0.157
	cifar10	2.15	0.446	12.9	0.162
	cifar100	2.02	0.439	11.8	0.192
	dtd	0.88	0.209	6.1	0.231
	flowers	1.43	0.356	8.1	0.164
	food	1.19	0.282	5.8	0.196
	pets	0.83	0.172	3.4	0.200
	sun397	0.67	0.151	3.0	0.215
	voc2007	0.48	0.086	2.1	0.253
Vision Transformers	aircraft	27.01	0.593	1815.6	0.144
	caltech101	9.69	0.323	200.1	0.248
	cars	24.09	0.606	1309.5	0.159
	cifar10	22.11	0.636	1024.0	0.226
	cifar100	20.87	0.629	910.5	0.260
	dtd	12.06	0.482	297.2	0.275
	flowers	24.28	0.582	1146.0	0.214
	food	19.89	0.526	807.6	0.238
	pets	12.55	0.391	380.2	0.204
	sun397	11.21	0.349	263.4	0.272
	voc2007	8.14	0.290	146.8	0.288
Mixed hub	aircraft	14.26	0.390	655.8	0.149
	caltech101	5.23	0.125	89.5	0.245
	cars	11.98	0.288	459.5	0.184
	cifar10	12.90	0.337	380.6	0.219
	cifar100	12.17	0.331	353.9	0.245
	dtd	6.67	0.216	148.1	0.269
	flowers	12.51	0.328	430.3	0.197
	food	10.31	0.256	298.3	0.247
	pets	6.01	0.139	133.1	0.200
	sun397	5.72	0.115	104.6	0.256
	voc2007	4.11	0.072	64.6	0.270

Tables 17–20 report each score's τ_w on the low- and high-shift halves, and the degradation between them, for every score defined on the complete hub.

A.7 Data scarcity: sample-budget sweep

Tables 21–23 give the full version of Table 4: τ_w averaged over the 11 targets for *every* score defined on the complete hub, at every sample budget from three samples per class upward. As in the main text a cell is filled only when the score is defined on every model of the hub at that budget. The combined hub is reported at the full budget only (Section 6) and so does not appear in this sweep.

Table 17: Supervised CNNs: τ_w on the low- and high-shift target halves, and degradation.

Score	low-shift	high-shift	degradation
k-NN	0.724	0.374	0.350
LEEP	0.677	0.362	0.315
LDA	0.650	0.342	0.309
$\mathcal{N}$LEEP	0.691	0.390	0.301
ETran	0.604	0.314	0.290
PACTran	0.674	0.402	0.272
NCE	0.466	0.235	0.231
k-means NMI	0.627	0.398	0.229
SA	0.728	0.511	0.217
OTCE	0.422	0.320	0.102
TMI	0.399	0.304	0.095
MSP	0.619	0.530	0.089
ITM	0.712	0.630	0.082
PARC	0.526	0.460	0.066
PED	0.413	0.347	0.066
NLL	0.601	0.538	0.062
α-weighted	0.745	0.692	0.053
NCTI	0.690	0.645	0.045
LEAD	0.425	0.382	0.043
DISCO	0.565	0.524	0.041
RankMe	-0.063	-0.101	0.038
Feature dim	0.483	0.459	0.024
GBC	0.452	0.430	0.022
#Params	0.568	0.553	0.015
Entropy	0.487	0.477	0.010
INB	0.658	0.650	0.008
LogME	0.437	0.443	-0.006
RepVar	0.196	0.221	-0.026
SFDA	0.647	0.673	-0.026
H-Score$_{\text{reg}}$	0.348	0.460	-0.112
TransRate	0.173	0.297	-0.124
LFC	0.150	0.286	-0.136
H-Score	0.513	0.696	-0.182
ATC	-0.132	0.254	-0.386

Table 18: Self-supervised CNNs: low/high-shift τ_w and degradation.

Score	low-shift	high-shift	degradation
$\mathcal{N}$LEEP	0.564	0.286	0.278
NCTI	0.809	0.574	0.234
RankMe	0.557	0.333	0.224
ITM	0.689	0.524	0.165
H-Score$_{\text{reg}}$	0.621	0.459	0.161
k-NN	0.578	0.417	0.161
LDA	0.793	0.662	0.131
TransRate	0.783	0.652	0.131
PACTran	0.787	0.659	0.128
SA	0.725	0.602	0.123
DISCO	0.083	-0.035	0.118
TMI	0.743	0.631	0.112
H-Score	0.481	0.398	0.083
SFDA	0.741	0.662	0.079
k-means NMI	0.494	0.417	0.077
PARC	0.388	0.319	0.069
LogME	0.457	0.444	0.013
GBC	0.189	0.199	-0.010
LFC	0.059	0.071	-0.012
PED	0.624	0.643	-0.019
LEAD	0.053	0.092	-0.040
RepVar	0.508	0.561	-0.053
α-weighted	-0.593	-0.524	-0.070

A.7.1 Per-target results at three samples per class

Table 4 aggregates over targets. Here we give the per-target breakdown at the lowest sample budget we report, three labelled samples per class, but now for *every* compatible score rather than a top five, restricted in each table to the scores defined on the complete hub at that budget. The mean-over-targets column reproduces the corresponding column of Table 4, the scores are grouped into the two families used throughout, and **bold** marks the best score in each column. A dash marks a target on

Table 19: Vision transformers: low/high-shift τ_w and degradation.

Score	low-shift	high-shift	degradation
NCTI	0.560	0.314	0.247
TransRate	0.673	0.437	0.235
PACTran	0.510	0.321	0.189
PED	0.563	0.403	0.160
LogME	0.692	0.533	0.158
SFDA	0.647	0.506	0.141
H-Score	0.749	0.621	0.128
RankMe	0.651	0.525	0.126
H-Score$_{\text{reg}}$	0.730	0.621	0.109
ITM	0.533	0.426	0.107
k-means NMI	0.473	0.375	0.098
Feature dim	0.738	0.659	0.079
#Params	0.710	0.639	0.071
k-NN	0.432	0.372	0.060
SA	0.535	0.494	0.041
LDA	0.422	0.384	0.037
LEAD	0.641	0.607	0.034
$\mathcal{N}$LEEP	0.452	0.424	0.028
α-weighted	0.115	0.100	0.015
DISCO	0.566	0.582	-0.016
GBC	0.510	0.544	-0.034
RepVar	-0.249	-0.206	-0.044
TMI	-0.008	0.056	-0.065
PARC	0.368	0.452	-0.083
LFC	0.253	0.442	-0.188

Table 20: Mixed hub: low/high-shift τ_w and degradation.

Score	low-shift	high-shift	degradation
NCTI	0.543	0.334	0.209
SA	0.526	0.340	0.186
ITM	0.566	0.398	0.168
LDA	0.367	0.208	0.160
TransRate	0.066	-0.093	0.158
PACTran	0.398	0.245	0.153
DISCO	0.419	0.295	0.125
PARC	0.365	0.244	0.121
GBC	0.370	0.274	0.096
LFC	0.375	0.284	0.091
$\mathcal{N}$LEEP	0.390	0.300	0.090
LogME	0.311	0.221	0.090
H-Score$_{\text{reg}}$	0.259	0.176	0.083
RankMe	0.067	-0.006	0.074
LEAD	0.245	0.176	0.069
TMI	0.212	0.145	0.067
k-means NMI	0.326	0.273	0.053
#Params	0.501	0.450	0.050
k-NN	0.350	0.331	0.020
Feature dim	0.143	0.157	-0.014
α-weighted	0.046	0.077	-0.031
SFDA	0.202	0.244	-0.042
H-Score	0.217	0.278	-0.061
RepVar	-0.022	0.099	-0.121
PED	0.087	0.259	-0.172

which the score is undefined at this budget, typically a singular class covariance; the combined hub is reported at the full budget only (Section 6).

A.7.2 Robustness to the sample budget

The full sample-budget sweeps (Tables 21 to 23) report, for every score, its target-averaged τ_w at each labelled-samples-per-class budget. Here we summarise how much that number moves *across* the budget, that is, a score's robustness to how many labels the practitioner has. For each score and hub we take the target-averaged τ_w at every budget from three samples per class to the full set at which the score is defined, and report its mean, standard deviation, minimum and maximum (Table 27). A small standard deviation and a high minimum mean the score's quality barely depends on the sample budget; a large standard deviation means a single full-budget number is a poor guide to the few-shot regime. Data-free scores are budget-invariant by construction and so have zero vari-

Table 21: Supervised CNNs: full sample-budget sweep, all complete scores.

Score	3	4	5	10	20	50	100	all
TE published methods								
ITM	0.310	0.350	0.376	0.564	0.497	0.642	0.656	0.674
NCTI	0.194	0.314	0.506	0.454	0.641	0.681	0.681	0.669
SFDA	0.201	0.201	0.281	0.241	0.454	0.457	0.588	0.659
SA	0.229	0.388	0.454	0.463	0.546	0.602	0.584	0.633
H-Score	0.163	0.048	0.235	0.229	0.321	0.505	0.495	0.596
k-NN	0.137	0.175	0.205	0.487	0.485	0.539	0.560	0.565
DISCO	0.477	0.535	0.530	0.530	0.535	0.545	0.531	0.546
LEEP	0.387	0.428	0.420	0.452	0.499	0.510	0.534	0.534
PARC	0.273	0.234	0.361	0.479	0.426	0.472	0.508	0.496
ETran	0.040	0.061	-0.048	0.143	0.269	0.401	0.465	0.472
GBC	0.145	0.169	0.267	0.274	0.408	0.472	0.434	0.442
LogME	0.549	0.551	0.569	0.513	0.437	0.451	0.437	0.440
LEAD	0.331	0.332	0.369	0.307	0.297	0.342	0.381	0.431
H-Score$_{\text{reg}}$	0.083	0.085	0.060	0.112	0.113	0.244	0.354	0.399
PED	0.201	0.201	0.361	0.334	0.313	0.388	0.400	0.383
OTCE	0.405	0.394	0.395	0.406	0.403	0.377	0.375	0.375
NCE	0.192	0.202	0.156	0.253	0.212	0.324	0.389	0.361
TMI	0.369	0.351	0.358	0.344	0.349	0.349	0.355	0.356
TransRate	0.320	0.310	0.274	0.117	0.170	0.202	0.211	0.230
ATC	0.089	0.063	0.011	0.068	0.023	0.072	0.050	0.044
Baselines evaluated in this paper								
α-weighted	0.721	0.721	0.721	0.721	0.721	0.721	0.721	0.721
INB	0.654	0.654	0.654	0.654	0.654	0.654	0.654	0.654
MSP	0.581	0.558	0.569	0.560	0.568	0.583	0.572	0.578
NLL	0.542	0.521	0.529	0.536	0.551	0.569	0.571	0.573
#Params	0.561	0.561	0.561	0.561	0.561	0.561	0.561	0.561
k-means NMI	0.343	0.425	0.288	0.465	0.507	0.540	0.574	0.523
LDA	0.040	0.061	0.067	0.166	0.259	0.430	0.486	0.510
Entropy	0.508	0.471	0.470	0.468	0.499	0.472	0.457	0.483
Feature dim	0.472	0.472	0.472	0.472	0.472	0.472	0.472	0.472
LFC	0.025	0.058	0.092	0.165	0.149	0.173	0.170	0.212
RepVar	0.159	0.174	0.231	0.187	0.199	0.206	0.240	0.207
RankMe	-0.212	-0.221	-0.237	-0.303	-0.233	-0.170	-0.120	-0.080

Table 22: Self-supervised CNNs: full sample-budget sweep, all complete scores.

