arXiv:2202.10054v1 [cs.LG] 21 Feb 2022

Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution

Ananya Kumar Aditi Raghunathan Robbie Jones
Tengyu Ma Percy Liang

Stanford University
Department of Computer Science

{ananya, aditir, rmjones, tengyuma, pliang}@cs.stanford.edu

Abstract

When transferring a pretrained model to a downstream task, two popular methods are full fine-tuning (updating all the model parameters) and linear probing (updating only the last linear layer—the "head"). It is well known that fine-tuning leads to better accuracy in-distribution (ID). However, in this paper, we find that fine-tuning can achieve worse accuracy than linear probing out-of-distribution (OOD) when the pretrained features are good and the distribution shift is large. On 10 distribution shift datasets (Breeds-Living17, Breeds-Entity30, DomainNet, CIFAR $\rightarrow$ STL, CIFAR10.1, FMoW, ImageNetV2, ImageNet-R, ImageNet-A, ImageNet-Sketch), fine-tuning obtains on average 2% higher accuracy ID but 7% lower accuracy OOD than linear probing. We show theoretically that this tradeoff between ID and OOD accuracy arises even in a simple setting: fine-tuning overparameterized two-layer linear networks. We prove that the OOD error of fine-tuning is high when we initialize with a fixed or random head—this is because while fine-tuning learns the head, the lower layers of the neural network change simultaneously and distort the pretrained features. Our analysis suggests that the easy two-step strategy of linear probing then full fine-tuning (LP-FT), sometimes used as a fine-tuning heuristic, combines the benefits of both fine-tuning and linear probing. Empirically, LP-FT outperforms both fine-tuning and linear probing on the above datasets (1% better ID, 10% better OOD than full fine-tuning).

1 Introduction

Pretraining a model on a large dataset before transferring to a downstream task's training data substantially improves accuracy over training from scratch—for example, pretraining a ResNet-50 on unlabeled ImageNet boosts accuracy on CIFAR-10 from 94% to 98% (Chen et al., 2020a,b). Achieving high in-distribution accuracy is not enough: high-stakes applications such as poverty mapping in under-resourced countries (Jean et al., 2016), self-driving cars (Yu et al., 2020), and medical diagnosis (AlBadawy et al., 2018), require models that also generalize to circumstances not seen in the training distribution. In addition to testing on data drawn from the downstream task's training distribution (in-distribution; ID), it is increasingly important to test on data distributions unseen during training (out-of-distribution; OOD). OOD accuracy can be much lower than ID accuracy; for example, an ImageNet pretrained ResNet-50 fine-tuned on CIFAR-10 gets 98% accuracy on CIFAR-10 (ID) but 82% on STL (OOD).

After initializing with a pretrained model, two popular transfer methods are fine-tuning (running gradient descent on all the model parameters), and linear probing (tuning the head but freezing lower layers). In the ID setting, it is well known that fine-tuning leads to better accuracy than linear probing (Kornblith et al., 2019;

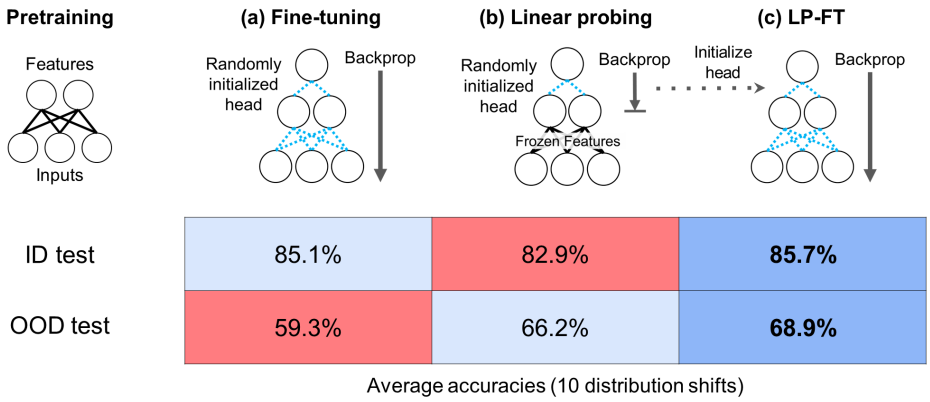


Figure 1: Given a good feature extractor (top-left), a randomly initialized head is added to map features to outputs and we can (a) fine-tune all the model parameters or (b) linear-probe, which freezes the feature extractor and trains only the head. We run experiments on ten distribution shifts. Fine-tuning does well when the test example is sampled from the fine-tuning distribution (ID), but can underperform on test examples sampled from OOD distributions (when the distribution shift is large). (c) Our theory indicates that fine-tuning can distort the pretrained feature extractor and lead to poor OOD accuracy, but initializing with a linear probed head can fix this—empirically LP-FT gets better accuracies both ID and OOD.

Zhai et al., 2020; He et al., 2020),[1] and even when testing OOD, prior work usually fine-tunes all parameters of their model (Hendrycks et al., 2019a; Miller et al., 2021; Andreassen et al., 2021). Intuitively, fine-tuning all layers of a network can improve pretrained features by adapting them to the specific task, while linear probing simply inherits the frozen pretrained features.

In this work, we investigate the OOD accuracy of fine-tuning and linear probing and find that surprisingly, fine-tuning can do *worse* than linear probing in the presence of large distribution shift. We experiment on ten distribution shift benchmarks (Breeds Living17, Breeds Entity30, DomainNet, CIFAR $\rightarrow$ STL, CIFAR10.1, FMoW geo-shift, ImageNetV2, ImageNet-R, ImageNet-A, ImageNet-Sketch), initializing with good pretrained features from MoCo-v2 (Chen et al., 2020b) and CLIP (Radford et al., 2021). While both methods offer gains over training from scratch, fine-tuning improves the average ID accuracy relative to linear probing from 83% to 85% but brings down the OOD accuracy from 66% to 59% (Figure 1).

Under what conditions does fine-tuning underperform linear probing? We theoretically consider fine-tuning a two-layer linear network in an overparameterized regression setting where the feature extractor layer has been pretrained to map high-dimensional inputs to useful, lower-dimensional, features. We prove that fine-tuning is worse than linear probing on directions outside the span of the training data when using "good" pretrained features. Even with an infinitesimally small learning rate, fine-tuning distorts pretrained features—the features of ID training data are updated while those of OOD data change less. Since the head and feature extractor are simultaneously optimized during fine-tuning to a configuration that works well on ID training data, the head only accomodates the distorted features of ID points and performs poorly (relative to linear probing) on the less changed features of OOD points. Interestingly, we show that this feature distortion issue cannot be simply fixed by early stopping—throughout the entire process of fine-tuning, we never pass through parameters that do well OOD (relative to linear probing). On the other hand, given "good" features, linear-probing

[1]Probing is commonly used but usually for interpretability or assessing feature quality.

extrapolates better OOD because it preserves pretrained features, but does not do as well as fine-tuning ID because linear probing cannot adapt the features to the downstream task.

Technical challenges. Existing theoretical work on transfer learning focuses on linear probing (Wu et al., 2020; Tripuraneni et al., 2020; Du et al., 2020). In contrast, analyses of fine-tuning is scarce and challenging because it requires understanding the training dynamics, instead of only the loss function and its global minimizers. In fact, fine-tuning and training from scratch optimize the *same* training loss and only differ in their initializations (pretrained vs random). A mathematical analysis that distinguishes them needs to capture properties of the different minima that these algorithms converge to, a phenomenon that is sometimes theoretically referred to as the implicit regularization effect of initialization (Neyshabur et al., 2014). Accordingly, our analysis reasons about the parameters that gradient methods pass through starting from the pretrained initialization, which is challenging because this is a non-convex optimization problem and there is no known closed form for this trajectory. Two-layer linear networks are widely studied in the literature on implicit regularization (Saxe et al., 2014; Gunasekar et al., 2017; Gidel et al., 2019; Arora et al., 2018). However, they analyze random and often small initializations, which don't capture pretraining.

Algorithmic implications. Our theory shows that fine-tuning underpeforms because when trying to fit ID training data with a randomly initialized head, the feature extractor changes significantly for ID examples, making features for ID and OOD examples largely inconsistent. This can be fixed by initializing with a good head that does not need to be updated much during fine-tuning, reducing how much the feature extractor changes. This suggests a simple two-step strategy of first linear-probing to find a good head and then full fine-tuning (LP-FT). *Empirically, LP-FT outperforms fine-tuning and linear-probing, both ID and OOD*. Even on CIFAR-10.1 (small distribution shift), where fine-tuning is better for both ID and OOD, we find LP-FT outperforms fine-tuning on both metrics. LP-FT and vanilla fine-tuning use similar amounts of compute because the first step of linear probing is relatively very cheap. Prior work has used LP-FT (Levine et al., 2016; Kanavati & Tsuneki, 2021) (or variants such as layerwise fine-tuning (Howard & Ruder, 2018) or larger learning rates for the head layer (Prabhu et al., 2021))—however it has not been used for robustness / OOD accuracy, and we show that it addresses the ID-OOD tradeoff theoretically and empirically. Note that LP-FT is not meant to be a SOTA method but a simple, principled way to get good ID and OOD accuracy—we hope our analysis inspires even better methods for robust fine-tuning.

Empirical validation. Finally, we check whether fine-tuning underperforms and LP-FT works, for the reasons predicted by our feature distortion theory. As predicted by the theory, we find that: (1) fine-tuning indeed never matches the OOD accuracy of linear probing throughout the course of training (if the pretrained features are good, and OOD shift is large); (2) fine-tuning changes the features for ID examples more than for OOD examples, leading to distortions; (3) LP-FT indeed changes both ID and OOD features $10\times - 100\times$ less than fine-tuning does; (4) fine-tuning can do better than linear probing OOD if the pretrained features are not very high quality (MoCo-v1 instead of MoCo-v2) or the ID and OOD datasets are very close (e.g., CIFAR-10 and CIFAR-10.1); and (5) LP-FT gets the best of both worlds, better accuracies than fine-tuning and linear probing, both ID and OOD (Figure 1).

2 Setup

Task and evaluation. Given training examples sampled from some distribution P_{id}, our goal is to learn a predictor $f : \mathbb{R}^d \to \mathcal{Y}$ to map inputs $x \in \mathbb{R}^d$ to outputs $y \in \mathcal{Y}$. We evaluate predictors on their standard "in-distribution" (ID) performance L_{id} on new test samples drawn from P_{id} that the training data is also

sampled from. We also evaluate classifiers on their "out-of-distribution" (OOD) performance L_{ood} on test samples drawn from a new distribution P_{ood} that is different from P_{id}. Formally, for some loss function ℓ, we evaluate classifiers on:

$$L_{\mathsf{id}}(f) = \mathop{\mathbb{E}}_{(x,y)\sim P_{\mathsf{id}}}[\ell(f(x), y)] \text{ and } L_{\mathsf{ood}}(f) = \mathop{\mathbb{E}}_{(x,y)\sim P_{\mathsf{ood}}}[\ell(f(x), y)]. \tag{2.1}$$

Models. In this work, we focus on predictors that leverage pretrained representations. We parameterize the final predictor f as follows: given features $g_B(x) \in \mathbb{R}^k$ for some feature extractor parameters $B \in \mathcal{B}$, and a linear "head" $v \in \mathcal{V}$, we have $f_{v,B}(x) = v^\top g_B(x)$. In our experiments (Section 4), g_B is a deep network and in our theory (Section 3), g_B is a linear projection.

We assume access to some initial pretrained feature extractor B_0 that is obtained by training on potentially large amounts of data from a distribution that contains unlabeled or weakly supervised x inputs from P_{id} and P_{ood}. We focus on two popular methods to learn a predictor $f_{v,B}$ given training data from P_{id}: (i) linear probing where $B = B_0$ and the linear head is obtained by minimizing some loss (e.g., logistic loss for classification, squared loss for regression) on the training data, and (ii) fine-tuning where both v and B are updated by performing gradient descent on some loss on the training data with B initialized at B_0.

3 Theory: fine-tuning distorts pretrained features

Our goal is to understand under what conditions fine-tuning does worse than linear probing out-of-distribution (OOD).[2] We consider a linear setting (feature extractor g_B is linear) where the pretrained features are "good" and the OOD shift is large (Section 3.1). We prove our main result: that fine-tuning, in which all model parameters are updated, distorts features and gets suboptimal OOD error (Section 3.2, Theorem 3.3). We use this result to show that linear probing gets better OOD error but worse ID error than fine-tuning (Section 3.3). Finally, we explain why linear probing then fine-tuning can mitigate this ID-OOD tradeoff (Section 3.4).

Our analysis handles two key challenges which distinguishes it from prior work on transfer learning in linear models (Wu et al., 2020; Tripuraneni et al., 2020; Du et al., 2020; Xie et al., 2021a). Prior work focuses on linear probing, while we study fine-tuning where the resulting optimization problem is *non-convex*. We also study *overparameterized models* where the training loss alone does not determine test performance—this captures the fact that both training neural networks from scratch and fine-tuning them have the same training loss but very different test performance. However, it also makes the analysis challenging because we need to reason about the trajectory of gradient methods starting from a pretrained initialization, which has no known closed form.

3.1 Linear overparameterized setting

For our analysis, we focus on regression, where $\mathcal{Y} = \mathbb{R}$ and $\ell(\widehat{y}, y) = (\widehat{y} - y)^2$ is the squared loss.

Models. Recall from Section 2 that we parameterize predictors in terms of feature extractor and head parameters. In this section, we study models where the feature extractor is linear, i.e. $f_{v,B}(x) = v^\top Bx$ where $B \in \mathcal{B} = \mathbb{R}^{k\times d}$, and $v \in \mathcal{V} = \mathbb{R}^k$.

[2]For example, without additional assumptions we can have $P_{\mathsf{id}} = P_{\mathsf{ood}}$ and so the same method will do better both ID and OOD.