Score	3	4	5	10	20	50	100	all
TE published methods								
SA	0.561	0.647	0.620	0.683	0.731	0.734	0.695	0.729
TransRate	0.711	0.713	0.713	0.713	0.712	0.712	0.718	0.724
SFDA	0.576	0.661	0.632	0.695	0.706	0.701	0.703	0.705
NCTI	0.475	0.527	0.525	0.645	0.668	0.706	0.716	0.702
TMI	0.690	0.694	0.699	0.699	0.702	0.697	0.702	0.692
PED	0.579	0.629	0.614	0.660	0.643	0.637	0.645	0.633
ITM	0.290	0.330	0.385	0.470	0.444	0.497	0.575	0.614
H-Score$_{\text{reg}}$	0.458	0.485	0.492	0.555	0.595	0.583	0.544	0.547
k-NN	0.275	0.231	0.358	0.400	0.441	0.518	0.518	0.505
LogME	0.743	0.738	0.723	0.635	0.605	0.525	0.518	0.451
H-Score	-0.013	-0.027	-0.046	-0.043	0.161	0.314	0.448	0.443
PARC	0.380	0.405	0.336	0.384	0.467	0.490	0.502	0.357
GBC	0.356	0.439	0.365	0.260	0.227	0.202	0.187	0.194
LEAD	-0.013	0.009	-0.003	-0.060	0.157	0.524	0.520	0.447
DISCO	0.097	0.028	-0.030	-0.014	0.047	0.019	0.024	0.030
Baselines evaluated in this paper								
LDA	0.718	0.719	0.717	0.729	0.731	0.734	0.734	0.734
RepVar	0.533	0.532	0.537	0.526	0.536	0.531	0.536	0.532
k-means NMI	0.148	0.142	0.242	0.203	0.369	0.487	0.450	0.459
RankMe	0.200	0.219	0.217	0.292	0.381	0.444	0.467	0.455
LFC	0.226	0.181	0.169	0.154	0.110	0.111	0.100	0.064
α-weighted	–	–	–	–	–	–	–	-0.562
Feature dim	–	–	–	–	–	–	–	–
#Params	–	–	–	–	–	–	–	–

Table 23: Vision transformers: full sample-budget sweep, all complete scores.

Score	3	4	5	10	20	50	100	all
TE published methods								
H-Score	0.296	0.276	0.386	0.576	0.598	0.612	0.677	0.691
H-Score$_{\text{reg}}$	0.532	0.530	0.547	0.579	0.588	0.625	0.669	0.681
LEAD	0.508	0.539	0.556	0.623	0.627	0.660	0.657	0.682
LogME	0.691	0.680	0.693	0.711	0.699	0.629	0.625	0.620
SFDA	0.323	0.437	0.460	0.488	0.564	0.592	0.582	0.583
DISCO	0.501	0.493	0.487	0.537	0.560	0.576	0.570	0.573
TransRate	0.312	0.331	0.348	0.442	0.522	0.549	0.563	0.566
GBC	0.197	0.339	0.503	0.521	0.562	0.538	0.539	0.525
PED	0.323	0.429	0.441	0.457	0.525	0.513	0.497	0.490
ITM	0.525	0.558	0.623	0.502	0.439	0.407	0.409	0.484
NCTI	0.209	0.286	0.210	0.384	0.436	0.448	0.466	0.448
SA	0.179	0.192	0.276	0.372	0.508	0.563	0.550	0.416
PARC	0.393	0.530	0.526	0.510	0.529	0.513	0.513	0.406
k-NN	0.362	0.224	0.289	0.379	0.375	0.381	0.404	0.405
TMI	0.032	0.028	0.025	0.021	0.025	0.021	0.021	0.021
Baselines evaluated in this paper								
Feature dim	0.702	0.702	0.702	0.702	0.702	0.702	0.702	0.702
#Params	0.678	0.678	0.678	0.678	0.678	0.678	0.678	0.678
RankMe	0.437	0.478	0.534	0.569	0.583	0.588	0.598	0.594
k-means NMI	0.356	0.401	0.394	0.426	0.449	0.461	0.437	0.429
LDA	0.159	0.216	0.219	0.346	0.305	0.388	0.429	0.405
LFC	0.308	0.330	0.341	0.400	0.400	0.429	0.419	0.339
α-weighted	–	–	–	–	–	–	–	0.108
RepVar	-0.213	-0.209	-0.215	-0.230	-0.225	-0.210	-0.218	-0.230

Table 24: **Supervised CNNs: per-target τ_w at 3 labelled samples per class.**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
LogME	0.681	0.301	0.617	0.685	0.638	0.613	0.221	0.521	0.569	0.477	**0.716**	0.549
DISCO	0.273	0.431	0.330	0.192	0.753	0.460	0.552	0.375	0.525	0.708	0.647	0.477
OTCE	0.508	0.153	0.613	0.370	0.517	0.614	-0.278	0.190	0.566	0.536	0.664	0.405
LEEP	-0.698	0.397	0.378	0.658	**0.869**	0.441	-0.066	0.601	0.524	0.636	0.515	0.387
TMI	-0.182	0.315	0.104	0.451	0.428	0.484	**0.772**	0.551	0.226	0.533	0.381	0.369
LEAD	0.318	0.264	0.248	0.631	0.527	0.295	0.184	0.297	0.362	0.000	0.509	0.331
TransRate	-0.060	0.313	0.046	0.451	0.450	0.408	0.712	0.507	0.234	0.078	0.381	0.320
ITM	0.014	0.517	-0.059	0.683	0.779	0.074	0.490	-0.011	0.100	0.576	0.249	0.310
PARC	-0.029	0.219	0.366	0.064	0.483	0.133	0.311	0.413	-0.170	0.677	0.540	0.273
SA	-0.416	-0.004	0.730	–	0.299	0.204	0.056	0.043	0.543	0.608	–	0.229
PED	–	–	–	–	–	–	–	–	–	0.201	–	0.201
SFDA	–	–	–	–	–	–	–	–	–	0.201	–	0.201
NCTI	-0.474	0.597	0.485	-0.397	0.669	-0.239	0.617	0.507	0.228	0.689	-0.546	0.194
NCE	-0.570	0.284	0.030	0.367	0.411	0.660	-0.418	0.317	0.382	0.371	0.281	0.192
H-Score	0.086	0.294	0.040	0.155	0.267	0.276	0.030	0.398	-0.113	0.182	0.182	0.163
GBC	-0.063	0.224	-0.060	0.129	0.026	-0.038	0.114	-0.077	0.462	0.281	0.600	0.145
k-NN	-0.241	0.662	0.344	–	0.526	-0.506	0.233	-0.034	0.027	0.496	-0.135	0.137
ATC	0.033	-0.090	0.287	-0.005	0.787	0.111	0.336	0.107	-0.135	-0.097	-0.359	0.089
H-Score$_{\text{reg}}$	0.010	0.536	0.326	-0.152	0.110	-0.103	0.171	0.415	-0.065	0.015	-0.349	0.083
ETran	-0.462	0.259	0.035	0.485	-0.145	0.001	0.315	0.045	0.043	0.107	-0.246	0.040
Baselines evaluated in this paper												
α-weighted	0.643	**0.766**	0.643	0.706	0.747	**0.757**	0.723	**0.931**	0.713	**0.780**	0.522	**0.721**
INB	**0.732**	0.630	**0.732**	**0.775**	0.759	0.736	0.252	0.540	**0.725**	0.666	0.649	0.654
MSP	0.728	0.425	0.552	0.410	0.800	0.478	0.498	0.595	0.702	0.686	0.515	0.581
#Params	0.652	0.520	0.652	0.699	0.681	0.650	0.080	0.392	0.679	0.550	0.618	0.561
NLL	0.728	0.545	0.512	0.332	0.675	0.478	0.419	0.580	0.631	0.653	0.415	0.542
Entropy	0.693	0.513	0.426	0.536	0.644	0.441	0.093	0.463	0.585	0.683	0.510	0.508
Feature dim	0.530	0.432	0.530	0.573	0.560	0.539	0.101	0.328	0.533	0.468	0.600	0.472
k-means NMI	-0.527	0.563	0.542	0.712	0.498	0.282	0.375	0.071	0.344	0.455	0.460	0.343
RepVar	0.522	-0.126	0.355	0.250	-0.060	0.091	-0.265	-0.019	0.345	0.178	0.473	0.159
LDA	-0.462	0.259	0.035	0.485	-0.145	0.001	0.315	0.045	0.043	0.107	-0.246	0.040
LFC	-0.163	-0.086	-0.063	0.204	0.382	-0.192	0.584	0.057	0.025	-0.100	-0.375	0.025
RankMe	-0.477	0.015	-0.465	-0.357	-0.219	-0.018	0.173	-0.008	-0.295	-0.259	-0.424	-0.212

Table 25: **Self-supervised CNNs: per-target τ_w at 3 labelled samples per class.**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
LogME	0.390	0.699	**0.960**	0.736	**0.817**	**0.912**	0.835	0.870	0.427	**0.873**	0.650	**0.743**
TransRate	0.054	**0.778**	0.736	**0.807**	0.735	0.886	0.832	0.835	0.605	0.850	0.703	0.711
TMI	0.063	0.570	0.796	0.729	0.763	0.886	**0.870**	**0.898**	0.491	**0.873**	0.650	0.690
PED	0.373	0.681	0.699	–	0.694	0.522	0.721	0.665	0.242	0.610	–	0.579
SFDA	0.353	0.681	0.699	–	0.671	0.522	0.741	0.665	0.242	0.610	–	0.576
SA	0.353	0.543	0.699	–	0.671	0.522	0.741	0.665	0.242	0.610	–	0.561
NCTI	-0.060	0.737	0.551	0.149	0.632	0.396	0.674	0.791	0.555	0.508	0.293	0.475
H-Score$_{\text{reg}}$	-0.218	0.631	0.642	0.393	0.533	0.291	0.604	0.591	0.326	0.566	0.680	0.458
PARC	-0.379	0.609	0.739	0.453	0.141	0.421	0.581	0.727	0.238	0.405	0.249	0.380
GBC	0.300	0.646	0.421	0.383	0.388	-0.008	0.636	0.527	0.666	0.243	-0.287	0.356
ITM	-0.245	0.140	0.454	0.483	0.132	0.486	0.510	0.367	0.581	0.162	0.118	0.290
k-NN	-0.212	0.016	0.621	–	0.486	0.183	0.529	0.547	0.285	0.504	-0.206	0.275
DISCO	-0.129	0.076	0.580	0.323	-0.285	0.309	0.032	0.248	0.373	-0.059	-0.406	0.097
LEAD	-0.322	-0.334	0.661	0.660	-0.420	-0.086	0.001	-0.049	0.156	0.025	-0.431	-0.013
H-Score	0.150	0.034	0.253	0.026	0.064	-0.025	-0.625	-0.219	0.212	0.019	-0.031	-0.013
Baselines evaluated in this paper												
LDA	0.063	0.754	0.796	0.753	0.781	0.911	**0.870**	0.832	0.605	0.850	0.679	0.718
RepVar	**0.747**	0.379	0.626	0.527	0.497	0.578	0.444	0.508	0.199	0.599	**0.762**	0.533
LFC	-0.425	0.374	0.459	0.522	0.145	0.520	0.235	0.555	0.313	-0.220	0.003	0.226
RankMe	-0.269	0.513	0.262	-0.021	0.347	0.227	0.481	0.357	0.064	0.535	-0.299	0.200
k-means NMI	-0.341	0.627	-0.478	0.627	0.446	-0.074	0.538	-0.386	**0.710**	0.134	-0.172	0.148

Table 26: **Vision transformers: per-target τ_w at 3 labelled samples per class.**

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
TE published methods												
LogME	0.510	0.589	0.675	0.522	0.628	**0.808**	**0.845**	0.740	**0.870**	0.599	0.818	0.691
H-Score$_{\text{reg}}$	0.598	0.248	0.702	-0.119	0.472	0.641	0.652	**0.754**	0.721	0.778	0.402	0.532
ITM	0.332	0.745	0.471	-0.193	**0.922**	0.570	0.656	0.655	0.347	**0.840**	0.431	0.525
LEAD	0.622	**0.779**	0.585	0.593	0.825	-0.310	0.480	0.250	0.296	0.832	0.631	0.508
DISCO	0.584	0.592	0.567	0.268	0.736	0.111	0.515	0.655	0.338	0.591	0.553	0.501
PARC	0.051	0.537	0.622	-0.309	0.670	-0.179	0.461	0.578	0.648	0.637	0.608	0.393
k-NN	0.540	0.437	0.266	–	0.573	0.114	0.373	0.410	0.836	0.219	-0.149	0.362
PED	0.119	0.292	0.533	0.122	0.242	0.336	0.376	0.336	0.376	0.443	0.379	0.323
SFDA	0.119	0.292	0.533	0.122	0.242	0.336	0.376	0.336	0.376	0.443	0.379	0.323
TransRate	-0.153	0.163	0.210	0.494	0.463	0.407	0.533	0.565	0.270	0.467	0.016	0.312
H-Score	0.293	-0.290	**0.712**	0.269	0.465	-0.186	0.500	0.330	0.221	0.832	0.108	0.296
NCTI	-0.157	0.662	-0.004	-0.161	0.560	0.328	0.478	0.596	0.445	0.169	-0.619	0.209
GBC	-0.064	-0.198	0.229	-0.052	0.037	0.184	0.298	0.400	0.114	0.692	0.527	0.197
SA	-0.271	-0.064	-0.029	0.122	0.378	0.132	0.366	0.313	0.400	0.248	0.379	0.179
TMI	-0.285	-0.351	-0.289	0.455	0.286	0.181	0.091	0.202	0.038	0.011	0.016	0.032
Baselines evaluated in this paper												
Feature dim	**0.660**	0.514	0.650	0.502	0.666	0.750	0.816	0.724	0.856	0.736	**0.845**	**0.702**
#Params	0.524	0.619	0.541	**0.653**	0.772	0.776	0.705	0.560	0.738	0.804	0.762	0.678
RankMe	0.195	0.236	0.612	-0.079	0.199	0.643	0.625	0.741	0.758	0.736	0.143	0.437
k-means NMI	-0.288	0.533	0.550	0.359	0.784	-0.026	0.279	0.410	0.413	0.560	0.338	0.356
LFC	-0.035	0.422	0.357	0.406	0.459	0.261	0.313	0.412	0.418	0.289	0.082	0.308
LDA	-0.163	-0.318	0.211	0.540	0.025	0.308	0.308	0.461	0.424	0.062	-0.110	0.159
RepVar	0.152	-0.204	-0.681	0.056	-0.286	-0.436	-0.299	-0.099	-0.236	-0.131	-0.181	-0.213

ation. Only the three hubs with a sample-budget sweep appear here; the combined hub is reported at the full budget only (Section 6).

Table 27: **Robustness of each score to the sample budget.** For every score, the mean, standard deviation, minimum and maximum of its target-averaged τ_w over the samples-per-class sweep of Tables 21 to 23 (three samples per class to the full set), on the three hubs with a complete sweep, together with the *degradation deg* $= \tau_w(\text{full budget}) - \tau_w(\text{lowest sample budget})$, the change from the fewest labels (three, or the smallest budget at which the score is defined) to all labels. A low *std* with a high *min* and a small *deg* marks a score whose quality is insensitive to the sample budget; a large positive *deg* means the full-budget number over-states its few-shot behaviour. Data-free scores are budget-invariant, so their std and deg are 0. A dash marks a hub where the score is undefined or depends on the source classifier head. **Bold** marks the best value per hub and column among data-dependent scores (highest mean, min and max; lowest std; smallest |deg|).

	Sup. CNNs					SSL CNNs					ViTs				
Score	mean	std	min	max	deg	mean	std	min	max	deg	mean	std	min	max	deg
TE published methods															
LogME	0.493	0.054	0.437	0.569	-0.109	0.617	0.105	0.451	**0.743**	-0.292	**0.669**	0.035	**0.620**	**0.711**	-0.071
SFDA	0.385	0.168	0.201	0.659	0.458	0.672	0.044	0.576	0.706	0.129	0.504	0.089	0.323	0.592	0.260
SA	0.488	0.125	0.229	0.633	0.404	0.675	0.058	0.561	0.734	0.168	0.382	0.144	0.179	0.563	0.236
NCTI	0.518	0.174	0.194	**0.681**	0.475	0.621	0.090	0.475	0.716	0.227	0.361	0.102	0.209	0.466	0.239
ITM	0.509	0.138	0.310	0.674	0.364	0.451	0.106	0.290	0.614	0.324	0.493	0.071	0.407	0.623	-0.041
PED	0.323	0.075	0.201	0.400	0.182	0.630	0.023	0.579	0.660	0.054	0.459	0.061	0.323	0.525	0.167
LEEP	0.470	0.053	0.387	0.534	0.147	–	–	–	–	–	–	–	–	–	–
TransRate	0.229	0.065	0.117	0.320	-0.090	0.715	0.004	0.711	0.724	0.013	0.454	0.103	0.312	0.566	0.253
PARC	0.406	0.099	0.234	0.508	0.223	0.415	0.059	0.336	0.502	-0.024	0.490	0.053	0.393	0.530	0.013
H-Score$_{\text{reg}}$	0.181	0.124	0.060	0.399	0.315	0.532	0.046	0.458	0.595	0.089	0.594	0.055	0.530	0.681	0.149
OTCE	0.391	0.013	0.375	0.406	-0.030	–	–	–	–	–	–	–	–	–	–
LEAD	0.349	0.041	0.297	0.431	0.100	0.198	0.240	-0.060	0.524	0.459	0.606	0.060	0.508	0.682	0.174
k-NN	0.394	0.175	0.137	0.565	0.428	0.406	0.104	0.231	0.518	0.230	0.352	0.059	0.224	0.405	0.043
DISCO	0.528	0.020	0.477	0.546	0.069	0.025	0.036	-0.030	0.097	-0.067	0.537	0.035	0.487	0.576	0.072
TMI	0.354	**0.007**	0.344	0.369	-0.014	0.697	0.004	0.690	0.702	0.002	0.024	**0.004**	0.021	0.032	**-0.011**
GBC	0.326	0.121	0.145	0.472	0.297	0.279	0.089	0.187	0.439	-0.162	0.466	0.121	0.197	0.562	0.328
H-Score	0.324	0.179	0.048	0.596	0.433	0.155	0.205	-0.046	0.448	0.456	0.514	0.158	0.276	0.691	0.395
NCE	0.261	0.081	0.156	0.389	0.169	–	–	–	–	–	–	–	–	–	–
ETran	0.226	0.192	-0.048	0.472	0.432	–	–	–	–	–	–	–	–	–	–
ATC	0.052	0.024	0.011	0.089	-0.045	–	–	–	–	–	–	–	–	–	–
Baselines evaluated in this paper															
INB	0.654	0.000	0.654	0.654	0.000	–	–	–	–	–	–	–	–	–	–
#Params	0.561	0.000	0.561	0.561	0.000	–	–	–	–	–	0.678	0.000	0.678	0.678	0.000
Feature dim	0.472	0.000	0.472	0.472	0.000	–	–	–	–	–	0.702	0.000	0.702	0.702	0.000
MSP	**0.571**	0.009	**0.558**	0.583	**-0.002**	–	–	–	–	–	–	–	–	–	–
NLL	0.549	0.019	0.521	0.573	0.030	–	–	–	–	–	–	–	–	–	–
Entropy	0.478	0.016	0.457	0.508	-0.025	–	–	–	–	–	–	–	–	–	–
LDA	0.252	0.186	0.040	0.510	0.470	**0.727**	0.007	**0.717**	0.734	0.016	0.308	0.094	0.159	0.429	0.246
k-means NMI	0.458	0.094	0.288	0.574	0.180	0.312	0.136	0.142	0.487	0.311	0.419	0.032	0.356	0.461	0.073
RankMe	-0.197	0.066	-0.303	-0.080	0.132	0.334	0.108	0.200	0.467	0.255	0.548	0.056	0.437	0.598	0.157
LFC	0.130	0.061	0.025	0.212	0.187	0.139	0.049	0.064	0.226	-0.161	0.371	0.043	0.308	0.429	0.031
RepVar	0.200	0.026	0.159	0.240	0.049	0.533	**0.003**	0.526	0.537	**-0.001**	-0.219	0.008	-0.230	-0.209	-0.017
α-weighted	0.721	0.000	0.721	0.721	0.000	-0.562	0.000	-0.562	-0.562	0.000	0.108	0.000	0.108	0.108	0.000

A.8 LABEL-FREE SCORES

Finally, Table 28 collects *every* label-free score, both the data-free family, the uncertainty family (ATC, MSP, entropy) and RankMe, per target and across all four hubs at the full budget. The uncertainty family needs the source head and so is absent on the self-supervised and mixed hubs, while RankMe reads only the target features and is defined on every hub. Any score not defined on every model of a hub is omitted for that hub under the all-models-or-nothing rule.