Good pretrained features. For simplicity, we assume the models are well-specified i.e. $y = v_\star^\top B_\star x$ where $v_\star \in \mathbb{R}^k$ and $B_\star \in \mathbb{R}^{k \times d}$. [3] Note that $B_\star$ and $v_\star$ are only unique up to rotations, i.e., for any rotation matrix U, $(Uv_\star)^T(UB_\star)x = v_\star^T B_\star x$. As in prior work (Tripuraneni et al., 2020) suppose $B_\star, B_0$ have been orthogonalized to have orthonormal rows. Suppose we have a pretrained feature extractor B_0 close to $B_\star$, so $d(B_0, B_\star) \leq \epsilon$ where the distance d is defined below:

Definition 3.1 (Feature Extractor Distance)**.** *The distance between feature extractors $B, B' \in \mathbb{R}^{k \times d}$ (with orthonormal rows) is given by (where the min is over rotation matrices $U \in \mathbb{R}^{k \times k}$):*

$$d(B, B') = \min_U \|B - UB'\|_2, \tag{3.1}$$

Pretraining coverage intuition: Intuitively, the existence of $B_\star$ corresponds to assuming that there exists a shared set of useful features for ID (P_{id}) and OOD (P_{ood}). We also assume that B_0 is close to $B_\star$—one way this can happen is if pretraining is done on large scale data and has seen unlabeled or weakly supervised x inputs that cover the support of P_{id} and P_{ood}. Formally, the task diversity assumption in Tripuraneni et al. (2020) is sufficient (but not necessary) for obtaining a good B_0. In our paper we show that even if we have these good features, fine-tuning can distort them and lead to low OOD accuracy.

Training data. Let $X \in \mathbb{R}^{n \times d}, X \neq 0$ be a matrix encoding n training examples from P_{id} where each of the n rows is a training input. Let $Y \in \mathbb{R}^n$ be the corresponding outputs. Let $S = \text{rowspace}(X)$ be the m-dimensional subspace spanning the training examples. We consider an overparameterized setting where $1 \leq m < d - k$. Intuitively, the input dimension d is high (e.g., 10K), feature dimension k is lower (e.g., 100) and m is in the middle (e.g., 5K).[4]

Large OOD shift. We assume that the OOD data contains examples outside the span of the training data. Formally, let P_{ood} have second moment $\Sigma = \mathbb{E}[xx^\top]$ where $x \sim P_{\mathsf{ood}}$, and we assume Σ is invertible.[5][6]

Training methods. Given training data and a pretrained feature extractor B_0, we study the two popular methods of linear probing (LP) and fine-tuning (FT) to learn the final predictor. Both methods involve optimizing the training loss via gradient descent (or variants). In order to effectively analyze these gradient based algorithms, we study vanishing step sizes leading to gradient flows. Gradient flows can be thought of as a continuous time analogue of gradient based methods and have been extensively studied in recent years as a way to understand gradient based methods (Gunasekar et al., 2017; Arora et al., 2018; Du et al., 2018).

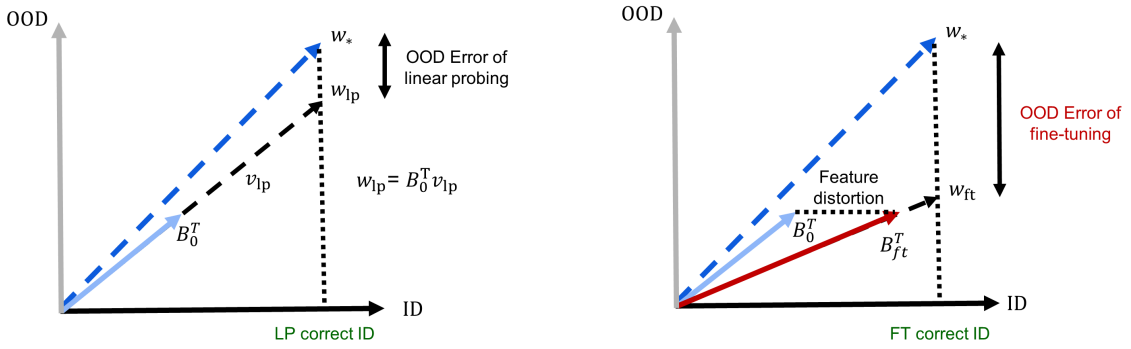
Formally, for training loss $\widehat{L}(v, B) = \|XB^\top v - Y\|_2^2$, the gradient flow differential equations for LP and FT

[3] We note that our main contribution—analysis of fine-tuning (Theorem 3.3)—does not require this well-specified assumption. We compare fine-tuning with linear probing by adapting earlier work on linear probing which requires well-specification.

[4] Indeed, in neural tangent kernel approximations, the input dimension d is the number of weights in a neural network which is much larger than the span of the training data m, while the feature dimension k of neural networks is usually smaller than m. Extending our results to the NTK regime could be an interesting future direction.

[5] We don't need Σ to be invertible, but just require the OOD span $T = \text{Range}(\Sigma)$ to have some directions outside the training span: $\dim(T \setminus S) > k$.

[6] Prior work on distribution shift (Rosenfeld et al., 2021; Kamath et al., 2021; Chen et al., 2021b) often considers a worst case loss over some set—we can equivalently write L_{ood} as a worst case loss over distributions (equivalently, individual points) of bounded norm: $\max_x (v_\star^\top B_\star x - v^\top Bx)^2$ over $x^\top \Sigma^{-1} x \leq 1$. If $\Sigma = I_d$ then this is just the worst case loss over $\|x\|_2 \leq 1$.



(a) Toy example (Linear probing)

(b) Toy example (fine-tuning)

Figure 2: A toy version of our theory illustrating why fine-tuning distorts features, with inputs in 2D. Given input x, the ground truth output is $y = w_\star^\top x$. The ID data is along the x-axis and the pretrained feature extractor is B_0. (a) Linear probing learns w_{lp}, a scaling of the pretrained feature extractor that gets the ID data correct (w_{lp} and $w_\star$ have the same x coordinate as indicated by the vertical dotted line). (b) Fine-tuning updates the pretrained feature extractor along the ID data (so horizontally) to get B_{ft}, and then learns a scaling of these features that gets the ID data correct. While both methods get ID data correct, fine-tuning makes large errors perpendicular to the ID data, because fine-tuning updates B_0 along the ID direction but not the perpendicular direction (we call this feature "distortion").

are as follows:

$$\partial_t v_{\text{ft}}(t) = -\nabla_v \widehat{L}(v_{\text{ft}}(t), B_{\text{ft}}(t)),\ \partial_t B_{\text{ft}}(t) = -\nabla_B \widehat{L}(v_{\text{ft}}(t), B_{\text{ft}}(t)), \tag{3.2}$$

$$\partial_t v_{\text{lp}}(t) = -\nabla_v \widehat{L}(v_{\text{lp}}(t), B_0),\ \partial_t B_{\text{lp}}(t) = 0, \tag{3.3}$$

initialized with $B_{\text{ft}}(0) = B_{\text{lp}}(0) = B_0$ and $v_{\text{ft}}(0) = v_{\text{lp}}(0) = v_0$. In practice, the head parameter v_0 is initialized randomly—our results hold for any standard random initialization (Glorot & Bengio, 2010), for example $v_0 \sim \mathcal{N}(0, \sigma^2 I)$ for any σ^2, or zero initialization where $v_0 = 0$. Recall that the initial value of the feature extractor B_0 is obtained via pretraining.

The final LP and FT solutions are the limit points of the corresponding gradient flows:

$$v_{\text{ft}}^\infty = \lim_{t\to\infty} v_{\text{ft}}(t) \text{ and } B_{\text{ft}}^\infty = \lim_{t\to\infty} B_{\text{ft}}(t), \tag{3.4}$$

$$v_{\text{lp}}^\infty = \lim_{t\to\infty} v_{\text{lp}}(t) \text{ and } B_{\text{lp}}^\infty = \lim_{t\to\infty} B_{\text{lp}}(t) = B_0. \tag{3.5}$$

3.2 Fine-tuning distorts pretrained features

The more common method of using a pretrained feature extractor is fine-tuning (FT) which typically improves ID performance relative to linear probing (LP). In this section, we show theoretically that FT can distort features leading to poor OOD performance. We first present the key intuitions demonstrating potential issues of FT and then present our formal theorem lower bounding the OOD error of FT (Section 3.2.2).

3.2.1 Key intuitions

There are two main observations that we use to characterize when and why FT has higher OOD error than linear probing.

1. Features get distorted: representations change only in the ID subspace (i.e., subspace spanned by the training data) and are unchanged in the orthogonal subspace. To see this, we take the derivative of the training loss $\widehat{L}(v, B) = \|XB^\top v - Y\|_2^2$ with respect to the feature extractor parameter B:

$$\nabla_B \widehat{L}(v, B) = 2v(Y - XBv)^\top X. \tag{3.6}$$

By definition, if u is a direction orthogonal to the training subspace $S = \text{rowspace}(X)$, then $\nabla_B \widehat{L}(v, B)u = 0$, that is the gradient updates to B do not modify Bu for $u \in S^\perp$. However, the gradient is non-zero for directions u in the ID subspace and the corresponding features Bu change across the fine-tuning process. We call this feature distortion: the features in some directions are changed but not others. Next, we explain why this can lead to high OOD error.

2. Distorted features can lead to higher OOD error. Consider a toy example (Figure 2) where $d = 2$ and the dimensionality of the representations $k = 1$. The linear head v is a scalar quantity that denotes how much the feature extractor B has to be scaled by. Suppose the ID-subspace is the x-axis. There are different ways of fitting the ID subspace depending on the feature extractors B as shown in the Figure—both fine-tuned and linear probed estimators match the true parameter in the ID subspace (since $w_{\mathsf{lp}}, w_{\mathsf{ft}}, w_\star$ have the same projection on the x-axis). If the feature extractor were optimal or scaled versions of the optimal, good performance on the ID subspace would translate to good performance everywhere, even in directions orthogonal to the ID subspace. However, in FT, the features change only for inputs in the ID subspace (see (1)) and thus the updated features are *not* simply scaled but distorted. In Figure 2, this corresponds to the feature extractor B_0 changing along the x-axis. In this case even if the ID error is low, error in directions orthogonal to the ID subspace can be high, leading to high OOD error.

The only way the pretrained features are not distorted and only scaled during FT is if the initial feature extractor B_0 is exactly aligned with the ID subspace. In Figure 2, if B_0 is along the x-axis (the ID subspace), then updating the features exclusively along the x-axis would simply scale the initial features. In this case linear probing and fine-tuning will have identical behavior. If the angle between B_0 and the x-axis is non-zero—which occurs with probability 1 if the training data X or pretrained feature extractor B_0 involves even a tiny amount of randomness e.g., from SGD in pretraining—the updates would lead to distortions. In high dimensions, we measure the alignment between B_0 and the ID subspace with the largest principal angle:

Definition 3.2 (largest principal angle)**.** *Let A and B be arbitrary subspaces, and E and F be matrices with orthonormal columns than span A and B respectively, with $r = \min(\dim(A), \dim(B))$. Then $\cos\theta_{\mathsf{max}}(A, B) = \sigma_r(E^\top F)$, which is the r-th largest singular value of $E^\top F$.*

Note that E, F are not unique in Definition 3.2, but $\sigma_r(E^\top F)$ is the same for every valid choice of E and F. See Appendix A.1 for more information on principal angles.

3.2.2 General result on the OOD error of fine-tuning

Our main theorem lower bounds the OOD error of fine-tuning outside the span of the training data. In Section 3.3 we compare this lower bound with an *upper bound* on the OOD error of linear probing.

Theorem 3.3. *In the overparameterized linear setting, let $S^\perp = \text{rowspace}(X)^\perp$, $R_0 = \text{rowspace}(B_0)$, and $v_\star, B_\star$ be the optimal parameters with $w_\star = B_\star v_\star$. If $\cos\theta_{\mathsf{max}}(R_0, S^\perp) > 0$, then for all time steps t, the*

OOD error of the fine-tuning iterates $(B_{\mathsf{ft}}(t), v_{\mathsf{ft}}(t))$ is lower bounded:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \geq \sqrt{\sigma_{\min}(\Sigma)}\Big(\frac{\cos\theta_{\mathsf{max}}(R_0, S^{\perp})}{\sqrt{k}}\frac{\min(\varphi, \varphi^2/\|w_\star\|_2)}{(1+\|w_\star\|_2)^2} - \epsilon\Big), \tag{3.7}$$

where $\varphi^2 = |(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2|$ is defined to be inital head alignment error and $\epsilon \geq d(B_0, B_\star)$ is the error in the pretrained feature extractor.

Proof sketch. Since the features do not change for examples in $S^\perp$ (perpendicular to the training data), we show that in order to achieve low error on $S^\perp$ the linear head $v_{\mathsf{ft}}(t)$ would have to become very similar to the optimal $v_\star$ at some time t. The head initialization v_0 is random (or zero) and likely to be far from $v_\star$ (measured by the alignment error φ), so the head would have to change a lot to get close to $v_\star$. As we see from the fine-tuning gradient flow (3.2), $v_{\mathsf{ft}}(t)$ and $B_{\mathsf{ft}}(t)$ change in a "coupled" manner, and a "'balancedness" invariant in Du et al. (2018) holds across the fine-tuning trajectory. Correspondingly, if $v_{\mathsf{ft}}(t)$ changes a lot and gets close to $v_\star$, the features $B_{\mathsf{ft}}(t)$ also change a lot for examples in S—we show that this would lead to high error on examples in S. Either way, fine-tuning would get some subspace (S or $S^\perp$) of examples wrong, leading to high OOD error. The full proof appears in Appendix A.

Interpretations of various quantities. *Quality of pretrained features (ϵ).* To unpack the bound consider a special case where the pretrained features are perfect ($\epsilon = 0$). With perfect features, Proposition A.20 shows that linear probing gets zero OOD error. Theorem 3.3 shows that $L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t)) > 0$ at all times t—so fine-tuning underperforms when the features are perfect. The $\epsilon > 0$ case just captures the fact that even if the features are not perfect, fine-tuning can still get positive error. Ideally we would like the lower bound to *increase* if we have worse features (so "$+\epsilon$" instead of "$-\epsilon$" in the bound)—the reason we do not is that the errors of the pretrained feature extractor $d(B_0, B_\star)$ and the fine-tuning step can potentially cancel out.[7]

Alignment error of random head initialization (φ^2). The lower bound (Equation A.14) increases as φ^2 increases i.e. alignment error increases because the gradient updates to the head and feature extractor are coupled. If the head were somehow initialized perfectly at $v_\star$, then fine-tuning updates may not increase the OOD error. However, when the head is randomly initialized (or initialized to zero) as is standard in fine-tuning, the alignment error is high, leading to high OOD error. We use this insight in Section 3.4 to show that better head initialization (namely via linear probing) improves OOD performance of fine-tuning.

Span of Training data (S). Theorem 3.3 lower bounds the error outside the span of the training data. If the training dataset is very small, then even the support of the ID distribution P_{id} may not be spanned by the training data, and the ID error can be large. Indeed, even in the ID setting Kornblith et al. (2019) show that linear probing can do better than fine-tuning if we have very few training examples, but fine-tuning does better on all 11 of their datasets once we have more than just 30 examples per class.