A.9 LEAVE-FOUR-OUT ROBUSTNESS

Tables 29–32 report the leave-four-out ablation for every score defined on the complete hub: the full-hub τ_w, and the minimum and standard deviation of τ_w over all subsets with four models removed (LOO-M) and with four targets removed (LOO-D). The minimum is the quantity a practitioner with a differently-composed hub is exposed to.

Table 28: **Every label-free score: per-target τ_w, all four hubs.** Label-free scores, RankMe included, read no target annotation; values are at the full budget. **Bold** marks the best score per column within each hub.

Score	Airc.	Cal101	Cars	C10	C100	DTD	Flow.	Food	Pets	SUN	VOC	Avg
Supervised CNNs												
RankMe	-0.147	0.065	-0.129	0.090	0.057	0.096	-0.375	-0.240	-0.093	-0.183	-0.021	-0.080
Entropy	0.460	0.229	0.518	0.626	0.628	0.563	0.154	0.606	0.353	0.686	0.488	0.483
ATC	-0.171	-0.163	0.214	0.546	0.629	0.015	0.054	-0.328	-0.226	0.276	-0.363	0.044
α-weighted	0.643	**0.766**	0.643	0.706	0.747	**0.757**	**0.723**	**0.931**	0.713	**0.780**	0.522	**0.721**
INB	**0.732**	0.630	**0.732**	**0.775**	**0.759**	0.736	0.252	0.540	**0.725**	0.666	0.649	0.654
MSP	0.492	0.587	0.471	0.566	0.693	0.583	0.427	0.462	0.685	0.712	**0.685**	0.578
#Params	0.652	0.520	0.652	0.699	0.681	0.650	0.080	0.392	0.679	0.550	0.618	0.561
Feature dim	0.530	0.432	0.530	0.573	0.560	0.539	0.101	0.328	0.533	0.468	0.600	0.472
Self-Supervised CNNs												
RankMe	-0.265	**0.547**	**0.577**	**0.552**	**0.302**	**0.728**	**0.500**	**0.576**	**0.502**	**0.535**	**0.453**	**0.455**
α-weighted	**-0.021**	-0.555	-0.733	-0.563	-0.716	-0.597	-0.586	-0.756	-0.208	-0.737	-0.708	-0.562
Vision Transformers												
RankMe	0.465	0.302	0.640	0.226	0.440	0.643	**0.854**	**0.768**	**0.892**	0.577	0.727	0.594
α-weighted	-0.082	-0.088	-0.069	0.380	0.078	0.323	0.192	0.211	0.031	0.197	0.018	0.108
Feature dim	**0.660**	0.514	**0.650**	0.502	0.666	0.750	0.816	0.724	0.856	0.736	**0.845**	**0.702**
#Params	0.524	**0.619**	0.541	**0.653**	**0.772**	**0.776**	0.705	0.560	0.738	**0.804**	0.762	0.678
Mixed hub												
RankMe	0.271	-0.004	0.044	-0.090	-0.138	0.016	-0.117	-0.101	**0.503**	-0.037	0.026	0.034
α-weighted	0.381	0.126	0.243	-0.158	-0.222	0.255	0.142	-0.013	0.180	-0.153	-0.116	0.060
#Params	0.132	**0.490**	**0.409**	**0.641**	**0.770**	**0.528**	**0.299**	**0.519**	0.394	**0.605**	**0.468**	**0.478**
Feature dim	**0.649**	0.338	0.332	-0.229	-0.106	0.113	0.139	-0.078	0.496	-0.032	0.020	0.149

Table 29: Supervised CNNs: leave-four-out robustness.

		LOO-M (models)			LOO-D (targets)		
Score	τ_w	min	max	std	min	max	std
TE published methods							
ITM	0.674	0.516	0.845	0.057	0.587	0.781	0.041
NCTI	0.669	0.455	0.800	0.065	0.587	0.762	0.034
SFDA	0.659	0.369	0.766	0.066	0.558	0.756	0.038
SA	0.633	0.471	0.746	0.055	0.521	0.790	0.063
H-Score	0.596	0.273	0.781	0.100	0.490	0.682	0.039
k-NN	0.565	0.399	0.733	0.061	0.389	0.813	0.093
PACTran	0.551	0.358	0.783	0.080	0.394	0.783	0.079
DISCO	0.546	0.262	0.761	0.106	0.425	0.681	0.048
LEEP	0.534	0.330	0.676	0.070	0.405	0.719	0.077
PARC	0.496	0.350	0.672	0.061	0.310	0.732	0.087
ETran	0.472	0.226	0.722	0.078	0.295	0.707	0.086
GBC	0.442	0.240	0.692	0.096	0.238	0.632	0.079
LogME	0.440	0.054	0.780	0.160	0.318	0.555	0.051
LEAD	0.431	-0.141	0.766	0.171	0.266	0.542	0.059
H-Score$_{\text{reg}}$	0.399	-0.166	0.738	0.169	0.281	0.496	0.049
PED	0.383	0.102	0.697	0.115	0.220	0.544	0.060
OTCE	0.375	-0.328	0.844	0.206	0.259	0.541	0.060
NCE	0.361	0.035	0.664	0.130	0.169	0.591	0.091
TMI	0.356	0.020	0.697	0.136	0.223	0.501	0.059
TransRate	0.230	-0.262	0.574	0.160	0.108	0.344	0.045
ATC	0.044	-0.186	0.336	0.114	-0.169	0.225	0.077
Baselines evaluated in this paper							
NLL	0.573	0.332	0.800	0.091	0.498	0.644	0.026
LDA	0.510	0.256	0.721	0.080	0.347	0.742	0.080
Entropy	0.483	0.045	0.722	0.109	0.395	0.588	0.039
LFC	0.212	-0.124	0.506	0.124	0.032	0.379	0.068
RepVar	0.207	-0.238	0.493	0.139	0.067	0.346	0.057
RankMe	-0.080	-0.436	0.445	0.144	-0.170	0.009	0.035
α-weighted	0.721	0.505	0.956	0.098	0.671	0.774	0.023
INB	0.654	0.261	0.885	0.141	0.599	0.732	0.034
MSP	0.579	0.327	0.801	0.089	0.512	0.644	0.024
#Params	0.561	-0.122	0.922	0.185	0.494	0.661	0.042
k-means NMI	0.523	0.319	0.672	0.059	0.378	0.732	0.086
Feature dim	0.472	0.101	0.700	0.119	0.417	0.552	0.033

Table 30: Self-supervised CNNs: leave-four-out robustness.

		LOO-M (models)			LOO-D (targets)		
Score	τ_w	min	max	std	min	max	std
TE published methods							
PACTran	0.729	0.494	0.850	0.048	0.645	0.844	0.054
SA	0.729	0.540	0.855	0.055	0.631	0.849	0.055
TransRate	0.724	0.545	0.827	0.050	0.637	0.839	0.053
SFDA	0.705	0.440	0.859	0.060	0.624	0.807	0.042
NCTI	0.702	0.294	0.846	0.060	0.597	0.836	0.058
TMI	0.692	0.508	0.838	0.058	0.583	0.822	0.056
PED	0.633	0.263	0.776	0.056	0.545	0.737	0.041
ITM	0.614	0.136	0.772	0.070	0.512	0.759	0.063
H-Score$_{\text{reg}}$	0.547	0.244	0.773	0.088	0.443	0.676	0.057
k-NN	0.505	0.170	0.718	0.085	0.354	0.699	0.082
LogME	0.451	0.047	0.790	0.131	0.315	0.621	0.061
H-Score	0.443	0.071	0.751	0.118	0.303	0.617	0.066
PARC	0.357	-0.072	0.638	0.120	0.169	0.620	0.088
GBC	0.194	-0.228	0.594	0.156	-0.009	0.353	0.072
LEAD	0.447	-0.365	0.494	0.168	-0.075	0.233	0.061
DISCO	0.030	-0.484	0.557	0.197	-0.150	0.193	0.065
Baselines evaluated in this paper							
LDA	0.734	0.555	0.855	0.052	0.639	0.857	0.056
RepVar	0.532	0.065	0.824	0.084	0.451	0.616	0.035
RankMe	0.455	-0.111	0.632	0.092	0.368	0.574	0.059
LFC	0.064	-0.179	0.369	0.100	-0.137	0.279	0.081
α-weighted	-0.562	-0.910	-0.132	0.110	-0.691	-0.463	0.054
Feature dim	nan	nan	nan	nan	nan	nan	nan
k-means NMI	0.459	0.085	0.632	0.093	0.289	0.691	0.096
#Params	nan	nan	nan	nan	nan	nan	nan

Table 31: Vision transformers: leave-four-out robustness.

		LOO-M (models)			LOO-D (targets)		
Score	τ_w	min	max	std	min	max	std
TE published methods							
H-Score	0.691	0.313	0.884	0.111	0.613	0.768	0.031
H-Score$_{\text{reg}}$	0.680	0.313	0.884	0.109	0.613	0.751	0.027
LEAD	0.682	0.134	0.814	0.136	0.544	0.694	0.029
LogME	0.620	0.192	0.839	0.121	0.538	0.714	0.037
SFDA	0.583	0.090	0.829	0.140	0.489	0.678	0.038
DISCO	0.573	0.079	0.769	0.133	0.480	0.666	0.038
TransRate	0.566	0.023	0.832	0.162	0.429	0.726	0.056
GBC	0.525	0.139	0.741	0.115	0.435	0.589	0.033
PED	0.490	-0.039	0.790	0.166	0.366	0.611	0.049
ITM	0.484	-0.055	0.785	0.160	0.403	0.597	0.042
NCTI	0.448	-0.188	0.714	0.169	0.305	0.612	0.068
PACTran	0.424	-0.044	0.689	0.128	0.263	0.621	0.072
SA	0.415	-0.189	0.696	0.164	0.292	0.602	0.068
PARC	0.406	-0.256	0.729	0.191	0.252	0.597	0.072
k-NN	0.405	-0.286	0.744	0.205	0.288	0.530	0.050
TMI	0.021	-0.666	0.545	0.289	-0.112	0.164	0.057
Baselines evaluated in this paper							
RankMe	0.594	0.240	0.890	0.141	0.470	0.729	0.049
LDA	0.405	-0.273	0.718	0.195	0.250	0.584	0.066
LFC	0.339	0.005	0.617	0.120	0.216	0.485	0.051
RepVar	-0.230	-0.513	0.281	0.191	-0.348	-0.120	0.052
α-weighted	0.108	-0.277	0.797	0.239	0.011	0.202	0.037
Feature dim	0.702	0.339	0.895	0.117	0.636	0.770	0.027
#Params	0.678	0.311	0.861	0.124	0.620	0.744	0.024
k-means NMI	0.429	-0.059	0.686	0.134	0.306	0.586	0.065

Table 32: Mixed hub: leave-four-out robustness.

		LOO-M (models)			LOO-D (targets)		
Score	τ_w	min	max	std	min	max	std
TE published methods							
ITM	0.490	0.395	0.589	0.033	0.405	0.594	0.040
NCTI	0.448	0.261	0.653	0.078	0.352	0.536	0.040
DISCO	0.363	0.138	0.605	0.089	0.262	0.473	0.041
k-NN	0.341	0.062	0.582	0.084	0.274	0.431	0.032
PACTran	0.329	0.110	0.588	0.082	0.247	0.417	0.032
GBC	0.327	0.167	0.556	0.080	0.254	0.399	0.031
PARC	0.310	0.131	0.515	0.077	0.213	0.421	0.042
LogME	0.271	0.138	0.440	0.051	0.194	0.356	0.033
SA	0.254	0.015	0.536	0.088	0.183	0.330	0.028
H-Score	0.245	0.086	0.468	0.059	0.166	0.320	0.032
H-Score$_{\text{reg}}$	0.222	0.064	0.415	0.054	0.130	0.312	0.036
SFDA	0.221	0.070	0.408	0.056	0.139	0.301	0.035
LEAD	0.214	0.069	0.441	0.060	0.153	0.289	0.032
TMI	0.182	-0.153	0.478	0.092	0.133	0.232	0.022
PED	0.165	-0.006	0.319	0.052	0.086	0.246	0.034
TransRate	-0.006	-0.279	0.325	0.082	-0.105	0.091	0.044
Baselines evaluated in this paper							
LFC	0.334	0.067	0.624	0.098	0.259	0.427	0.036
LDA	0.294	0.013	0.599	0.097	0.229	0.356	0.028
RankMe	0.034	-0.289	0.304	0.078	-0.067	0.117	0.044
RepVar	0.033	-0.431	0.259	0.136	-0.051	0.107	0.035
α-weighted	0.060	-0.104	0.318	0.074	-0.056	0.188	0.046
#Params	0.478	0.210	0.638	0.070	0.387	0.574	0.039
k-means NMI	0.301	0.089	0.508	0.079	0.221	0.413	0.041
Feature dim	0.149	-0.143	0.463	0.070	-0.025	0.298	0.062

B FURTHER ANALYSIS

B.1 ALTERNATIVE EVALUATION METRICS

The main text reports the weighted Kendall correlation τ_w, which rewards getting the *top* of the ranking right and is the standard TE metric. It is not the only reasonable target, and a score can look strong under one metric and weak under another. For every (score, hub, target) we therefore also record three complementary metrics and report their target-averages at the full budget in Table 33. Let s_m be the score and a_m the fine-tuning accuracy of model m over the M models of a hub on one target.

Pearson correlation r. The ordinary linear correlation between score and accuracy,

$$r \;=\; \frac{\sum_m (s_m - \bar{s})(a_m - \bar{a})}{\sqrt{\sum_m (s_m - \bar{s})^2}\,\sqrt{\sum_m (a_m - \bar{a})^2}}. \tag{11}$$

Unlike τ_w, which depends only on the ranking, r rewards a score whose *values* track accuracy linearly; it is sensitive to outliers and penalises monotone-but-nonlinear score–accuracy relationships that a rank metric ignores.

Recall@k. A practitioner ultimately fine-tunes the single model the score ranks first. Recall@k is 1 if that top-ranked model, $\arg\max_m s_m$, lies among the k truly best models $\{m : a_m \text{ in the top } k\}$, and 0 otherwise; averaged over targets it is the fraction of targets on which the score's top pick lands in the true top-k. Recall@1 is the strict "did it pick the best model" rate; recall@3 relaxes this to the top three. Both are budget-agnostic in spirit but computed here at the full budget.

B.2 ABSOLUTE VERSUS RELATIVE TRANSFERABILITY

The weighted Kendall correlation used in the paper measures relative transferability: whether a score orders the checkpoints of a hub on a target in the same way as fine-tuning does. It does not measure absolute transferability, that is, whether a (model, target) pair reaches a high accuracy after fine-tuning, because the ranking is computed within each target and the accuracy level of the target plays no role. Here we ask the complementary question: when all (model, target) pairs of a hub are pooled, does a score predict the accuracy each pair reaches?

Chance-corrected accuracy. Raw fine-tuning accuracy cannot be compared across targets, since 80% on a 10-class target and 80% on a 397-class target are very different results. We therefore correct each accuracy for chance,

$$\tilde{a}(m, \mathcal{T}) \;=\; \frac{a(m, \mathcal{T}) - 1/C_{\mathcal{T}}}{1 - 1/C_{\mathcal{T}}}, \tag{12}$$

where $a(m, \mathcal{T})$ is the fine-tuning top-1 accuracy of model m on target $\mathcal{T}$ and $1/C_{\mathcal{T}}$ is the chance accuracy of a target with $C_{\mathcal{T}}$ classes. The corrected accuracy is 0 at chance and 1 at perfect accuracy. We pool all (model, target) pairs of a hub and compute the weighted Kendall correlation between each score and the chance-corrected accuracy. A score that orders the models well within each target but does not follow the accuracy level across targets obtains a value near zero or below.

Results. Table 34 reports this pooled correlation for every score on the four hubs. Among the published scores, ITM (0.661 on average), PACTran (0.637) and k-NN (0.594) predict the pooled accuracy best. Among the baselines, k-means NMI is the strongest (0.430), followed by INB (0.347) and the parameter count (0.342). Several scores that rank the models well within a target fall below zero once the targets are pooled: H-Score (-0.147), LogME (-0.151), TransRate (-0.141) and H-Score$_{\text{reg}}$ (-0.099), although H-Score is the best score on the ViT hub and TransRate is among the best on the self-supervised hub within targets (Table 4). The values of these scores depend on properties of the target such as its number of classes, so when targets are pooled they do not follow the accuracy a pair reaches. A high within-target correlation therefore does not mean that the values of a score can be compared across targets.

Table 33: **Alternative evaluation metrics, all four hubs.** For every complete score, the Pearson correlation r, recall@1 (R@1) and recall@3 (R@3) between the score and the fine-tuning accuracy, averaged over the 11 targets at the full budget. **Bold** marks the best value per column within each hub. Compare with the weighted-Kendall headline (Table 1) and full per-target table (Table 21 ff.).

	Sup. CNNs			SSL CNNs			ViTs			Mixed		
Score	r	R@1	R@3	r	R@1	R@3	r	R@1	R@3	r	R@1	R@3
TE published methods												
NCTI	0.611	0.45	0.82	0.790	**0.64**	0.82	0.538	0.64	0.73	0.405	0.27	0.73
SFDA	0.565	0.45	**0.91**	0.783	0.55	**0.91**	0.670	0.55	0.91	0.275	0.00	0.27
H-Score	0.534	0.45	0.82	0.583	0.27	0.64	0.732	**0.73**	**1.00**	0.323	0.00	0.27
ITM	0.628	0.36	**0.91**	0.564	**0.64**	0.82	0.629	0.45	0.73	0.294	0.09	0.64
LogME	0.332	0.45	0.64	0.608	0.27	0.64	0.709	0.64	0.82	0.335	0.00	0.18
NCE	0.478	0.18	0.55	–	–	–	–	–	–	–	–	–
DISCO	0.601	0.27	0.73	0.399	0.00	0.09	0.658	0.64	0.73	0.248	0.18	0.45
PACTran	0.474	0.45	0.64	**0.826**	0.55	**0.91**	0.487	0.55	0.64	0.116	0.27	0.45
GBC	0.530	0.27	0.45	0.658	0.09	0.27	0.513	0.36	0.64	0.149	0.09	0.18
LEEP	0.457	0.36	0.73	–	–	–	–	–	–	–	–	–
k-NN	0.591	0.55	0.73	0.628	0.55	0.82	0.461	0.55	0.64	0.136	0.09	0.45
SA	0.660	0.55	0.82	0.643	0.55	**0.91**	0.534	0.55	0.73	-0.025	0.09	0.36
H-Score$_{\text{reg}}$	0.255	0.45	0.64	0.532	0.27	**0.91**	0.731	0.64	**1.00**	0.270	0.00	0.18
PED	0.281	0.18	0.82	0.688	0.27	0.73	0.552	0.55	0.82	0.221	0.09	0.09
PARC	0.499	0.36	0.73	0.530	0.36	0.64	0.541	0.55	0.55	0.149	0.18	0.27
LEAD	0.288	0.55	0.73	0.582	0.27	0.64	0.730	**0.73**	**1.00**	0.260	0.00	0.00
TMI	0.554	0.09	0.36	0.642	0.36	**0.91**	0.035	0.18	0.18	0.286	0.09	0.36
TransRate	0.078	0.09	0.45	0.747	0.55	**0.91**	0.652	0.55	0.82	0.030	0.00	0.27
OTCE	0.361	0.36	0.82	–	–	–	–	–	–	–	–	–
ETran	0.425	0.27	0.55	–	–	–	–	–	–	–	–	–
$\mathcal{N}$LEEP	–	–	–	–	–	–	0.538	0.55	0.55	0.088	0.09	0.55
ATC	0.147	0.09	0.09	–	–	–	–	–	–	–	–	–
Baselines evaluated in this paper												
LDA	0.471	0.45	0.73	0.646	0.55	**0.91**	0.436	0.55	0.82	-0.072	0.27	0.73
NLL	0.413	0.36	0.45	–	–	–	–	–	–	–	–	–
Entropy	0.308	0.09	0.36	–	–	–	–	–	–	–	–	–
LFC	0.132	0.09	0.27	0.152	0.00	0.27	0.563	0.00	0.18	0.275	0.27	0.73
RankMe	-0.159	0.00	0.09	0.450	0.36	0.73	**0.743**	0.27	0.55	-0.023	0.09	0.09
RepVar	0.333	0.09	0.18	0.527	0.09	0.73	-0.395	0.00	0.00	-0.082	**0.36**	**0.82**
α-weighted	**0.908**	0.18	**0.91**	-0.751	0.00	0.09	-0.149	0.00	0.09	0.110	0.00	0.18
INB	0.598	**0.73**	**0.91**	–	–	–	–	–	–	–	–	–
#Params	0.566	**0.73**	**0.91**	–	–	–	0.584	**0.73**	**1.00**	**0.416**	**0.36**	**0.82**
MSP	0.461	0.36	0.55	–	–	–	–	–	–	–	–	–
k-means NMI	0.542	0.55	0.82	0.600	0.55	0.73	0.571	0.36	0.55	0.079	0.09	0.45
Feature dim	0.434	0.00	0.09	–	–	–	0.719	0.00	0.45	0.145	0.18	0.36