Conjectures for improved bounds. We believe it may be possible to improve $\cos\theta_{\mathsf{max}}(R_0, S^\perp)$ to the *cosine* of the *minimum* principal angle.[8] This may look like a technicality but would be a substantial improvement, because it would imply that fine-tuning has error in *every* direction outside the training span, whereas we show that it would have errors in *some* directions. Our proof strategy requires the maximum principal angle (a crucial step is a variational characterization of the maximal principal angle in Lemma A.2—we use this in Step 1 of the proof in Appendix A to show that to get low OOD error $v_{\mathsf{ft}}(t)$ must become similar to $v_\star$).

[7] Intuitively this cancelation is very "unlikely" to happen, and we hope future work can capture this intuition.

[8] Which would be a larger/better lower bound since the cosine of a smaller quantity is larger.

3.3 Linear probing vs. fine-tuning

In this section, we use our main theorem on fine-tuning (Theorem 3.3) and adapt prior work on linear probing to show that linear probing is better than fine-tuning OOD, but worse ID, when the ID distribution has density on a lower $m < d$ dimensional subspace S, and B_0 is close to $B_\star$ (so we have "good" pretrained features).

Assumption 3.4 (ID subspace assumption)**.** *We assume that the ID data lies on an m-dimensional subspace S where $k < m < d - k$, and we have $n \geq m$ training examples. Formally, let P_z be a distribution on $\mathbb{R}^m$ which has density, and let the columns of $F \in \mathbb{R}^{d \times m}$ form an orthonormal basis for S. Then P_{id} has the distribution of Fz where $z \sim P_z$.*

Recall that the ID error is the expected mean-squared error over the ID distribution P_{id}:

$$L_{\mathsf{id}}(v, B) = \mathop{\mathbb{E}}_{x \sim P_{\mathsf{id}}}[(v_\star^\top B_\star x - v^\top B x)^2] \tag{3.8}$$

OOD comparison: Under mild non-degeneracy conditions, we show that as the feature extractor error ϵ goes to 0, linear probing does much better than fine-tuning OOD: the ratio of the losses goes to 0. The non-degeneracy conditions are similar to Section 3.2—we require that the training data cannot be exactly in the same direction or orthogonal to the pretrained features, formally that $\cos\theta_{\mathsf{max}}(R_*, S)$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp)$ are not 0 where $R_* = \text{rowspace}(B_\star)$. In the toy example in Figure 2, this means that x_{id} cannot be exactly in the same direction or orthogonal to $B_0^\top$—in these cases fine-tuning and linear probing get the same loss but in all other cases in the toy example in Figure 2 linear probing does better OOD.

Theorem 3.5 (Informal version of Theorem A.8)**.** *In the linear overparameterized setting, under the ID subspace assumption (Assumption 3.4), if $\cos\theta_{\mathsf{max}}(R_*, S) \neq 0$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp) \neq 0$ where $R_* = \text{rowspace}(B_\star)$, then,*

$$\frac{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)}{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \xrightarrow{p} 0, \text{as } \; B_0 \to B_\star. \tag{3.9}$$

This holds for all times t for FT (and therefore also for the limit $v_{\mathsf{ft}}^\infty, B_{\mathsf{ft}}^\infty$) and the LP iterates converge to $v_{\mathsf{lp}}^\infty, B_0$ as a result of the gradient flow on a convex problem.

Intuitively, if the pretrained features are good, LP learns a near optimal linear head which has small OOD error (Lemma A.14) but fine-tuning has high OOD error (Theorem 3.3). We give a more formal version of Theorem 3.5 and a proof in Appendix A.3. If P_z is isotropic Gaussian, we can get a better result: Theorem A.15 derives a threshold T (in terms of d, n, k) where LP does better than FT if $\epsilon < T$, instead of just the asymptotic result ($B_0 \to B_\star$). Theorem 3.5 requires that $\cos\theta_{\mathsf{max}}(R_*, S) \neq 0$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp) \neq 0$—intuitively, for any subspace a small perturbation would make these angles non-zero and the assumption would hold. To illustrate that these assumptions typically hold, Lemma A.16 in Appendix A proves that if S is a random m-dimensional subspace then these angles are non-zero almost surely.

ID comparison: When the pretrained features have some error, we show that fine-tuning does better than linear probing ID because fine-tuning can update the features to fit the ID data.

If the pretrained features are perfect so that the optimal predictor can be written as a linear combination of the pretrained features ($w_\star = B_\star^\top v_\star \in \text{rowspace}(B_0)$), then both linear probing and fine-tuning get zero ID error. However, if the pretrained representation has some error, and the training data satisfies a mild non-degeneracy

condition, then LP has high ID error because there is no linear head on B_0 that fits the training data perfectly. FT, on the other hand, can update the features to find a new B_{ft}^{∞} that can fit the training data perfectly with a linear head v_{ft}^{∞}.

The non-degeneracy condition is similar to our previous results, and holds with probability 1 if the ID subspace is chosen randomly, from Lemma A.16. Formally, let R_{aug} be a $k+1$ dimensional subspace spanning $R_0 \cup \{w_\star\}$, where we recall that $R_0 = \text{rowspace}(B_0)$. Then we just require that the ID subspace S and R_{aug} are not orthogonal: $\cos\theta_{\mathsf{max}}(S, R_{\mathsf{aug}}) \neq 0$. We state the formal proposition below and give a proof in Appendix A.

Proposition 3.6. *In the linear overparameterized setting, under the ID subspace assumption (Assumption 3.4), let $R_0 = \text{rowspace}(B_0)$, and $R_{\mathsf{aug}} = \text{Span}(\{w_\star\} \cup R_0)$. Suppose $w_\star \notin R_0$, $\cos\theta_{\mathsf{max}}(S, R_{\mathsf{aug}}) \neq 0$, and that fine-tuning converges to a local minimum of its loss, then fine-tuning does better ID almost surely: $L_{\mathsf{id}}(v_{\mathsf{ft}}^{\infty}, B_{\mathsf{ft}}^{\infty}) < L_{\mathsf{id}}(v_{\mathsf{lp}}^{\infty}, B_0)$ with probability 1 (over the randomness of the training examples).*

To summarize, we proved that there are tradeoffs between ID and OOD error: FT has lower ID error but higher OOD error than LP. In the next section, we extend our theoretical insights to illustrate why a simple variant of FT may mitigate such tradeoffs.

3.4 Linear probing then fine-tuning: a simple variant to mitigate tradeoffs

The advantage of fine-tuning is it can adapt both the feature extractor and head to fit the downstream task. Can we keep this benefit while ensuring that our OOD error is low when we have good pretrained features?

Going back to Theorem 3.3, we see that the alignment error in the head initialization $\varphi^2 = (v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2$ plays an important role. The issue with FT was that under random or zero initialization, φ^2 is usually large and since the gradient updates to the feature extractor parameter are coupled with that of the head parameter, the features get distorted in a manner that increases the OOD error. This suggests that we should use a better head initialization—one obtained from linear probing. If the pretrained features are decent, a linear probed head would be much better aligned with $v_\star$ allowing the features to be updated in a manner that does not increase the OOD error much.

We formally prove this intuition in a simple setting where we have perfect pretrained features. Of course, if we have perfect pretrained features, linear probing alone gets zero OOD error—so Proposition 3.7 is just a first cut result to illustrate that if initialized well, full fine-tuning does not distort features.

Proposition 3.7. *Suppose we have perfect pretrained features $B_0 = UB_\star$ for some rotation U. Let $R_0 = \text{rowspace}(B_0)$. Under the non-degeneracy conditions $\cos\theta_{\mathsf{max}}(R_0, S) \neq 0, \cos\theta_{\mathsf{max}}(R_0, S^\perp) \neq 0$:*

$$\forall t, L_{\mathsf{ood}}(B_{\mathsf{ft}}(t)^\top v_{\mathsf{ft}}(t)) > 0, \text{ if } v_0 \sim \mathcal{N}(0, \sigma^2 I) \text{ is randomly initialized (FT)}, \tag{3.10}$$

$$\forall t, L_{\mathsf{ood}}(B_{\mathsf{ft}}(t)^\top v_{\mathsf{ft}}(t)) = 0, \text{ if } v_0 \text{ is initialized to } v_{\mathsf{lp}}^{\infty} \text{ (LP-FT)}. \tag{3.11}$$

The case where we do not have perfect features ($d(B_0, B_\star) > 0$) is challenging to analyze because except in very special cases, there is no closed form for the fine-tuning iterates $(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))$. Our proof of Theorem 3.3 leveraged invariants to show a *lower bound* on the error of fine-tuning when v_0 and $v_\star$ are different, but we were not able to show an *upper bound*.

4 Experiments

We run experiments on ten distribution shifts to see if our theoretical predictions on the relative performance of linear probing (LP), fine-tuning (FT), and LP-FT, generalize to deep neural networks on real datasets. As expected, given good pretrained features, fine-tuning (FT) does better ID but worse on large OOD shifts than linear probing (LP). In particular, ID and OOD accuracy are not correlated, unlike Recht et al. (2018) but like Xie et al. (2021a). As predicted by the theory, we find that LP-FT does better than both methods ID and OOD and gets around this tradeoff. Our theory also predicts that *the reason* for these trends is that fine-tuning distorts features, and we see that this distortion indeed happens in practice. For more details on datasets, pretraining models, and experiment protocols, see Appendix B. The datasets we use are:

- **DomainNet** (Peng et al., 2019) is a standard domain adaptation dataset. Here, our ID dataset contains "sketch" images (e.g., drawings of apples, elephants, etc), and the OOD dataset contains "real", "clipart", and "painting" images of the same categories. We use the version of the dataset from Tan et al. (2020).

- **Living-17** and **Entity-30** are sub-population shift datasets from the BREEDS benchmark (Santurkar et al., 2020). In Living-17 the goal is to classify an image as one of 17 animal categories such as "bear"—for example, the ID dataset contains images of black bears and sloth bears and the OOD dataset has images of brown bears and polar bears. In Entity-30 the goal is to classify an image as one of 30 entities such as "fruit" or "insect".

- **FMoW Geo-shift** is adapted from the satellite remote sensing dataset *Functional Map of the World* (Christie et al., 2018; Koh et al., 2021). The goal is to classify a satellite image into one of 62 categories such as "impoverished settlement" or "hospital". Our ID dataset contains images from North America, and the OOD dataset contains images from Africa and Europe.

- **CIFAR-10 $\rightarrow$ STL** is a standard domain adaptation dataset (French et al., 2018), where the ID is CIFAR-10 (Krizhevsky, 2009), and the OOD is STL (Coates et al., 2011). The task is to classify an image into one of 10 categories such as "dog", "cat", or "airplane"—as usual, we remove the "monkey" class in STL since CIFAR-10 has no "monkey" images.

- **CIFAR-10 $\rightarrow$ CIFAR-10.1** (Recht et al., 2018) is a dataset collected using a very similar protocol to CIFAR-10, and the authors describe it as "a minute distributional shift". The hope is that a classifier trained on CIFAR-10 gets high accuracy on CIFAR-10.1.

- **ImageNet**-1K (Russakovsky et al., 2015) is a large scale dataset containing over a million images, where the goal is to classify an image into one of 1000 categories such as "Yorkshire terrier", "Labrador retriever", "acoustic guitar", "library", "school bus", etc. We fine-tune on ImageNet as the ID dataset, and evaluate on four standard OOD datasets: **ImageNetV2** (Recht et al., 2019), **ImageNet-R** (Hendrycks et al., 2020), **ImageNet-A** (Hendrycks et al., 2019b), and **ImageNet-Sketch** (Wang et al., 2019).

Pretraining and models. We use a CLIP pretrained ViT-B/16 for ImageNet. For the other datasets we use a ResNet-50 architecture and consider a diverse range of pretraining methods and datasets: MoCo-v2 (Chen et al., 2020b), CLIP (Radford et al., 2021), and MoCo-TP (Ayush et al., 2020). In Appendix B, we also show results for a CLIP-ViT-B/16 and more fine-tuning baselines on Living-17.

Table 1: **ID accuracies** with 90% confidence intervals over 3 runs—fine-tuning does better than linear probing on all datasets except DomainNet (which could be because the version of the DomainNet training dataset from Tan et al. (2020) is fairly small, with around 20K examples). LP-FT does the best on all except FMoW where it is in between linear probing and fine-tuning.

	CIFAR-10	Ent-30	Liv-17	DomainNet	FMoW	ImageNet	Average
FT	**97.3 (0.2)**	**93.6 (0.2)**	97.1 (0.2)	84.5 (0.6)	**56.5 (0.3)**	**81.7 (-)**	85.1
LP	91.8 (0.0)	90.6 (0.2)	96.5 (0.2)	89.4 (0.1)	49.1 (0.0)	79.7 (-)	82.9
LP-FT	**97.5 (0.1)**	**93.7 (0.1)**	**97.8 (0.2)**	**91.6 (0.0)**	51.8 (0.2)	**81.7 (-)**	**85.7**

4.1 Linear probing vs. fine-tuning

Experiment protocols. We initialize with the pretrained model, and fine-tune or linear probe on ID training examples. For fine-tuning on each dataset we swept over 6 learning rates, using a cosine learning rate schedule and batch size of 64. We early stop and choose the best learning rate using ID validation accuracy. For linear probing we train an ℓ_2-regularized logistic regression classifier on frozen features from the penultimate layer of the pretrained model, selecting the best ℓ_2-regularization hyperparameter based on ID validation accuracy. For all methods, we run each hyperparameter configuration 3 times (with different random seeds), and take the average accuracy. We used a slightly different protocol for ImageNet because the dataset is much larger and running these experiments involves more computational resources: we used a batch size of 128, swept over 3 learning rates for both fine-tuning and linear probing (we did not sweep over ℓ_2-regularization), and ran each hyperparameter configuration once. In all cases, OOD data was only used for evaluation.

Results. Fine-tuning does better than linear probing on 5 out of 6 ID datasets (average accuracy of 85.1% for fine-tuning vs. 82.9% for linear probing, see Table 1). This is consistent with prior work and intuitions. However, linear-probing does better on 8 out of 10 OOD datasets (average accuracy of 66.2% for linear probing vs. 59.3% for fine-tuning, see Table 2)—linear probing does better on all datasets except CIFAR-10.1 and ImageNetV2, where the OOD is designed to closely replicate the ID dataset. This matches our theoretical predictions, which says that linear probing does better than fine-tuning when the ID and OOD are very different (and the pretrained features are "good"). Our training datasets vary in size from 20K examples to over a million examples, so linear probing does not appear to perform better than fine-tuning simply because of a small training set.

4.2 Linear probing then fine-tuning (LP-FT)

Experiment protocols. For LP-FT, we initialize the neural network head using the linear probed solution, and then fine-tune the model. LP-FT and fine-tuning use similar compute because the linear probing step is much faster than fine-tuning. As with fine-tuning, we swept over 6 learning rates, early stopping using ID validation accuracy. For the ImageNet experiments we swept over 3 learning rates, and explicitly ensured that LP-FT and fine-tuning use exactly the same compute (we ran each stage of LP-FT for half as many epochs as we ran vanilla fine-tuning).

Results. We find that LP-FT gets the best accuracy ID (average: 85.7%) and OOD (average: 68.9%). This is true for 5/6 ID and 10/10 OOD datasets—every dataset except FMoW ID, where LP-FT is better than linear probing but worse than fine-tuning. Since the ID accuracy on FMoW is low (56.5%), this could be because

Table 2: **OOD accuracies** with 90% confidence intervals over 3 runs. Linear probing does better than fine-tuning on all datasets except CIFAR-10.1 and ImageNetV2, where the ID and OOD are very similar (this is consistent with our theory). LP-FT matches or exceeds fine-tuning and linear probing on all 10 datasets.

	STL	CIFAR-10.1	Ent-30	Liv-17	DomainNet	FMoW
FT	82.4 (0.4)	92.3 (0.4)	60.7 (0.2)	77.8 (0.7)	55.5 (2.2)	32.0 (3.5)
LP	85.1 (0.2)	82.7 (0.2)	**63.2 (1.3)**	82.2 (0.2)	79.7 (0.6)	**36.6 (0.0)**
LP-FT	**90.7 (0.3)**	**93.5 (0.1)**	**62.3 (0.9)**	**82.6 (0.3)**	**80.7 (0.9)**	**36.8 (1.3)**

	ImNetV2	ImNet-R	ImNet-Sk	ImNet-A	Average
FT	**71.5 (-)**	52.4 (-)	40.5 (-)	27.8 (-)	59.3
LP	69.7 (-)	70.6 (-)	46.4 (-)	45.7 (-)	66.2
LP-FT	**71.6 (-)**	**72.9 (-)**	**48.4 (-)**	**49.1 (-)**	**68.9**

the pretrained features are not good.

4.3 Examining the feature distortion theory

Early stopping does not mitigate feature distortion. One might think that fine-tuning is simply overfitting ID, and so early stopping on OOD data (if it were available) might match linear probing OOD. However, our theory predicts that fine-tuning can do worse OOD (than linear probing) throughout the process of fine-tuning, and not just at the end. To test this, we early stop each fine-tuning method and choose the best learning rate based on OOD test accuracy (OOD data was not used except for this ablation). As expected, fine-tuning does improve a little, but linear probing (average accuracy: 67.1%) is still better than fine-tuning (average accuracy: 61.3%). See Appendix B for per-dataset results.

ID-OOD features get distorted from fine-tuning. The feature distortion theory predicts that fine-tuning changes features for ID examples more than for OOD examples, which is why fitting a head on ID examples performs poorly OOD. To test this, for each example x in Living-17 (results for other datasets are in Appendix B), we took the Euclidean distance of the ResNet-50 features before and after fine-tuning: $\|g_B(x) - g_{B_0}(x)\|_2$. As expected, the average distance for ID examples (0.0188 ± 0.0001) is more than for OOD examples (0.0167 ± 0.0001). The theory also predicts that LP-FT changes features less than fine-tuning does. As expected, the average distance changed by LP-FT both ID (0.0011 ± 0.0001) and OOD (0.0009 ± 0.0001) is $20\times$ smaller than for fine-tuning.

Pretrained features must be good, ID-OOD far apart. Our theory gives *conditions* under which linear probing can do better than fine-tuning OOD. Specifically, we require that the ID distribution P_{id} and OOD distribution P_{ood} are quite different, and the pretrained features are good (B_0 is close to $B_\star$)—otherwise fine-tuning can do better OOD by adjusting the feature extractor ID. Here we test that these conditions are essential—when they are violated fine-tuning can do better than linear probing OOD.

Feature quality: We use a checkpoint of MoCo-v1 that got 10% worse accuracy (on ImageNet) and compare linear probing and fine-tuning on Living-17. With worse features, both methods do worse, but fine-tuning (96% ID, 71% OOD) does better than linear probing (92% ID, 66% OOD).

ID $\approx$ *OOD*: We fine-tune / linear probe on CIFAR-10, and test on CIFAR-10.1, a dataset collected using a similar protocol to CIFAR-10. As expected, fine-tuning (92.3%) outperforms linear probing OOD (82.7%). Even in this case, where we have no tradeoffs, LP-FT does the best (93.5%).

5 Related work and discussion

Fine-tuning vs. linear probing. Fine-tuning (FT) and linear probing (LP) are popular transfer learning algorithms. There is substantial evidence of FT outperforming LP in-distribution (ID) including recent large-scale investigations (Kornblith et al., 2019; Chen et al., 2021a; Zhai et al., 2020; Chen et al., 2020b) (the only notable exception is in Peters et al. (2019) where LP performs better than FT when using ELMo representations, but worse using BERT). [9] FT is therefore the method of choice for improving accuracy, while LP is used to analyze properties of representations (Peters et al., 2018; Belinkov et al., 2017; Hewitt & Manning, 2019). In our work, we find that FT can underperform LP especially when using high quality pretrained features in the presence of a large distribution shift. There are a variety of other fine-tuning heuristics (Ge & Yu, 2017; Guo et al., 2019; Zhang et al., 2020; Zhu et al., 2020; Jiang et al., 2021; Aghajanyan et al., 2021)—combining our insights with these ideas might lead to better methods.

The benefit of preserving pretrained features. Our work adds to growing evidence that *lightweight* fine-tuning, where only a small part of a pretrained model are updated, performs better under distribution shifts—and we give a theoretical grounding to why this might be the case. Zero-shot language prompting in vision (Radford et al., 2021) and other lightweight fine-tuning approaches in NLP (Houlsby et al., 2019; Li & Liang, 2021; Xie et al., 2021b; Lester et al., 2021; Utama et al., 2021; Zhou et al., 2021) have been shown to improve OOD performance. In independent and concurrent work, Andreassen et al. (2021) observe that through the course of fine-tuning, ID accuracy continues to increase but OOD accuracy plateaus. Our work shows something stronger: at no point in the fine-tuning process does FT outperform LP.

Mitigating ID-OOD tradeoffs. While LP-FT has sometimes been used as a fine-tuning heuristic (Levine et al., 2016; Kanavati & Tsuneki, 2021; fastai), it has not been used for robustness / OOD accuracy, and we show that it addresses the ID-OOD tradeoff theoretically and empirically. Tradeoffs between ID and OOD accuracy are widely studied and prior work self-trains on large amounts of unlabeled data to mitigate such tradeoffs (Raghunathan et al., 2020; Xie et al., 2021a; Khani & Liang, 2021). In contrast, LP-FT uses no extra unlabeled data and is a simple variant of fine-tuning. In concurrent and independent work, Wortsman et al. (2021) show that ensembling the weights of a zero-shot and fine-tuned model mitigates the ID-OOD tradeoff between these approaches, and this method could be promising for our datasets as well.

Theoretical analysis of transfer learning. Prior works on transfer learning mainly analyze linear probing (Wu et al., 2020; Tripuraneni et al., 2020; Du et al., 2020). In recent work, (Chua et al., 2021) study regularized fine-tuning in an underparameterized regime where there is a unique global optimum. In contrast, our analysis studies the overparameterized regime (mirroring modern settings of zero train loss) where we need to analyze the trajectory of fine-tuning from the pretrained initialization because there is no unique optimizer of the objective function. Prior works also focus on ID error, while we analyze OOD error. See Section C for additional related work on theory of overparameterized models.

[9] This is not intended to be a comprehensive list. There is a large body of past work across different domains that have reported a similar observation.

6 Conclusion.

There is a strong trend towards leveraging pretrained models to improve downstream performance, and whenever feasible, it is common to fine-tune all model parameters. In this work, we show theoretically and empirically that preserving features might be important for robustness, and simpler approaches like linear-probing can improve out-of-distribution (OOD) performance. *This OOD gap between fine-tuning and linear probing grows as the quality of pretrained features improve, so we believe our results are likely to gain significance over time with growing innovations and scale of pretraining*.

Theoretical understanding of modern deep learning remains limited, especially the effect of pretraining and transfer learning. In addition to our specific results on fine-tuning, our work introduces some tools and ideas for dealing with the main challenge of characterizing properties of the trajectory from a specific initialization in the presence of multiple global optima (implicit regularization effect of initialization). There are several open questions and extensions such as dealing with non-linear activations, different layerwise learning rates, and the effect of explicit regularization. [10]

Finally, we showed LP-FT can mitigate tradeoffs between ID and OOD accuracy in our context. LP-FT could be useful in other situations, for example in CLIP we could initialize the final layer with the zero-shot classifier and then fine-tune the entire model, as done in concurrent work (Wortsman et al., 2021). LP-FT is just a first step in leveraging the intuition from our theoretical analysis and we hope that this work inspires new methods of leveraging powerful pretrained models.

Proofs and Reproducibility: We include proofs for our theoretical results in Appendix A and additional experiment details in Appendix B.

Acknowledgements: We would like to thank Kumar Ayush and Burak Uzkent for MoCo checkpoints pretrained on unlabeled FMoW images, Nilesh Tripuraneni for clarifications on his work and references on principal angles, Daniel Levy for useful suggestions on experiments to run, Niladri Chatterji, Jeff Z. HaoChen, and Colin Wei for useful papers and comments on figures, Niladri Chatterji and Kaidi Cao for reviewing the paper at ML paper swap, Kevin Yang for his help with analyzing differential equations, Tri Dao and Pang Wei Koh for help with writing, Suriya Gunasekar, Adam Kalai, Simon Kornblith, Ting Chen, Sang Michael Xie, Albert Gu, and Kendrick Shen for useful discussions, and Pang Wei Koh, Niladri Chatterji, and Tri Dao for suggestions on framing our results better.

Ananya Kumar was supported by the Rambus Corporation Stanford Graduate Fellowship. Percy Liang was supported by the Open Philantropy Project and NSF Award Grant No. 1805310. Aditi Raghunathan was supported by a Google PhD Fellowship and Open Philanthropy Project AI Fellowship. Tengyu Ma acknowledges support of a Google Faculty Award, NSF IIS 2045685, the Sloan Fellowship, JD.com, SAIL, and SDSI.

[10] We found that LP-FT outperforms explicit regularization and using a higher learning rate for the linear layer on Living-17 (Appendix B.4), but a more extensive theoretical and empirical study on this is important.

References

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. Better fine-tuning by reducing representational collapse. In *International Conference on Learning Representations (ICLR)*, 2021.

EA AlBadawy, A Saha, and MA Mazurowski. Deep learning for segmentation of brain tumors: Impact of cross-institutional training and testing. *Med Phys.*, 45, 2018.

Anders Andreassen, Yasaman Bahri, Behnam Neyshabur, and Rebecca Roelofs. The evolution of out-of-distribution robustness throughout fine-tuning. *arXiv*, 2021.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International Conference on Machine Learning (ICML)*, pp. 244–253, 2018.

Kumar Ayush, Burak Uzkent, Chenlin Meng, Kumar Tanmay, M. Burke, D. Lobell, and Stefano Ermon. Geography-aware self-supervised learning. *arXiv*, 2020.

Peter L. Bartlett, Philip M. Long, G´abor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *arXiv*, 2019.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do neural machine translation models learn about morphology? In *Association for Computational Linguistics (ACL)*, pp. 861–872, 2017.

Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *arXiv*, 2019.

Koby Bibas, Yaniv Fogel, and Meir Feder. A new look at an old problem: A universal learning approach to linear regression. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 2304–2308, 2019.

Tianle Cai, Ruiqi Gao, J. Lee, and Qi Lei. A theory of label propagation for subpopulation shift. In *International Conference on Machine Learning (ICML)*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pp. 1597–1607, 2020a.

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

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021a.

Yining Chen, Elan Rosenfeld, Mark Sellke, Tengyu Ma, and Andrej Risteski. Iterative feature matching: Toward provable domain generalization with logarithmic environments. *arXiv*, 2021b.

Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Kurtland Chua, Qi Lei, and Jason D Lee. How fine-tuning allows for effective meta-learning. *arXiv preprint arXiv:2105.02221*, 2021.

Adam Coates, Andrew Ng, and Honlak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pp. 215–223, 2011.

Simon S. Du, Wei Hu, Sham M. Kakade, Jason D. Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv*, 2020.

Simon Shaolei Du, Wei Hu, and Jason Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

fastai. fastai tutorial on transfer learning. `https://github.com/fastai/course-v3/blob/master/nbs/dl1/lesson1-pets.ipynb`.

Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

Weifeng Ge and Yizhou Yu. Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

Gauthier Gidel, Francis R. Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in deep linear neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, 2010.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 2013.

Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6151–6159, 2017.

Yunhui Guo, Honghui Shi, Abhishek Kumar, Kristen Grauman, Tajana Rosing, and Rogerio Feris. Spottune: Transfer learning through adaptive fine-tuning. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019.

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

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning (ICML)*, 2019a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *arXiv preprint arXiv:1907.07174*, 2019b.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *Association for Computational Linguistics (ACL)*, 2019.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. *arXiv*, 2019.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *Association for Computational Linguistics (ACL)*, 2018.

Neal Jean, Marshall Burke, Michael Xie, W. Matthew Davis, David B. Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353, 2016.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Pritish Kamath, Akilesh Tangella, Danica J. Sutherland, and Nathan Srebro. Does invariant risk minimization capture invariance? In *Artificial Intelligence and Statistics (AISTATS)*, 2021.

Fahdi Kanavati and Masayuki Tsuneki. Partial transfusion: on the expressive influence of trainable batch norm parameters for transfer learning. In *Medical Imaging with Deep Learning*, 2021.

Fereshte Khani and Percy Liang. Removing spurious features can hurt accuracy and affect groups disproportionately. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2021.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

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

Thomas Laurent and James H. von Brecht. Deep linear neural networks with arbitrary loss: All local minima are global. In *International Conference on Machine Learning (ICML)*, 2018.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

S. Levine, Chelsea Finn, Trevor Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research (JMLR)*, 17, 2016.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Association for Computational Linguistics (ACL)*, 2021.