Table 34: **Absolute transferability.** Weighted Kendall correlation between each score and the chance-corrected accuracy (Eq. (12)) over all pooled (model, target) pairs of each hub. A high value means that the score predicts the accuracy a pair reaches across targets, not only the order of the models within a target. A dash marks a score that is undefined on the hub, and **bold** marks the best score of each column within each group. Rows are sorted by the average.

Score	Sup. CNNs	SSL CNNs	ViTs	Mixed	Avg
TE published methods					
ITM	**0.724**	**0.689**	0.676	0.557	**0.661**
PACTran	0.606	0.586	**0.698**	**0.657**	0.637
k-NN	0.547	0.579	0.649	0.600	0.594
SFDA	0.590	0.583	0.593	0.446	0.553
SA	0.483	0.568	0.572	0.563	0.546
PARC	0.548	0.351	0.595	0.640	0.533
LEAD	0.482	0.426	0.599	0.612	0.530
$\mathcal{N}$LEEP	0.460	0.498	0.583	0.532	0.518
OTCE	0.451	–	0.519	0.471	0.480
GBC	0.415	0.238	0.627	0.520	0.450
NCTI	0.295	0.433	0.332	0.329	0.347
LEEP	0.344	–	–	–	0.344
DISCO	0.343	0.141	0.364	0.303	0.288
PED	0.207	0.299	0.352	0.004	0.215
NCE	0.212	–	–	–	0.212
ETran	0.183	–	–	–	0.183
TMI	-0.113	0.201	0.016	0.006	0.028
H-Score$_{\text{reg}}$	-0.118	-0.029	-0.064	-0.184	-0.099
TransRate	-0.266	0.043	-0.014	-0.327	-0.141
H-Score	-0.109	-0.198	-0.065	-0.218	-0.147
LogME	-0.240	-0.208	-0.100	-0.055	-0.151
ATC	-0.240	–	–	–	-0.240
Baselines evaluated in this paper					
LDA	**0.510**	**0.493**	**0.559**	0.443	**0.501**
k-means NMI	0.470	0.341	0.512	0.396	0.430
INB	0.308	–	0.407	0.326	0.347
LFC	0.361	0.153	0.334	**0.529**	0.344
#Params	0.267	–	0.392	0.366	0.342
Feature dim	0.290	–	0.412	0.077	0.260
α-weighted	0.379	-0.129	0.190	0.093	0.133
RepVar	0.027	0.140	0.007	-0.066	0.027
NLL	-0.074	–	–	–	-0.074
RankMe	-0.233	-0.114	0.129	-0.107	-0.081
MSP	-0.085	–	–	–	-0.085
Entropy	-0.110	–	–	–	-0.110

Table 35: How the top scores depend on the feature dimension d on the hubs where capacity varies. The two width measures use the targets differently. $\tau_w(\text{score}, d)$ correlates the score's model ranking with d within each target and averages over targets, exactly the weighted-Kendall protocol of the headline score-versus-accuracy number but with d replacing the accuracy, so it asks whether the score orders the models the way width does. $\tau_w(\text{magnitude}, d)$ instead averages each model's raw score over the targets and correlates these per-model means with d once across the models, so it captures scale rather than ranking. $\tau_w(\text{bias}, d)$ correlates the per-model estimation bias of Eq. (13) with d.

	$\tau_w(\text{score}, d)$		$\tau_w(\text{magnitude}, d)$	$\tau_w(\text{bias}, d)$
Score	Sup-CNN	ViT	Sup-CNN	Sup-CNN
LDA	0.150	0.480	0.144	−0.484
H-Score	0.763	0.772	0.882	0.550
SFDA	0.643	0.712	0.678	0.517

B.3 The feature-dimension confound

Averaged over the three hubs, SFDA is the strongest published estimator ($\tau_w = 0.649$), so we look more closely at SFDA, H-Score, and LDA. We study them through the feature dimension, the width d of the penultimate layer, because that is where these scores are computed: each turns an $N \times d$ feature matrix into a scalar. On the two hubs where this width varies, H-Score and SFDA rank the models almost the way feature dimension does, with τ_w against d of 0.763 and 0.643 on supervised CNNs and 0.772 and 0.712 on ViTs, whereas LDA barely tracks it (0.150 and 0.480); the self-supervised hub fixes d and cannot show the effect. The dependence is one of raw magnitude too, not only of ranking: the mean score value itself rises with width, at $\tau_w = 0.882$ for H-Score and 0.678 for SFDA against 0.144 for LDA, and H-Score ranges from about 6 to 100 across the hub.

H-Score is $\mathrm{tr}(\Sigma_f^{-1}\Sigma_b)$, the between-class covariance Σ_b whitened by the total feature covariance Σ_f and summed over all d coordinates, and SFDA is a regularised Fisher discriminant whose class Mahalanobis distances are taken under the same Σ_f^{-1}. Both invert a covariance estimated from a finite feature sample, so as the width d grows relative to the sample size n, Σ_f becomes ill conditioned and its regularised inverse is biased upward, in the Gaussian case by roughly a factor $n/(n-d-1)$. The trace and the distances then accumulate spurious mass that scales with d rather than with any real gain in class separability, so a wider penultimate layer inflates both scores almost mechanically.

We confirm this mechanism directly by removing width as a variable. We project every probe to a common dimension with PCA ($K = 256$) and rescore, so that all models enter each score at the same feature dimension. On the supervised CNN hub, where width varies across models, the projection pulls H-Score and SFDA down: their agreement with the fine-tuning ground truth falls from $\tau_w = 0.596$ to 0.469 for H-Score and from 0.659 to 0.576 for SFDA, while LDA is left unchanged, from 0.510 to 0.520 (Table 36). Their correlation with the feature dimension falls at the same time, from 0.763 to 0.600 for H-Score and from 0.643 to 0.391 for SFDA, so part of what they read at native width was the width itself rather than class separability. On the self-supervised hub, where every backbone is the same ResNet-50 and the width is already fixed, the same projection barely moves any score (H-Score 0.443 to 0.428, SFDA 0.705 to 0.686, LDA 0.734 to 0.734), the control we expect when there is no width spread to remove.

The scores are not empty, though: on the self-supervised hub, where d is constant, they keep a positive if moderate signal ($\tau_w = 0.705$ for SFDA and 0.443 for H-Score), and the confound only takes over once capacity varies across models. To quantify it per model, we define the bias of a score s as its mean rank position minus the ground-truth rank position over the targets $\mathcal{T}$,

$$b_s(m) \;=\; \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\Big(\mathrm{rank}_s(m,t) - \mathrm{rank}_a(m,t)\Big), \tag{13}$$

so a positive $b_s(m)$ means the score ranks model m above its true position. This bias correlates with feature dimension at $\tau_w = 0.550$ for H-Score and 0.517 for SFDA, so both systematically over-rank the widest backbones, whereas for LDA the correlation is -0.484: LDA under-rates wide models, so for it the spread of capacity itself is what hurts. Removing that spread helps, and LDA rises from $\tau_w = 0.510$ on the supervised CNN hub to 0.734 on the self-supervised one.

Table 36: Same-dimension control: each score's weighted Kendall against the fine-tuning ground truth when every probe is scored at its native feature dimension and when all probes are first projected to a common $K = 256$ with PCA, with the change Δ. Removing the width spread degrades H-Score and SFDA on the supervised CNN hub, where width varies across models, while leaving LDA unchanged; on the self-supervised hub, where every backbone is the same ResNet-50 and width is already fixed, the projection changes little.

Score	native d	$K = 256$	Δ
Supervised CNNs (width varies)			
LDA	0.510	0.520	0.010
H-Score	0.596	0.469	−0.127
SFDA	0.659	0.576	−0.083
Self-supervised CNNs (width fixed)			
LDA	0.734	0.734	0.000
H-Score	0.443	0.428	−0.015
SFDA	0.705	0.686	−0.019

B.4 Weight-only scores: α-weighted versus effective rank

Both scores read only the trained weights and no target data. For a layer l we flatten its weight into $W_l \in \mathbb{R}^{m_l \times n_l}$, with m_l outputs and n_l inputs (for a convolution $n_l = C_{\text{in}} k_h k_w$), and write $\sigma_{l,1} \geq \cdots \geq \sigma_{l,r_l} > 0$ for its singular values, so that the eigenvalues of $W_l^\top W_l$ are $\lambda_{l,i} = \sigma_{l,i}^2$.