Xuhong Li, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International Conference on Machine Learning (ICML)*, 2018.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*, 2019.

John Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning (ICML)*, 2021.

Vidya Muthukumar, Kailas Vodrahalli, Vignesh Subramanian, and Anant Sahai. Harmless interpolation of noisy data in regression. *IEEE Journal on Selected Areas in Information Theory*, 1(1):67–83, 2020.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv*, 2014.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *International Conference on Computer Vision (ICCV)*, 2019.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *North American Association for Computational Linguistics (NAACL)*, 2018.

Matthew E Peters, Sebastian Ruder, and Noah A Smith. To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pp. 7–14, 2019.

Viraj Prabhu, Shivam Khare, Deeksha Karthik, and Judy Hoffman. Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *International Conference on Computer Vision (ICCV)*, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, volume 139, pp. 8748–8763, 2021.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John C. Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2020.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do CIFAR-10 classifiers generalize to CIFAR-10? *arXiv*, 2018.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning (ICML)*, 2019.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Mark Rudelson and Roman Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics*, 62:1707–1739, 2009.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej

Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. *arXiv*, 2020.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv*, 2014.

Shuhan Tan, Xingchao Peng, and Kate Saenko. Class-imbalanced domain adaptation: An empirical odyssey. *arXiv preprint arXiv:1910.10320*, 2020.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *arXiv preprint arXiv:2007.00644*, 2020.

Nilesh Tripuraneni, Michael I. Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *arXiv*, 2020.

Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8:1–230, 2015.

Prasetya Ajie Utama, Nafise Sadat Moosavi, Victor Sanh, and Iryna Gurevych. Avoiding inference heuristics in few-shot prompt-based finetuning. *arXiv preprint arXiv:2109.04144*, 2021.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Mitchell Wortsman, Gabriel Ilharco, Mike Li, Jong Wook Kim, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. *arXiv preprint arXiv:2109.01903*, 2021.

Sen Wu, Hongyang R. Zhang, and Christopher Ré. Understanding and improving information transfer in multi-task learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-N-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. In *International Conference on Learning Representations (ICLR)*, 2021a.

Sang Michael Xie, Tengyu Ma, and Percy Liang. Composed fine-tuning: Freezing pre-trained denoising autoencoders for improved generalization. In *International Conference on Machine Learning (ICML)*, 2021b.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2020.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, Lucas Beyer, Olivier Bachem, Michael Tschannen, Marcin Michalski, Olivier Bousquet, Sylvain Gelly, and Neil Houlsby. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv*, 2020.

Jeffrey O Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. Side-tuning: A baseline for network adaptation via additive side networks. In *European Conference on Computer Vision (ECCV)*, 2020.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134*, 2021.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. FreeLB: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2020.

A Proofs for Section 3

A.1 Preliminaries on Important Notations and Principal Angles

Big-Oh Notation: For convenience, we use big-oh notation in a way that differs from standard theoretical computer science texts. When we say $O(\text{<expr1>})$ we mean that this can be replaced by c <expr1> for some universal constant such that the statement holds. As an example, we can say $5x^2 \leq O(x^2)$ because there exists some universal constant ($c = 5$) such that $5x^2 \leq 5x^2$. More examples: we can also say $5x^2 \geq O(x^2)$ or if $x \geq 1$ then $7x^2 \leq O(x^3)$ and $0.1x^2 \geq O(x)$.

Singular Values: Given a rectangular matrix $A \in \mathbb{R}^{m \times n}$, let $r = \min(m, n)$. The minimum singular value is defined as the r-th largest singular value of A, so $\sigma_{\min}(A) = \sigma_r(A)$.

Working with minimum singular values requires more care than maximum singular vectors. In particular, when we have rectangular matrices some bounds depend on whether the matrix is 'fat' (has more columns than rows) or 'tall' (has more rows than columns).

Given a matrix A, the operator norm $\|A\|_2$ is the maximum singular value: $\|A\|_2 = \sigma_{\max}(A)$.

Projectors: Given a subspace R of $\mathbb{R}^d$, let Π_R denote the orthogonal projection onto R, satisfying that for all $x \in \mathbb{R}^d$:

$$\Pi_R(x) \in R \text{ and } \forall r \in R,\ \|x - \Pi_R(x)\|_2 \leq \|x - r\|_2. \tag{A.1}$$

If $E \in \mathbb{R}^{d \times \dim(R)}$ has orthonormal columns that form a basis for R, then we have:

$$\Pi_R = EE^\top \tag{A.2}$$

From this we can easily check that $\Pi_R^2 = \Pi_R$ and $\Pi_R^\top = \Pi_R$. See e.g., Chapter 2.5.1 Golub & Loan (2013) for more information.

Principal Angles: Given two non-zero vectors x and y, the cosine of the angle between them, $\cos\theta$, is:

$$\cos\theta = \frac{x^\top y}{\|x\|_2 \|y\|_2} \tag{A.3}$$

If we consider the 1-dimensional subspaces (so basically lines) S_x and S_y spanned by x and y respectively, then the angle between them, $\cos\theta'$ is given by the absolute value (since lines are undirected):

$$\cos\theta' = \frac{|x^\top y|}{\|x\|_2 \|y\|_2} \tag{A.4}$$

Principal angles generalize this notion to higher dimensions. See e.g., Chapter 6.4.3 in Golub & Loan (2013) for more information on principal angles.

Definition A.1. *Given two non-empty subspaces R and S of $\mathbb{R}^d$, where $r = \min(\dim(R), \dim(S))$, we have r principal angles:*

$$0 \leq \theta_1 \leq \ldots \leq \theta_r \leq \pi/2. \tag{A.5}$$

The directions of the inequalities swap when we take the cosine of the principal angles:

$$1 \geq \cos\theta_1 \geq \ldots \geq \cos\theta_r \leq 0. \tag{A.6}$$

The cosines of the principal angles are given by the SVD—let $E \in \mathbb{R}^{d\times\dim(R)}$ and $F \in \mathbb{R}^{d\times\dim(S)}$ have orthonormal columns which span R and S respectively. Then we have:

$$\cos\theta_i = \sigma_i(E^\top F), \tag{A.7}$$

where σ_i denotes the i-th largest singular value. In this paper, we are interested in the cosine of the largest angle between them, given by:

$$\cos\theta_{\mathsf{max}}(R, S) = \cos\theta_r \tag{A.8}$$

We can massage this into a variational characterization of the maximum principal angle, which is important for lower bounding the error of fine-tuning outside the span of the training data.

Lemma A.2. *Suppose $\dim(R) \le \dim(S)$, and let $F \in \mathbb{R}^{d\times\dim(S)}$ have orthonormal columns that form a basis for S. We have:*

$$\cos\theta_{\mathsf{max}}(R, S) = \min_{r\in R, \|r\|_2=1} \|F^\top(r)\|_2 \tag{A.9}$$

Proof. Let $E \in \mathbb{R}^{d\times\dim(R)}$ and $F \in \mathbb{R}^{d\times\dim(S)}$ have orthonormal columns that span R and S respectively. Since $\dim(R) \le \dim(S)$ (a crucial condition!), $F^\top E$ is a 'tall' matrix (it has more rows than columns) so we have:

$$\sigma_{\min}(F^\top E) = \min_{\|v\|_2=1} \|F^\top E v\|_2. \tag{A.10}$$

The result now follows from some algebra:

$$\begin{aligned}
\cos\theta_{\mathsf{max}}(R, S) &= \sigma_{\min}(F^\top E) && \text{(A.11)}\\
&= \min_{\|v\|_2=1} \|F^\top E v\|_2 && \text{(A.12)}\\
&= \min_{r\in R, \|r\|_2=1} \|F^\top(r)\|_2. && \text{(A.13)}
\end{aligned}$$

□

A.2 Feature distortion theorem

We first prove our core theorem, that fine-tuning distorts pretrained features.

Restatement of Theorem 3.3. *In the overparameterized linear setting, let $S^\perp$ = rowspace$(X)^\perp$, R_0 = rowspace(B_0), and $v_\star, B_\star$ be the optimal parameters with $w_\star = B_\star v_\star$. If $\cos\theta_{\mathsf{max}}(R_0, S^\perp) > 0$, then for all time steps t, the OOD error of the fine-tuning iterates $(B_{\mathsf{ft}}(t), v_{\mathsf{ft}}(t))$ is lower bounded:*

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \ge \sqrt{\sigma_{\min}(\Sigma)}\Big(\frac{\cos\theta_{\mathsf{max}}(R_0, S^\perp)}{\sqrt{k}} \frac{\min(\varphi, \varphi^2/\|w_\star\|_2)}{(1+\|w_\star\|_2)^2} - \epsilon\Big), \tag{A.14}$$

where $\varphi^2 = |(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2|$ is defined to be inital head alignment error and $\epsilon \ge d(B_0, B_\star)$ is the error in the pretrained feature extractor.

We follow the sketch in the main paper. We begin with a few lemmas, showing that certain quantities are preserved throughout the fine-tuning process.

Our first lemma says that the representations $B_{ft}^t x$ do not change for examples perpendicular the span of the training examples. Note that the final output ${v_{ft}^t}^\top B_{ft}^t x$ still changes, because v_{ft}^t changes.

Lemma A.3. *For all times t and all $x \in S^\perp$, we have:*

$$B_0 x = B_{ft}^t x \tag{A.15}$$

Proof. We initialized fine-tuning with the feature extractor $B_{\mathsf{ft}}(0) = B_0$. It suffices to show that $\partial_t B_{ft}^t x = 0$ for all $x \in S^\perp$. Recall that $\partial_t B_{ft}^t$ is given by the gradient flow update equation:

$$\partial_t B_{ft}^t = -\partial_B \widehat{L}(v_{ft}^t, B_{ft}^t) = -\partial_B \|XB^\top v - Y\|_2^2 \tag{A.16}$$

Computing the RHS explicitly using multivariable chain rule, we get:

$$\partial_t B_{ft}^t = -2v(XB^\top v - Y)^\top X \tag{A.17}$$

Since x is a constant, we get:

$$\partial_t B_{ft}^t x = -2v(XB^\top v - Y)^\top Xx \tag{A.18}$$

But $Xx = 0$ for $x \in S^\perp$, since $x \in S^\perp$ is defined as x is perpendicular to the rowspace of X (i.e., perpendicular to the rows of X). So the RHS is 0—that is, $\partial_t B_{ft}^t x = 0$, as desired. □

Next, we show that the change in the head and feature extractor are 'coupled'. So if the head changes in a certain way, then the feature extractor cannot just stay the same. In the literature, this is sometimes called the "balancedness" lemma, and has been proved in prior work on two layer linear networks.

Lemma A.4. *For all t we have:*

$$v_0 v_0^\top - B_0 B_0^\top = v_{ft}^t {v_{ft}^t}^\top - B_{ft}^t {B_{ft}^t}^\top \tag{A.19}$$

Proof. This follows by showing that the derivative is 0:

$$\partial_t [v_{ft}^t {v_{ft}^t}^\top - B_{ft}^t {B_{ft}^t}^\top] = 0 \tag{A.20}$$

Which can be verified by direct calculation. See Theorem 2.2 in Du et al. (2018) and the proof of Theorem 1 in Arora et al. (2018). □

For our proof we will require that every feature $r \in R$ can be generated from some OOD direction, that is $r = B_0 u$ for some $u \in S^\perp$. We will show that this is implied by the condition on the principal angle: $\cos\theta_{\mathsf{max}}(R, S^\perp) > 0$ where $R = \text{rowspace}(B_0)$, which we assumed in Theorem 3.3. The following lemma shows this (and also quantifies that the norm of u does not shrink too much when projected onto R).

Lemma A.5. *Let R, S be subspaces of $\mathbb{R}^d$ with $\dim(R) \leq \dim(S)$. For all $r \in R$ with $\|r\|_2 = \cos\theta_{\mathsf{max}}(R, S)$, there exists $s \in S$ with $\Pi_R(s) = r$ and $\|s\|_2 \leq 1$. Here $\Pi_R \in \mathbb{R}^{d\times d}$ projects a vector onto R.*

Proof. Let $c = \cos\theta_{\mathsf{max}}(R, S)$. Firt, we get rid of an easy case—if $c = 0$, then we need to show the claim for all $r \in R$ with $\|r\|_2 = c = 0$, which means $r = 0$. Then we can just pick $s = 0$, and $\Pi_R(s) = 0 = r$ and $\|s\|_2 = 0 \leq 1$. So for the rest of the proof we assume $c > 0$.

Consider arbitrary vector $r \in R$ with $\|r\|_2 = c$. Let $E \in \mathbb{R}^{d \times \dim(S)}, F \in \mathbb{R}^{d \times \dim(R)}$ have orthonormal columns, which form a basis for R and S respectively.

Step 1: Finding s: Since the columns of E span R, $r = Ez$ for some $z \in \mathbb{R}^{\dim(R)}$. $c = \sigma_{\min}(E^\top F) > 0$, which means that $E^\top F \in \mathbb{R}^{\dim(R) \times \dim(S)}$ has rank $\dim(R)$ since $\dim(R) \leq \dim(S)$—in other words, $E^\top F$ has full column rank since the column dimension is smaller than the row dimension. So $z = E^\top F w$ for some $w \in$ rowspace$(E^\top F)$. Then we set $s = Fw$—this means $s \in S$ because the columns of F form a basis for S. In addition, following the steps above we have $r = Ez = EE^\top Fw = EE^\top s$. We note that $\Pi_R = EE^\top$ is the projection onto R (see e.g., Chapter 2.5.1 of Golub & Loan (2013)).

Step 2: Bounding norm of s: It suffices to show that $\|s\|_2 \leq 1$. Since F has orthonormal columns, $\|s\|_2 = \|Fw\|_2 = \|w\|_2$, so it suffices to show that $\|w\|_2 \leq 1$. Since E has orthonormal columns, $\|r\|_2 = \|z\|_2$. Recall that $z = E^\top Fw$—since $w \in$ rowspace$(E^\top F)$, from Lemma A.6 we have:

$$\|z\|_2 \geq \sigma_{\min}(E^\top F)\|w\|_2 = c\|w\|_2. \tag{A.21}$$

Rearranging, we get $\|w\|_2 \leq \|z\|_2 / c = 1$, as desired.

□

In the lemma above, we used a standard linear algebraic result that we include for completeness. This says that A cannot shrink vectors in its rowspace too much, where the shrinkage factor is given by the minimum singular value of A.

Lemma A.6. *Let $A \in \mathbb{R}^{m \times n}$. Let $r = \min(m, n)$. Then if $x \in$ rowspace(A), we have $\|Ax\|_2 \geq \sigma_r(A)\|x\|_2$.*

Proof. We bound the norm of x using the SVD. Consider the singular value decomposition (SVD) of A:

$$A = UDV^\top \tag{A.22}$$

Where $U \in \mathbb{R}^{m \times r}, D \in \mathbb{R}^{r \times r}, V^\top \in \mathbb{R}^{r \times n}$, where U and V have orthonormal columns, and $D = \text{diag}(\sigma_1, \ldots, \sigma_r)$ is a diagonal matrix with $\sigma_1 \geq \ldots \geq \sigma_r \geq 0$.

$$\begin{aligned}
\|Ax\|_2 &= \|UDV^\top x\|_2 && \text{[Definition of } r\text{]} && \text{(A.23)}\\
&= \|DV^\top x\|_2 && [U \in \mathbb{R}^{m \times r} \text{ has orthonormal columns}] && \text{(A.24)}\\
&\geq \sigma_r \|V^\top x\|_2 && [D \text{ is diagonal}] && \text{(A.25)}\\
&= \sigma_r \|x\|_2 && \text{[rows of } V^\top \text{ are orthonormal, } x \text{ is in rowspace]} && \text{(A.26)}\\
&= \sigma_r(A)\|x\|_2 && && \text{(A.27)}
\end{aligned}$$

Where for the fourth step, we used the fact that if $x \in$ rowspace$(V^\top)$ and the rows of $V^\top$ are orthonormal, then $\|V^\top x\|_2 = \|x\|_2$. One way to see this is by writing $x = \sum_i \alpha_i v_i$, where v_i are rows of $V^\top$, and then noting that $V^\top x = (\alpha_1, \ldots, \alpha_r)$ and so x and $V^\top x$ have the same norm. □

We recall that P_{ood} has second moment Σ: $\mathbb{E}[xx^\top] = \Sigma$ when $x \sim P_{\mathsf{ood}}$, where Σ is invertible. So with some simple algebra we can write the OOD error L_{ood} in terms of Σ (the proof is standard and basic, but we include it just for completeness):

Lemma A.7.

$$L_{\mathsf{ood}}(v, B) = (B_\star^\top v_\star - B^\top v)^\top \Sigma (B_\star^\top v_\star - B^\top v) \leq \sigma_{\min}(\Sigma) \|B_\star^\top v_\star - B^\top v\|_2^2. \tag{A.28}$$

Proof. Let $x \sim P_{\mathsf{ood}}$. We have,

$$\begin{aligned} L_{\mathsf{ood}}(v, B) &= \mathbb{E}[(v_\star^\top B_\star x - v^\top B x)^2] && \text{(A.29)} \\ &= \mathbb{E}[(B_\star^\top v_\star - B^\top v)^\top x x^\top (B_\star^\top v_\star - B^\top v)] && \text{(A.30)} \\ &= (B_\star^\top v_\star - B^\top v)^\top \mathbb{E}[x x^\top](B_\star^\top v_\star - B^\top v) && \text{(A.31)} \\ &= (B_\star^\top v_\star - B^\top v)^\top \Sigma (B_\star^\top v_\star - B^\top v). && \text{(A.32)} \end{aligned}$$

The inequality follows immediately because $\sigma_{\min}(A)$ (for a square matrix A) is simply the min over x with unit ℓ_2 norm of $x^\top A x$. □

We now prove Theorem 3.3, following the 3 steps outlined in the main text.

Proof of Theorem 3.3. Let $c = \cos \theta_{\mathsf{max}}(R, S^\perp)$. From Lemma A.7, we have $L_{\mathsf{ood}}(v_{ft}^t, B_{ft}^t) \leq \sigma_{\min}(\Sigma) \|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2^2$ so it suffices to bound $\|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2$.

Because it makes the proof much easier, we will prove the contrapositive, and then convert back to the original theorem statement. We assume $\|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2 \leq \Delta$, and will show that:

$$|(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2| \leq \frac{\Delta + \epsilon}{c} g_1(\|w\|_2)\sqrt{k} + \frac{(\Delta + \epsilon)^2}{c^2} g_2(\|w\|_2) k \tag{A.33}$$

Where g_1 and g_2 are non-negative polynomials we will bound in the proof.

We gave a basic outline of the proof in the main paper, and here we are just trying to be careful about capturing all the dependencies. We also give intuition for each step before diving into algebra (which we include for completeness).

Recall that in the overparameterized linear setting we assumed we have orthonormal B_0 with $\|B_0 - U B_\star\|_2 \leq \epsilon$ for some U. We note that the setup is rotationally symmetric so without loss of generality we can suppose $\|B_0 - B_\star\|_2 \leq \epsilon$. This is because we can let $B_\star' = U B_\star$ and $v_\star' = U v_\star$, and we have $w_\star = B_\star^\top v_\star = (U B_\star)^\top (U v_\star)$, where $w_\star$ is the optimal classifier—so we can now write the entire proof in terms of $B_\star'$ and $v_\star'$.

Step 1: Show that $\|v_{ft}^t - v_\star\|_2 \leq \Delta / c$: We first give intuition and then dive into the math. The key insight is to use the fact that in 'many' directions B_{ft}^t and B_0 are the same (formally, for all $x \in S^\perp$, $B_{ft}^t x = B_0 x$). But B_0 and $B_\star$ are close by assumption, which means that B_{ft}^t and $B_\star$ are close in 'many' directions. Then since we assumed in the contrapositive that ${v_{ft}^t}^\top B_{ft}^t$ and $v_\star^\top B_\star$ are close, we get that v_{ft}^t and $v_\star$ are close in 'many' directions. Because $S^\perp$ covers the rowspace of B_0, we get that 'many' is k, which is precisely the dimensionality of $v_\star$, so the two vectors v_{ft}^t and $v_\star$ must be close.

We now dive into the math. Since B_0 has orthogonal rows, B_0 has full column rank.

Let z be given by:

$$z = \frac{c}{\|v_{ft}^t - v_\star\|_2}(v_{ft}^t - v_\star) \tag{A.34}$$

We note that $\|z\|_2 = c$. Then, we can find $y \in R = \text{rowspace}(B_0)$ such that $B_0 y = z$ (since B_0 has full column-rank) and then $\|y\|_2 = \|z\|_2 = c$ (since B_0 has orthonormal rows).

Since $c = \cos\theta_{\mathsf{max}}(R, S^\perp) > 0$, and $y \in R$ with $\|y\| = c$, from Lemma A.5 we can choose $x \in S^\perp$ with $\|x\|_2 \leq 1$ and $\Pi_R(x) = y$. Then, we have $B_0 x = z$.

From Proposition A.3, since $x \in S^\perp$, B_0 does not change in directions of x when fine-tuning so we have: $B_0 x = B_{ft}^t x$.

The claim now follows from simple algebraic manipulation, following the intuition we described. The algebra just captures what 'close' means and adds up the error terms.

$$\begin{aligned}
\|v_{ft}^t - v_\star\|_2 &= \frac{1}{c}(v_{ft}^t - v_\star)^\top \left(\frac{c(v_{ft}^t - v_\star)}{\|v_{ft}^t - v_\star\|_2}\right) && \text{[Algebra]} && \text{(A.35)}\\
&= \frac{1}{c}(v_{ft}^t - v_\star)^\top z && \text{[Definition of } z\text{]} && \text{(A.36)}\\
&= \frac{1}{c}(v_{ft}^t - v_\star)^\top B_0 x && \text{[Since } B_0 x = z\text{]} && \text{(A.37)}\\
&= \frac{1}{c}({v_{ft}^t}^\top B_0 x - v_\star^\top B_0 x) && \text{[Algebra]} && \text{(A.38)}\\
&= \frac{1}{c}({v_{ft}^t}^\top B_{ft}^t x - v_\star^\top B_0 x) && [B_{ft}^t x = B_0 x \text{ since } x \in S^\perp] && \text{(A.39)}\\
&= \frac{1}{c}({v_{ft}^t}^\top B_{ft}^t - v_\star^\top B_0) x && \text{[Algebra]} && \text{(A.40)}\\
&\leq \frac{1}{c}\|{v_{ft}^t}^\top B_{ft}^t - v_\star^\top B_0\|_2 \|x\|_2 && \text{[Cauchy-Schwarz]} && \text{(A.41)}\\
&\leq \frac{1}{c}\|{v_{ft}^t}^\top B_{ft}^t - v_\star^\top B_0\|_2 && \text{[since } \|x\|_2 \leq 1\text{]} && \text{(A.42)}\\
&\leq \frac{1}{c}\|{v_{ft}^t}^\top B_{ft}^t - v_\star^\top B_\star\|_2 + \frac{1}{c}\|v_\star^\top B_\star - v_\star^\top B_0\|_2 && \text{[Triangle inequality]} && \text{(A.43)}\\
&\leq \frac{1}{c}\|{B_{ft}^t}^\top v_{ft}^t - B_\star^\top v_\star\|_2 + \frac{1}{c}\|v_\star^\top B_\star - v_\star^\top B_0\|_2 && \text{[Taking transpose]} && \text{(A.44)}\\
&= \frac{1}{c}\|{B_{ft}^t}^\top v_{ft}^t - B_\star^\top v_\star\|_2 + \frac{1}{c}\sigma_{\max}(B_0 - B_\star)\|v_\star\|_2 && \text{[definition of max singular value]} && \text{(A.45)}\\
&= \frac{1}{c}\|{B_{ft}^t}^\top v_{ft}^t - B_\star^\top v_\star\|_2 + \frac{1}{c}\epsilon\|v_\star\|_2 && \text{[since } \sigma_{\max}(B_0 - B_\star) \leq \epsilon\text{]} && \text{(A.46)}\\
&\leq \frac{\Delta + \epsilon\|v_\star\|_2}{c} && \text{[since } \|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2 \leq \Delta\text{]} && \text{(A.47)}\\
& && && \text{(A.48)}
\end{aligned}$$

Which shows that $\|v_{ft}^t - v_\star\|_2 \leq (\Delta + \epsilon\|v_\star\|_2)/c$.

Step 2A: Show that $\|B_{ft}^t\|_F^2$ is small: The key insight is to take the trace on both sides of Proposition A.4,

which bounds the Frobenius norm of B_{ft}^t and therefore the operator norm.

Rearranging Proposition A.4, we have:

$$B_{ft}^t {B_{ft}^t}^\top = B_0 B_0^\top + v_\star v_\star^\top - v_0 v_0^\top \tag{A.49}$$

Taking the trace everywhere, we get:

$$\mathrm{Tr}(B_{ft}^t {B_{ft}^t}^\top) = \mathrm{Tr}(B_0 B_0^\top) + \mathrm{Tr}(v_\star v_\star^\top) - \mathrm{Tr}(v_0 v_0^\top) \tag{A.50}$$

For any matrix A, $\mathrm{Tr}(AA^\top) = \|A\|_F^2$, and for a vector v the Frobenius norm is just the ℓ_2-norm, so $\mathrm{Tr}(vv^\top) = \|v\|_2^2$. So we have:

$$\|B_{ft}^t\|_F^2 = \|B_0\|_F^2 + \|v_\star\|_2^2 - \|v_0\|_2^2 \tag{A.51}$$

Squares are non-negative, so we get the inequality:

$$\|B_{ft}^t\|_F^2 \leq \|B_0\|_F^2 + \|v_\star\|_2^2 \tag{A.52}$$

Step 2B: Show that $\|B_0^\top v_\star\|_2^2 - \|{B_{ft}^t}^\top v_\star\|_2^2$ is small: This step doesn't involve much insight, and is standard peturbation analysis—we simply factor the difference of squares and bound each term.

First, we bound $\|{B_{ft}^t}^\top v_{ft}^t - {B_{ft}^t}^\top v_\star\|_2$:

$$\begin{aligned}
\|{B_{ft}^t}^\top v_{ft}^t - {B_{ft}^t}^\top v_\star\|_2 &\leq \sigma_{\max}(B_{ft}^t)\|v_{ft}^t - v_\star\|_2 && \text{(A.53)}\\
&\leq \|B_{ft}^t\|_F \|v_{ft}^t - v_\star\|_2 && \text{(A.54)}\\
&\leq \sqrt{\|B_0\|_F^2 + \|v_\star\|_2^2}\,\|v_{ft}^t - v_\star\|_2 && \text{(A.55)}\\
&\leq \sqrt{\|B_0\|_F^2 + \|v_\star\|_2^2}\Big(\frac{\Delta + \epsilon\|v_\star\|_2}{c}\Big) && \text{(A.56)}
\end{aligned}$$

Next, we bound $\|B_0^\top v_\star - {B_{ft}^t}^\top v_\star\|_2$:

$$\begin{aligned}
\|B_0^\top v_\star - {B_{ft}^t}^\top v_\star\|_2 &\leq \|B_0^\top v_\star - B_\star^\top v_\star\|_2 + \|B_\star^\top v_\star - {B_{ft}^t}^\top v_\star\|_2 && \text{(A.57)}\\
&\leq \sigma_{\max}(B_0 - B_\star)\|v_\star\|_2 + \|B_\star^\top v_\star - {B_{ft}^t}^\top v_\star\|_2 && \text{(A.58)}\\
&\leq \epsilon\|v_\star\|_2 + \|B_\star^\top v_\star - {B_{ft}^t}^\top v_\star\|_2 && \text{(A.59)}\\
&\leq \epsilon\|v_\star\|_2 + \|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2 + \|{B_{ft}^t}^\top v_{ft}^t - {B_{ft}^t}^\top v_\star\|_2 && \text{(A.60)}\\
&\leq \epsilon\|v_\star\|_2 + \Delta + \sqrt{\|B_0\|_F^2 + \|v_\star\|_2^2}\Big(\frac{\Delta + \epsilon\|v_\star\|_2}{c}\Big) && \text{(A.61)}\\
&=: \Delta_2 && \text{(A.62)}
\end{aligned}$$

Finally, we bound $|\|B_0^\top v_\star\|_2^2 - \|{B_{ft}^t}^\top v_\star\|_2^2|$, using the identity:

$$\begin{aligned}
|\|u\|_2^2 - \|v\|_2^2| &= |(u-v)^\top (u+v)| && \text{(A.63)}\\
&\leq \|u-v\|_2 \|u+v\|_2 && \text{(A.64)}\\
&\leq \|u-v\|_2 (2\|u\|_2 + \|u-v\|_2) && \text{(A.65)}
\end{aligned}$$

Applying this:

$$|\|B_0^\top v_\star\|_2^2 - \|{B_{ft}^t}^\top v_\star\|_2^2| \le \|B_0^\top v_\star - {B_{ft}^t}^\top v_\star\|_2 (2\|B_0^\top v_\star\|_2 + \|B_0^\top v_\star - {B_{ft}^t}^\top v_\star\|_2) \tag{A.66}$$

$$\le \Delta_2 (2\|B_0^\top v_\star\|_2 + \Delta_2) \tag{A.67}$$

$$\le \Delta_2 (2\|B_\star^\top v_\star\|_2 + 2\|B_0^\top v_\star - B_\star^\top v_\star\|_2 + \Delta_2) \tag{A.68}$$

$$\le \Delta_2 (2\|w_\star\|_2 + 2\epsilon\|v_\star\|_2 + \Delta_2) \tag{A.69}$$

$$=: \Delta_3 \tag{A.70}$$

Step 3: Use Proposition A.4 to show v_0 and $v_\star$ must be close: The key insight is that we start from Proposition A.4, and left and right multiply by $v_\star$, after that we use the previous steps and do some some standard perturbation analysis.