α-weighted. A power law $\rho(\lambda) \propto \lambda^{-\alpha_l}$ is fitted to the eigenvalues of each layer, and the score averages the scale-weighted exponent over a layer set $\mathcal{L}$ and negates it, since a smaller value marks a better-trained model (Eq. (2)),

$$S_{\alpha\text{w}} \;=\; -\frac{1}{|\mathcal{L}|} \sum_{l \in \mathcal{L}} \alpha_l \, \log_{10} \lambda_{l,\max}. \tag{14}$$

where $\mathcal{L}$ is the set of all layers.

Effective rank. With the singular values of a layer normalised to a distribution, $p_{l,i} = \sigma_{l,i} / \sum_j \sigma_{l,j}$, the effective rank is their exponentiated entropy, the same quantity RankMe computes on features (Eq. (1)) but applied to a weight matrix, and the score takes its maximum over the non-head layers,

$$\text{erank}(W_l) \;=\; \exp\Big(-\sum_i p_{l,i} \log p_{l,i} \Big), \qquad S_{\text{er}} \;=\; \max_{l \,\in\, \mathcal{L} \setminus \mathcal{H}} \text{erank}(W_l), \tag{15}$$

where $\mathcal{H}$ is the classifier head. Two properties explain its behaviour. It is invariant to the scale of the weights, $\text{erank}(cW_l) = \text{erank}(W_l)$ for any $c > 0$, so it ignores the spectral norm that enters α-weighted through $\log_{10} \lambda_{l,\max}$. It is not invariant to size, since $1 \leq \text{erank}(W_l) \leq \min(m_l, n_l)$, so its maximum is bounded by the widest non-head layer and grows with capacity when architectures differ.

Table 37: Weight-only scores on the four hubs: τ_w averaged over the 11 targets. The two indented rows split α-weighted into its exponent (scored so that a smaller α is better) and its scale term (scored so that a larger spectral norm is better). The no-head-maximum setting of effective rank is the best of 48 layer-subset and aggregation choices on these same hubs and is therefore optimistic.

Score	Layers, aggregation	Sup. CNNs	SSL CNNs	ViTs	Mixed
α-weighted	all, mean	0.721	−0.562	0.108	0.060
exponent α only	all, mean	0.464	−0.508	0.296	0.069
scale $\log \lambda_{\max}$ only	all, mean	−0.765	0.672	−0.132	0.076
Effective rank	all, mean	0.144	0.593	0.362	0.377
Effective rank	no head, max	0.520	0.593	0.662	0.513

Table 37 compares the two scores. Over all 48 settings, α-weighted is positive on the supervised and negative on the self-supervised hub in every one and sign-consistent across the four hubs in

Table 38: **Robustness to pre-trained weights.** Weighted Kendall τ_w of every score on the four same-architecture checkpoints (ResNet-50, ResNet-101, ResNet-152 and MobileNet-v2) against the SFDA (V1) and ITM (V2) ground truths, with $\Delta = \tau_w^{\text{ITM}} - \tau_w^{\text{SFDA}}$. Rows are sorted by the SFDA τ_w, and **bold** marks the best value of each ground-truth column. With only four checkpoints, τ_w takes coarse and tied values, so the numbers show a direction rather than a significant difference.

		τ_w					τ_w		
Score	Fam.	**SFDA**	**ITM**	Δ	Score	Fam.	**SFDA**	**ITM**	Δ
INB	Base	**+0.976**	+0.661	-0.315	NCE	TE	+0.618	**+0.701**	+0.082
#Params	Base	**+0.976**	+0.661	-0.315	LEEP	TE	+0.616	+0.661	+0.045
α-weighted	Base	**+0.976**	+0.461	-0.515	GBC	TE	+0.610	+0.457	-0.153
H-Score	TE	+0.927	+0.516	-0.411	PACTran	TE	+0.602	+0.250	-0.353
LogME	TE	+0.927	+0.406	-0.521	SA	TE	+0.593	-0.064	-0.657
H-Score$_{\text{reg}}$	TE	+0.879	+0.476	-0.402	LFC	Base	+0.556	+0.178	-0.378
ITM	TE	+0.879	+0.356	-0.523	TransRate	TE	+0.534	+0.496	-0.039
NLL	Base	+0.845	+0.661	-0.184	PARC	TE	+0.531	+0.353	-0.178
MSP	Base	+0.828	+0.661	-0.167	k-means NMI	Base	+0.474	+0.229	-0.245
LEAD	TE	+0.821	+0.429	-0.392	ETran	TE	+0.469	+0.471	+0.002
SFDA	TE	+0.821	+0.379	-0.441	LDA	Base	+0.442	+0.360	-0.082
Entropy	Base	+0.736	+0.617	-0.119	ATC	TE	+0.245	-0.257	-0.502
OTCE	TE	+0.719	+0.125	-0.594	TMI	TE	+0.239	+0.436	+0.198
DISCO	TE	+0.695	+0.559	-0.136	RepVar	Base	+0.091	-0.475	-0.566
NCTI	TE	+0.662	+0.467	-0.195	PED	TE	+0.074	-0.074	-0.148
Feature dim	Base	+0.643	+0.618	-0.025	RankMe	Base	-0.071	+0.428	+0.499
k-NN	TE	+0.634	+0.044	-0.590					

none, so the reversal is structural rather than a choice of layers or aggregation. The two indented rows show why: both of its ingredients flip. On supervised CNNs a smaller exponent and a smaller spectral norm accompany better transfer, whereas on the self-supervised hub a larger exponent and a larger norm do, and their product inherits the inversion. Effective rank never shows this flip and is sign-consistent across the four hubs in 31 of the 48 settings. Its width-normalised form, $\text{erank}(W_l)/\min(m_l, n_l)$, is sign-consistent in none, which indicates that on the hubs where architectures differ much of its signal is capacity. On the self-supervised hub, where all ten checkpoints share one ResNet-50 and the same parameter count, this explanation is not available, so the positive score there comes from how the spectra spread.

B.5 Robustness to pre-trained weights

Four checkpoints, ResNet-50, ResNet-101, ResNet-152 and MobileNet-v2, appear in both ground-truth tables with the same architecture but different pre-trained weights. The SFDA hub loads the torchvision `IMAGENET1K_V1` weights, while the mixed (ITM) hub loads the torchvision `DEFAULT` (`V2`) weights, which are on average 3.37 points more accurate on ImageNet (Appendix A.1). This is the only place in the benchmark where the architecture is held fixed and only the weights change, so it shows how much the ranking of a score depends on the particular checkpoint. Table 38 reports the weighted Kendall correlation of each score on these four models against the V1 (SFDA) and V2 (ITM) ground truths, and the difference between the two.

Two observations stand out. First, most of the ImageNet gain does not carry over to the targets. A difference-in-differences against the seven checkpoints that share the same weights in both tables attributes only 0.29 points of downstream accuracy to the 3.37-point ImageNet gain, about 9% of it. Second, the four models are close in fine-tuning accuracy, so the new weights change their order. INB and the parameter count rank the V1 checkpoints almost perfectly ($\tau_w = 0.976$) but the V2 checkpoints less well (0.661), and most scores lose correlation in the same way: 28 of the 33 scores have a negative Δ. Only a few keep a similar correlation under both ground truths, such as ETran ($+0.002$), the feature dimension (-0.025) and TransRate (-0.039). With only four checkpoints, these numbers show a direction rather than a significant difference.

B.6 The language modality

Model hub. To check that TE-BENCH extends beyond vision, we assemble a small hub of eight pre-trained language models that differ widely in family and size: RoBERTa and its distilled variant DistilRoBERTa, uncased and cased DistilBERT, ALBERT v1 and v2, and ELECTRA base and

Table 39: Per-target weighted Kendall τ_w on the language hub, against the fine-tuning accuracies of You et al. (2021). TE-Ref is the same untuned $S_{\text{lda}} \oplus S_{\text{params}}$ reference used on the vision hubs. Only targets with at least three reported models are scored (models per target: MNLI 8, SST-2 6, all others 4).

Score	MNLI	QQP	QNLI	SST-2	CoLA	MRPC	RTE	Avg
LogME	0.324	1.000	1.000	0.676	0.813	0.733	0.813	0.766
SFDA	0.080	0.733	0.533	0.676	0.813	0.000	−0.667	0.310
H-Score	0.048	1.000	1.000	0.676	1.000	0.733	0.533	0.713
QDA	0.024	0.813	0.000	0.762	0.813	0.200	0.813	0.489
ITM	0.260	1.000	1.000	0.762	0.813	0.307	1.000	0.735
RankMe	−0.351	−0.333	−0.333	−0.550	−0.333	0.000	−0.667	−0.367
α-weighted	−0.033	−0.667	−0.667	−0.299	−0.667	−0.440	−0.333	−0.444
MSP	0.261	0.333	0.658	0.328	0.520	0.000	0.872	0.425
#Params	0.280	1.000	1.000	0.647	1.000	0.733	0.813	0.782
TE-Ref (TE-Ref)	0.297	1.000	1.000	0.676	1.000	0.733	0.813	**0.789**

small, spanning roughly 11M to 125M parameters. The wide spread of both architecture and capacity is what makes the hub a useful stress test for a transferability score.

Targets. The targets are seven sentence and sentence-pair classification tasks from the GLUE benchmark: MNLI, QQP, QNLI, SST-2, CoLA, MRPC, and RTE. For each (model, target) pair we cache a probe by mean-pooling the final hidden states of the frozen model over a sample of the task's training set, and compute the feature-based scores on that probe exactly as on the vision hubs.

Ground truth. We take the fine-tuning accuracies reported by You et al. (2021) (their Figure 6) as the ground truth, and score each estimator by the weighted Kendall correlation between its ranking of the models and those accuracies, per task. This ground truth is limited in several ways: it comes from a single published table under one fine-tuning protocol, most models are reported on only a subset of the tasks (four of the eight on all tasks except MNLI and SST-2, so a per-task ranking often rests on just four points), and the hub itself is small. The language results are therefore indicative rather than conclusive, a sanity check that the platform and the reference baseline behave as expected outside vision.