We start from Proposition A.4:

$$v_0 v_0^\top - B_0 B_0^\top = v_{ft}^t {v_{ft}^t}^\top - B_{ft}^t {B_{ft}^t}^\top \tag{A.71}$$

The key step is to left multiply both sides by $v_\star^\top$ and right multiply both sides by $v_\star$ to get:

$$(v_0^\top v_\star)^2 - \|B_0^\top v_\star\|_2^2 = ({v_{ft}^t}^\top v_\star)^2 - \|{B_{ft}^t}^\top v_\star\|_2^2 \tag{A.72}$$

Rearranging, and then using Equation A.66, we get:

$$|({v_{ft}^t}^\top v_\star)^2 - (v_0^\top v_\star)^2| = |\|{B_{ft}^t}^\top v_\star\|_2^2 - \|B_0^\top v_\star\|_2^2| \le \Delta_3 \tag{A.73}$$

This is close to what we want, except we have $({v_{ft}^t}^\top v_\star)^2$ on the LHS instead of $(v_\star^\top v_\star)^2$. We previously showed that v_{ft}^t and $v_\star$ are close, in Step 1, so with some algebra we can bound the difference between $({v_{ft}^t}^\top v_\star)^2$ and $(v_\star^\top v_\star)^2$:

$$|({v_{ft}^t}^\top v_\star)^2 - (v_\star^\top v_\star)^2| = |({v_{ft}^t}^\top v_\star - v_\star^\top v_\star)^\top ({v_{ft}^t}^\top v_\star + v_\star^\top v_\star)| \tag{A.74}$$

$$= |({v_{ft}^t}^\top v_\star - v_\star^\top v_\star)^\top [2v_\star^\top v_\star + ({v_{ft}^t}^\top v_\star - v_\star^\top v_\star)]| \tag{A.75}$$

$$= |(v_\star^\top (v_{ft}^t - v_\star))^\top [2v_\star^\top v_\star + (v_\star^\top (v_{ft}^t - v_\star))]| \tag{A.76}$$

$$\le \|v_{ft}^t - v_\star\|_2 \|v_\star\|_2^2 [2\|v_\star\|_2 + \|v_{ft}^t - v_\star\|_2] \tag{A.77}$$

$$= (\Delta/c)\|v_\star\|_2^2 (2\|v_\star\|_2 + (\Delta/c)) := \Delta_4 \tag{A.78}$$

Above, from the third line to the fourth line, we used triangle inequality and Cauchy-Schwarz.

So finally, by triangle-inequality we can now bound $|(v_\star^\top v_\star)^2 - (v_0^\top v_\star)^2|$:

$$|(v_\star^\top v_\star)^2 - (v_0^\top v_\star)^2| \le |(v_\star^\top v_\star)^2 - ({v_{ft}^t}^\top v_\star)^2| + |({v_{ft}^t}^\top v_\star)^2 - (v_0^\top v_\star)^2| \tag{A.79}$$

$$\le \Delta_4 + \Delta_3 \tag{A.80}$$

Wrap up i.e., writing out $\Delta_4 + \Delta_3$ explicitly: This is basically the bound we want, but we would like to express Δ_3, Δ_4 in terms of Δ and ϵ. Note that this step has no insight, and is just algebra—we include the details for reference and verifiability. We recall:

$$\Delta_4 = (\Delta/c)\|v_\star\|_2^2 (2\|v_\star\|_2 + (\Delta/c)) \tag{A.81}$$

$$\Delta_3 = \Delta_2 (2\|w_\star\|_2 + 2\epsilon\|v_\star\|_2 + \Delta_2) \tag{A.82}$$

$$\Delta_2 = \epsilon\|v_\star\|_2 + \Delta + \sqrt{\|B_0\|_F^2 + \|v_\star\|_2^2} \Big(\frac{\Delta + \epsilon\|v_\star\|_2}{c} \Big) \tag{A.83}$$

Since B_0 has orthogonal rows (by assumption), $B_0^\top$ has orthogonal columns, so $\|w_\star\|_2 = \|B_0^\top v_\star\|_2 = \|v_\star\|_2$. In addition, since B_0 has k orthogonal rows, $\|B_0\|_F = \sqrt{k}$. We also note that $\sqrt{\|B_0\|_F^2 + \|v_\star\|_2^2} \leq \|B_0\|_F + \|v_\star\|_2 = \sqrt{k} + \|w_\star\|_2$. Since $c \leq 1$, we have:

$$\epsilon\|v_\star\|_2 + \Delta \leq \Big(\frac{\Delta + \epsilon\|v_\star\|_2}{c}\Big) \tag{A.84}$$

So for Δ_2, up to constant factors we can ignore the $\epsilon\|v_\star\|_2 + \Delta$ term—this means we get:

$$\Delta_2 \leq O\Big((\sqrt{k} + \|w_\star\|_2)\Big(\frac{\Delta + \epsilon\|w_\star\|_2}{c}\Big)\Big) \tag{A.85}$$

Using the fact that $\sqrt{k} + \|w_\star\|_2 \leq \sqrt{k}(1 + \|w_\star\|)$ we get:

$$\Delta_2 \leq O\Big(\sqrt{k}(1 + \|w_\star\|)\Big(\frac{\Delta + \epsilon\|w_\star\|_2}{c}\Big)\Big) \tag{A.86}$$

Then since $\Delta + \epsilon\|w_\star\|_2 \leq (1 + \|w_\star\|_2)(\Delta + \epsilon)$, we get:

$$\Delta_2 \leq O\Big(\sqrt{k}(1 + \|w_\star\|)^2\Big(\frac{\Delta + \epsilon}{c}\Big)\Big) \tag{A.87}$$

Now for Δ_3, first note that $\epsilon \leq 2$, since $B_\star$ and B_0 have orthogonormal rows so $\|B_\star - B_0\|_2 \leq 2$. This means that $\epsilon\|w_\star\|_2 \leq \|w_\star\|_2$, so Δ_3 simplifies to:

$$\Delta_3 \leq O(\Delta_2(\|w_\star\|_2 + \Delta_2)) = O(\Delta_2\|w_\star\|_2 + \Delta_2) \tag{A.88}$$

Substituting the bound for Δ_2 into Δ_3, we get:

$$\Delta_3 \leq O\Big(\sqrt{k}\|w_\star\|_2(1 + \|w_\star\|)^2\Big(\frac{\Delta + \epsilon\|w_\star\|_2}{c}\Big) + k(1 + \|w_\star\|)^4\Big(\frac{\Delta + \epsilon\|w_\star\|_2}{c}\Big)^2\Big) \tag{A.89}$$

For Δ_4, we get:

$$\Delta_4 \leq O\Big(\|w_\star\|_2^3\frac{\Delta}{c} + \|w_\star\|_2(\frac{\Delta}{c})\Big) \tag{A.90}$$

Since $\Delta/c \leq (\Delta + \epsilon)/c$ and $\|w_\star\|_2^2 \leq (1 + \|w_\star\|_2)^2$ we have for the final error $\Delta_3 + \Delta_4$:

$$\Delta_3 + \Delta_4 \leq \sqrt{k}w(1 + \|w_\star\|_2^2)^2\Big(\frac{\Delta + \epsilon}{c}\Big) + k(1 + \|w_\star\|_2^2)^4\Big(\frac{\Delta + \epsilon}{c}\Big)^2 \tag{A.91}$$

Wrap up i.e., taking the contrapositive: So we've shown that if $\|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2^2 \leq \Delta$, then:

$$|(v_\star^\top v_\star)^2 - (v_0^\top v_\star)^2| \leq \frac{\Delta + \epsilon}{c}w(1 + \|w_\star\|_2^2)^2\sqrt{k} + \frac{(\Delta + \epsilon)^2}{c^2}(1 + \|w_\star\|_2^2)^4 k \tag{A.92}$$

We'd like to flip this around: suppose $|(v_\star^\top v_\star)^2 - (v_0^\top v_\star)^2| \geq \varphi^2$ for some φ. To lower bound $\|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2^2$, we simply take the contrapositive of what we have proved. Let Δ be given by:

$$\Delta = \min\Big(\frac{c}{w(1 + \|w_\star\|_2^2)^2\sqrt{k}}\varphi^2, \frac{c}{\sqrt{(1 + \|w_\star\|_2^2)^4 k}}\varphi\Big) - \epsilon \tag{A.93}$$

In this case with some algebra, we can show that:

$$|(v_\star^\top v_\star)^2 - (v_0^\top v_\star)^2| \geq \varphi^2 \geq \frac{\Delta + \epsilon}{c}w(1 + \|w_\star\|_2^2)^2\sqrt{k} + \frac{(\Delta + \epsilon)^2}{c^2}(1 + \|w_\star\|_2^2)^4 k \tag{A.94}$$

To see this, we bound each of the terms in the RHS separately using our definition of Δ. Then, from the contrapositive of what we proved (compare with Equation A.92, we get:

$$\|B_\star^\top v_\star - {B_{ft}^t}^\top v_{ft}^t\|_2^2 \geq \Delta \tag{A.95}$$

Finally, we can massage Δ to combine terms and make it look slightly nicer:

$$\Delta \geq \frac{c}{\sqrt{k}} \frac{\min(\varphi, \varphi^2/\|w_\star\|_2)}{(1+\|w_\star\|_2)^2} - \epsilon \tag{A.96}$$

Then applying Lemma A.7 we get the desired result. For even more interpretability, if $\|w\|_2 = 1$ and φ is bounded above by some constant, then you can think of Δ as approximately $\frac{c}{\sqrt{k}}\varphi^2 - \epsilon$. This completes the proof. □

A.3 LP vs. FT (OOD)

We now prove Theorem 3.5, which compares linear probing and fine-tuning in the linear overparameterized setting, when the ID data lies in a lower dimensional subspace.

We first state a more precise version of Theorem 3.5—basically we fix all problem parameters except B_0 (which limits to $B_\star$). To define the limit, we consider a sequence of pretrained feature extractors: $\{B_0^i\}_{i=1}^\infty$. We define the corresponding limit points of fine-tuning and linear probing when we start from the i-th pretrained feature extractor. That is, let ${v_{\mathsf{ft}}}^i(t), {B_{\mathsf{ft}}}^i(t)$ denote the parameters at time t of fine-tuning if we initialize with v_0, B_0^i (see Equation 3.2 for the fine-tuning updates). Let $v_{\mathsf{lp}}^{\infty i}, B_0^i$ be the linear probing solution when initialized with v_0, B_0^i (see Equation 3.5 for the linear probing updates). We note that the LP iterates converge to $v_{\mathsf{lp}}^{\infty i}, B_0^i$ as a result of gradient flow on a convex problem.

Finally, Theorem 3.5 says that as the pretrained representations get better, linear probing does much better than fine-tuning OOD:

Theorem A.8 (Formal statement of Theorem 3.5)**.** *In the linear overparameterized setting, under the ID subspace assumption, fix the dimensions of the setting d, k, m, number of examples n, the ID subspace S, ID distribution P_{id}, the distribution over the head v_0, and the ground truth parameters $v_\star, B_\star$. Assume the non-degeneracy conditions $\cos\theta_{\mathsf{max}}(R_*, S) > 0$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp) > 0$ where $R_* = $ rowspace$(B_\star)$. Given a sequence of pretrained feature extractors $\{B_0^i\}_{i=1}^\infty$ with $B_0^i \to B_\star$, where the limit is in the pseudometric given by Definition 3.1, the ratio of OOD errors of linear probing and fine-tuning converges in probability to 0:*

$$\frac{L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i)}{\inf_{t\geq 0} L_{\mathsf{ood}}({v_{\mathsf{ft}}}^i(t), {B_{\mathsf{ft}}}^i(t))} \xrightarrow{p} 0, \text{as } \ i \to \infty. \tag{A.97}$$

The purpose of the infimum is to capture the fact that the bound holds for all times t for fine-tuning (and therefore also for the limit $v_{\mathsf{ft}}^\infty, B_{\mathsf{ft}}^\infty$ when it exists). Note that the ratio is a random variable because the training data is sampled from P_{id} and the head is sampled ($v_0 \sim \mathcal{N}(0, \sigma^2 I)$ for some σ^2).

Proof. Recall that we say a sequence of real-valued random variables converges in probability to 0 (written as $X_i \xrightarrow{p} 0$) if for every $\epsilon', \delta > 0$, for all large enough i (that is, for all $i \geq N_i$ for some N_i), we have:

$$P(|X_i| > \epsilon') \leq \delta. \tag{A.98}$$

Accordingly, fix arbitrary $\epsilon', \delta > 0$, and we will show that the ratio of errors is eventually smaller than ϵ' with probability at least $1 - \delta$.