Results. Table 39 reports the per-target τ_w, and the picture matches the vision hubs: the reference baseline TE-Ref is the strongest entry on the average (0.789). Among the evaluated estimators the best are LogME (0.766), ITM (0.735) and H-Score (0.713), with the rest trailing (QDA 0.489, MSP 0.425, SFDA 0.310) or negative (RankMe and α-weighted). The striking point is that the bare parameter count already reaches $\tau_w = 0.782$, above every dedicated estimator, because the models span an order of magnitude in size and larger backbones simply transfer better on these tasks. In this regime capacity is the dominant signal and the transferability scores appear unnecessary: a parameter count orders the models at least as well as any of them, and TE-Ref inherits that strength while remaining the top entry. This is the sharpest form of the capacity effect documented in Section 6; we read it not as evidence that these estimators fail in general, but as a hub where capacity varies so widely, and the ground truth is so coarse, that a capacity heuristic is hard to beat.

B.7 Score redundancy and relations

Redundancy in the sample-budget response. Figure 1 compares score *values* at a single budget. A complementary, stricter question is whether scores *respond to the sample budget* the same way: as the labelled samples per class grow from three to the full set, does a score's accuracy improve on the same schedule as LDA? For each score we average its weighted Kendall over the 11 targets at each budget $k \in \{3, 4, 5, 10, 20, 50, 100, \text{all}\}$, its *budget curve*, and correlate that curve with LDA's on each of the three hubs (Table 40); data-free scores have a constant budget curve and are omitted.

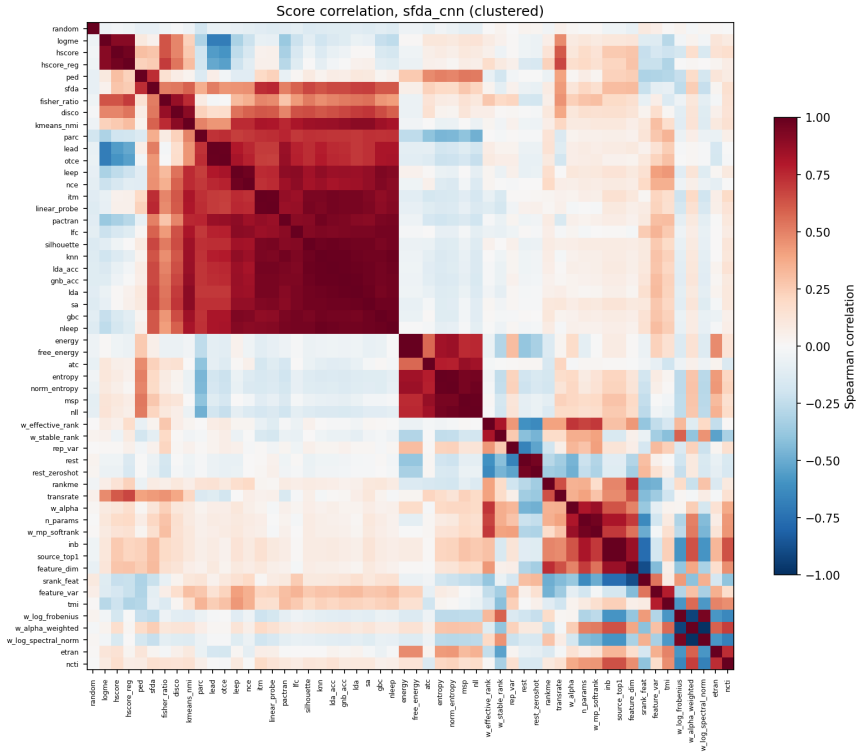

Figure 1: Spearman correlation between every pair of scores over the (model, target) pairs of the supervised-CNN hub, hierarchically clustered. The ordering separates a capacity-like block — parameter count, INB, source top-1 and the norm-based weight scores, which agree because they all track model size and pre-training quality — from a separability-like block — LDA, H-Score, SFDA, the probe accuracies and the likelihood scores, which agree because they all read target-feature class structure. The two blocks are only weakly correlated with each other, which is the redundancy-side view of the two-signal decomposition: within a block scores are near-substitutes, across blocks they are complementary.

Most label-based estimators track LDA closely. The class-separability family (SFDA, NCTI, H-Score and its shrinkage variant), the k-NN and k-means readouts, the likelihood scores (LEEP, NCE) and the adaptation scores (PED, SA), LEAD and RankMe all reach a mean budget-curve correlation above 0.75, and ETran reaches 0.98 on the supervised hub, the only hub where it is defined; in total, 14 of the 27 varying scores exceed 0.70 on average. Once the source head and raw scale are set aside, then, much of the literature improves with labels on the same schedule as a plain linear discriminant: on the budget axis these scores are near-substitutes for the generalisation half of TE-Ref. The exceptions are informative: LogME, OTCE and TMI *anti*-correlate, so the test discriminates rather than reporting a trivial "everything improves with more labels" artefact.

B.8 Variance reduction by cross-fitting

LDA estimates the class means from the target features and then scores each target sample against them. A sample therefore contributes to the class mean it is scored against, which makes the score optimistic, and the size of this optimism depends on the ratio of the feature dimension to the number of samples per class. On the supervised CNN hub the feature dimension ranges from 512 to $2{,}048$, so with few samples per class the optimism differs between models and can change their ranking. We test whether cross-fitting, which evaluates each sample only with statistics fitted on other samples, reduces this effect.

Cross-fitting. For a target and a sample budget, we split the samples of every class at random into two halves A and B of equal size. The class means and class shares are estimated on A and

Table 40: **Redundancy in the sample-budget response.** Pearson correlation between each score's target-averaged weighted-Kendall curve over the sample budget (3 to all samples per class) and LDA's, on the three hubs. **Bold** marks a correlation of at least 0.70; a dash marks a hub where the score is undefined or depends on the source classifier head. Data-free scores have a constant budget curve and are omitted. A high value means the score improves with labels the same way LDA does, which indicates that it is largely a proxy for target-feature separability.

Score vs. LDA	Sup. CNNs	SSL CNNs	ViTs	mean
TE published methods				
ETran	**+0.98**	–	–	**+0.98**
LEEP	**+0.96**	–	–	**+0.96**
NCE	**+0.94**	–	–	**+0.94**
NCTI	**+0.86**	**+0.99**	**+0.93**	**+0.93**
H-Score$_{\text{reg}}$	**+0.94**	**+0.88**	**+0.93**	**+0.92**
SFDA	**+0.95**	**+0.88**	**+0.91**	**+0.91**
H-Score	**+0.96**	**+0.82**	**+0.95**	**+0.91**
SA	**+0.88**	**+0.90**	**+0.87**	**+0.88**
k-NN	**+0.90**	**+0.91**	+0.69	**+0.83**
LEAD	+0.56	**+0.79**	**+0.97**	**+0.77**
PED	**+0.77**	**+0.72**	**+0.80**	**+0.76**
PARC	**+0.83**	+0.60	+0.12	+0.52
DISCO	+0.58	-0.09	**+0.91**	+0.47
ITM	**+0.94**	**+0.89**	-0.75	+0.36
TransRate	-0.44	+0.48	**+0.94**	+0.33
GBC	**+0.92**	-0.94	**+0.78**	+0.25
ATC	-0.08	–	–	-0.08
TMI	-0.26	+0.35	-0.93	-0.28
OTCE	-0.84	–	–	-0.84
LogME	-0.91	-0.94	-0.69	-0.85
Baselines evaluated in this paper				
NLL	**+0.94**	–	–	**+0.94**
RankMe	**+0.78**	**+0.95**	**+0.91**	**+0.88**
k-means NMI	**+0.87**	**+0.86**	**+0.84**	**+0.85**
MSP	+0.46	–	–	+0.46
LFC	**+0.87**	-0.89	**+0.75**	+0.24
RepVar	+0.56	-0.20	-0.46	-0.03
Entropy	-0.31	–	–	-0.31

the nearest-class-mean softmax of LDA is evaluated on B, the two halves are then swapped, and the two values are averaged. We repeat this with $K = 5$ random splits and average the K values, which reduces the variance that a single split introduces. The splits depend only on the target and the budget, so every model of the hub is scored on the same halves. The budgets of 10 and 50 samples per class are drawn once per target with a fixed seed shared by all models.

Results. Table 41 compares the published score (base) with the cross-fitted score (cf) on each target. The gain is largest at small budgets. At 10 samples per class, cross-fitting raises the average τ_w of LDA from 0.166 to 0.531 and improves 10 of the 11 targets, with the largest gains on CIFAR-100 (+0.686), DTD (+0.596) and SUN397 (+0.588). At 50 samples per class the average gain is +0.133, on 8 of the 11 targets. With all samples the change is small (+0.039), since the optimism shrinks as the number of samples per class grows. A large part of the drop of LDA at small budgets on this hub therefore comes from evaluating it on the samples that define its class means.

Table 41: **LDA with and without cross-fitting on the supervised CNN hub.** Per-target τ_w of the published score (base) and the cross-fitted score (cf, averaged over 5 random splits) at 10 and 50 samples per class and with all samples, and the change $\Delta = \text{cf} - \text{base}$.

	10 per class			50 per class			All samples		
Target	base	cf	Δ	base	cf	Δ	base	cf	Δ
Aircraft	-0.132	-0.216	-0.084	-0.171	-0.141	$+0.030$	-0.212	-0.212	$+0.000$
Caltech-101	0.303	0.647	$+0.343$	0.594	0.485	-0.109	0.641	0.528	-0.113
Cars	0.093	0.280	$+0.187$	0.053	0.225	$+0.173$	0.074	0.195	$+0.122$
CIFAR-10	0.165	0.619	$+0.455$	0.811	0.684	-0.127	0.811	0.811	$+0.000$
CIFAR-100	0.227	0.913	$+0.686$	0.777	0.869	$+0.092$	0.846	0.869	$+0.023$
DTD	0.155	0.751	$+0.596$	0.298	0.621	$+0.323$	0.364	0.634	$+0.269$
Flowers	0.180	0.215	$+0.035$	0.213	0.213	$+0.000$	0.191	0.122	-0.069
Food	0.446	0.588	$+0.142$	0.628	0.671	$+0.043$	0.754	0.781	$+0.028$
Pets	0.272	0.818	$+0.546$	0.521	0.868	$+0.347$	0.671	0.742	$+0.070$
SUN397	0.227	0.815	$+0.588$	0.560	0.878	$+0.318$	0.774	0.862	$+0.088$
VOC2007	-0.111	0.411	$+0.523$	0.446	0.822	$+0.376$	0.699	0.713	$+0.014$
Average	0.166	0.531	$+0.365$	0.430	0.563	$+0.133$	0.510	0.550	$+0.039$
Targets improved		10 of 11			8 of 11			7 of 11	