Lower bounding fine-tuning error: Since $B_0^i \to B_\star$, from Lemma A.10 we have that $\cos\theta_{\mathsf{max}}(R^i, S^\perp) \to \cos\theta_{\mathsf{max}}(R_*, S^\perp)$ where $R^i = \text{rowspace}(B_0^i)$. Since $\cos\theta_{\mathsf{max}}(R_*, S^\perp) > 0$, this means that for all large enough i we have:

$$\cos\theta_{\mathsf{max}}(R^i, S^\perp) > \cos\theta_{\mathsf{max}}(R_*, S^\perp)/2. \tag{A.99}$$

Next, from Lemma A.12, we have that with probability at least $1 - \delta/2$, Head-Error$(v_0, v_\star) = |(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2| \geq c_\delta$ for some $c_\delta > 0$. Plugging this into the fine-tuning bound in Theorem 3.3, this means that for all large enough i with probability at least $1 - \delta/2$:

$$\inf_{t \geq 0} \sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}{}^i(t), B_{\mathsf{ft}}{}^i(t))} \geq c'_\delta - d(B_0^i, B_\star), \tag{A.100}$$

for some $c'_\delta > 0$. But since $B_0^i \to B_\star$ we have $d(B_0^i, B_\star) \to 0$ as $i \to \infty$. So this means that for all large enough i with probability at least $1 - \delta/2$:

$$\inf_{t \geq 0} L_{\mathsf{ood}}(v_{\mathsf{ft}}{}^i(t), B_{\mathsf{ft}}{}^i(t)) \geq c''_\delta, \tag{A.101}$$

for some $c''_\delta > 0$.

Upper bounding the linear probing error: Since $B_0^i \to B_\star$, from Lemma A.10 we have that $\cos\theta_{\mathsf{max}}(R^i, S) \to \cos\theta_{\mathsf{max}}(R_*, S)$ and so since $\cos\theta_{\mathsf{max}}(R_*, S) > 0$, for all large enough i we have:

$$\cos\theta_{\mathsf{max}}(R^i, S) > \cos\theta_{\mathsf{max}}(R_*, S)/2. \tag{A.102}$$

Plugging this into the RHS of Lemma A.14, Equation A.132, which upper bounds the OOD error of linear probing, we get that for all large enough i, with probability at least $1 - \delta/2$:

$$L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i) \leq u_\delta (d(B_0^i, B_\star))^2, \tag{A.103}$$

for some $u_\delta > 0$. Again since $d(B_0^i, B_\star) \to 0$ as $i \to \infty$, this means for all large enough i, with probability at least $1 - \delta/2$, $d(B_0^i, B_\star)$ will be small enough so that:

$$L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i) \leq c''_\delta \epsilon. \tag{A.104}$$

Taking the ratio: So taking the ratio of the lower bound for fine-tuning, and upper bound for linear probing, we get with with probability at least $1 - \delta$:

$$\frac{L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i)}{\inf_{t \geq 0} L_{\mathsf{ood}}(v_{\mathsf{ft}}{}^i(t), B_{\mathsf{ft}}{}^i(t))} \leq \epsilon, \tag{A.105}$$

as desired.

□

We now prove the Lemmas that we used in the above proof.

A.3.1 Convergence of principal angle

Theorem 3.5 assumes conditions on the angle between the perfect feature extractor $B_\star$ and the ID subspace S. However, fine-tuning and linear probing start from features B_0 with some error, and do not get access to $B_\star$. We show that if B_0 and $B_\star$ are close, then the angles between their rowspaces to a third subspace T (which could be the the ID subspace S) is similar.

Lemma A.9. *Given two feature extractors $B_0, B_\star \in \mathbb{R}^{k\times d}$ with orthonormal rows, where $R_0 = $ rowspace(B_0), $R_* = $ rowspace$(B_\star)$, and a subspace T with dimension at least 1, we have:*

$$|\cos\theta_{\mathsf{max}}(R_0, T) - \cos\theta_{\mathsf{max}}(R_*, T)| \le d(B_0, B_\star) \tag{A.106}$$

Proof. Recall that $k = \dim(R_0) = \dim(R_*)$. Let $r = \min(k, \dim(T))$ and let F be a d-by-$\dim(T)$ matrix with orthonormal columns that form a basis for T. We have, for arbitrary rotation matrix $U \in \mathbb{R}^{k\times k}$:

$$\begin{aligned}
\cos\theta_{\mathsf{max}}(R_0, T) &= \sigma_r(B_0F) && \text{(A.107)}\\
&= \sigma_r(UB_0F) && \text{(A.108)}\\
&= \sigma_r(B_\star F + (UB_0 - B_\star)F) && \text{(A.109)}\\
&\ge \sigma_r(B_\star F) - \sigma_1((UB_0 - B_\star)F) && \text{(A.110)}\\
&\ge \sigma_r(B_\star F) - \sigma_1(UB_0 - B_\star) && \text{(A.111)}\\
&= \sigma_r(B_\star F) - \|UB_0 - B_\star\|_2 && \text{(A.112)}\\
&= \cos\theta_{\mathsf{max}}(R_*, T) - \|UB_0 - B_\star\|_2 && \text{(A.113)}
\end{aligned}$$

Here in the first step we used the definition of $\cos\theta_{\mathsf{max}}$ (Definition 3.2), and the fact that $B_0^\top$ has orthonormal columns which form a basis for R_0 (the rowspace of B_0), so in Definition 3.2 we can subtitute $E = B_0^\top$. To get Equation A.110 we used Weyl's theorem, which bounds the singular value under perturbations: $\sigma_r(A + B) \ge \sigma_r(A) - \sigma_1(B)$. To get Equation A.111 we used the fact that $\|Fv\|_2 = \|v\|$ since F has orthonormal columns.

Since this holds for all rotation matrices U, we can take the minimum over U to get:

$$\cos\theta_{\mathsf{max}}(R_0, T) \ge \cos\theta_{\mathsf{max}}(R_*, T) - \min_U \|UB_0 - B_\star\|_2 = \cos\theta_{\mathsf{max}}(R_*, T) - d(B_0^i, B_\star) \tag{A.114}$$

Since the relationship between B_0 and $B_\star$ are symmetric (and the distance d is symmetric), this gives us the desired result:

$$|\cos\theta_{\mathsf{max}}(R_0, T) - \cos\theta_{\mathsf{max}}(R_*, T)| \le d(B_0, B_\star) \tag{A.115}$$

□

Lemma A.10. *Given a sequence of pretrained feature extractors $\{B_0^i\}_{i=1}^\infty$ with $B_0^i \to B_\star$, where $B_0^i, B_\star \in \mathbb{R}^{k\times d}$ have orthonormal rows, let $R^i = $ rowspace(B_0^i), $R_* = $ rowspace$(B_\star)$. Then for any subspace T, we have:*

$$\cos\theta_{\mathsf{max}}(R^i, T) \to \cos\theta_{\mathsf{max}}(R_*, T), \text{as } \ i \to \infty. \tag{A.116}$$

Proof. This follows directly from Lemma A.9. $B_0^i \to B_\star$ means $d(B_0^i, B_\star) \to 0$. Then from Lemma A.9:

$$|\cos\theta_{\mathsf{max}}(R^i, T) - \cos\theta_{\mathsf{max}}(R_*, T)| \to 0, \text{as } \ i \to \infty \tag{A.117}$$

This means $\cos\theta_{\mathsf{max}}(R^i, T) \to \cos\theta_{\mathsf{max}}(R_*, T)$ as $i \to \infty$ □

A.3.2 Bounding the head error

We prove a lower bound on Head-Error$(v_0, v_\star) = |(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2|$, which was a key term in the fine-tuning lower bound (Theorem 3.3). Note that if the head is initialized as $v_0 = 0$, then Head-Error$(v_0, v_\star) = \|v_\star\|_2^2 = \|w_\star\|_2^2$. In practice, the head is usually initialized randomly, for example normally distributed. Intuitively, the head error is still high because we do not know which direction the head is pointing in, so most of the time the initial (randomly sampled) head will be pointing in the wrong direction. If $v_0 \sim N(0, \sigma^2 I)$ can show that for any σ^2, the head error will still typically be at least $\Omega(\|v_\star\|_2)$ This is an illustrative result, one can show similar results for other random initializations as well.

We first prove an anti-concentration lemma, which says that if u is univariate Gaussian, then it cannot be too close to any particular constant a, no matter how the variance of the Gaussian is chosen.

Lemma A.11. *For some universal constant c, given $a > 0$, for all ν^2 if $u \sim N(0, \nu^2)$ then for all $0 \le \delta \le 1$:*

$$P(|u - a| \le c\delta a) \le \delta \tag{A.118}$$

Proof. Consider δ such that $\delta \le 1/10$. Then for all u with $|u - a| \le \delta a$, we have $u \ge 9a/10$. For all $u \ge 9a/10$, the density $f(u)$ is upper bounded (from the formula for the density of a Gaussian random variable) by:

$$f(u) \le O(\frac{1}{v} \exp \frac{-9^2 a^2}{2 \cdot 10^2 v^2}) \tag{A.119}$$

We can maximize this explicitly (e.g., use Mathematica or by taking the logarithm and then setting the derivative to 0) and we get for some universal constant $c' \ge 10$ (it is OK to choose a larger universal constant than needed):

$$f(u) \le \frac{c'}{a} \tag{A.120}$$

Since the density is less than c'/a and if $|u - a| \le \delta a$ the size of the interval is $2\delta a$, we get for all $\delta \le 1/10$:

$$P(|u - a| \le \delta a) \le \frac{2c'\delta a}{a} = 2c'\delta \tag{A.121}$$

Now, we substitute $\delta' = 2c'\delta$. We get for all $\delta' \le 2c'/10$:

$$P(|u - a| \le \frac{1}{2c'}\delta' a) \le \delta' \tag{A.122}$$

Since $c' \ge 10$, $2c'/10 \ge 1$, so the statement is true for all $0 \le \delta' \le 1$. □

We now bound the error in the head if the initialization is Gaussian. This bound holds for all initialization variances σ^2. Similar bounds can be shown for other (non-Gaussian) head initializations using similar anti-concentration arguments.

Lemma A.12. *For some universal constant c, for all $v_\star \in \mathbb{R}^k$ with $v_\star \ne 0$, $\sigma \in \mathbb{R}^+$, $\delta \in [0, 1]$, if $v_0 \sim N(0, \sigma^2 I_k)$, we have with probability at least $1 - \delta$:*

$$(\textit{Head-Error}(v_0, v_\star))^2 := |(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2| \ge c\delta(v_\star^\top v_\star)^2 \tag{A.123}$$

Proof. First note that Head-Error$(v_0, v_\star)$ = Head-Error$(-v_0, v_\star)$ and v_0 is symmetric around 0 (v_0 and $-v_0$ have the same probability), and is almost surely not exactly 0. So without loss of generality, we can suppose that $v_0^\top v_\star \geq 0$.

Suffices to bound $|v_0^\top v_\star - v_\star^\top v_\star|$: We decompose the error:

$$|(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2| = |v_0^\top v_\star - v_\star^\top v_\star|(|v_0^\top v_\star + v_\star^\top v_\star|) \tag{A.124}$$

$$\geq |v_0^\top v_\star - v_\star^\top v_\star|(v_\star^\top v_\star)| \tag{A.125}$$

So we bound $|v_0^\top v_\star - v_\star^\top v_\star|$.

$v_0^\top v_\star$ **is normally distributed**: We note that $v_0^\top v_\star$ is distributed as:

$$v_0^\top v_\star \sim N(0, \sigma^2 v_\star^\top v_\star) \tag{A.126}$$

In other words, a normal with mean 0, and *variance* $\sigma_1^2 = \sigma^2 v_\star^\top v_\star$, and therefore standard deviation $\sigma_1 = \sigma\sqrt{v_\star^\top v_\star}$.

Apply Gaussian anti-concentration lemma: Then, from Lemma A.11, we have for some universal constant c that with probability at least $1 - \delta$:

$$|v_0^\top v_\star - v_\star^\top v_\star| \geq c\delta v_\star^\top v_\star \tag{A.127}$$

So substituting this back into Equation A.124, we get the desired result:

$$|(v_0^\top v_\star)^2 - (v_\star^\top v_\star)^2| \geq c\delta(v_\star^\top v_\star)^2 \tag{A.128}$$

□

A.3.3 Upper bounding linear probing error

We showed a lower bound for the OOD error of fine-tuning in Theorem 3.3. To compare this with linear probing, we prove an *upper bound* on the OOD error of linear probing.

For completeness we include an elementary lemma (note that the condition that the matrices are tall is important for composing $\sigma_{\min}$, unlike for $\sigma_{\max}$, and we included this lemma to be careful about these conditions):

Lemma A.13. *Suppose we have two matrices A, B of shape (r, s) and (s, t) respectively, and they are tall matrices so $r \geq s \geq t$. Then we have:*

$$\sigma_{\min}(AB) \geq \sigma_{\min}(A)\sigma_{\min}(B) \tag{A.129}$$

Proof. For a tall matrix A, we have:

$$\sigma_{\min}(A) = \min_{\|x\|_2 \leq 1} \|Ax\|_2 \tag{A.130}$$

So we have:

$$\sigma_{\min}(AB) = \min_{\|x\|_2 \leq 1} \|ABx\|_2 \geq \sigma_{\min}(A)\sigma_{\min}(B) \min_{\|x\|_2 \leq 1} \|x\|_2 \tag{A.131}$$

And $\min_{\|x\|_2 \leq 1} \|x\|_2 = 1$ which completes the proof. □

Lemma A.14. *In the linear overparameterized setting, under the ID subspace assumption, fix arbitrary P_z. Then there exists c_δ such that with probability at least $1-\delta$, for all $d, n, m, k, w_\star$, feature extractors $B_\star, B_0$, and ID subspaces S with corresponding F (whose columns are orthonormal and form a basis for S), if $\cos\theta_{\mathsf{max}}(S,R) > 0$, we have:*

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)} \leq \Big(\frac{c_\delta}{\cos\theta_{\mathsf{max}}(S,R)}\Big)^2 d(B_0, B_\star)||w*||_2 \tag{A.132}$$

If P_z is isotropic Gaussian so $\mathcal{N}(0, I_m)$, then we derive a bound for c_δ analytically: if $n \geq 5m$ and $n \geq 10\log\frac{1}{\delta}$ then with probability at least $1-\delta$, the linear probing OOD error is upper bounded by:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)} \leq O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R,S))^2} d(B_0, B_\star)\|w_\star\|_2\Big) \tag{A.133}$$

Proof. From the ID subspace assumption, the data matrix X of shape (n, d) can be written as $X = ZF^\top$ where Z be a matrix of shape (n, m) with each row Z_i sampled iid from P_z, and F is a matrix of shape (d, m) whose columns are orthonormal and form a basis for the ID subspace S.

Let $\epsilon = \|B_\star - B_0\|_2 \leq$. We first prove the bounds for ϵ, in terms of $d(B_0, B_\star)$ and we later handle the fact that the feature extractor distance involves the min over rotation matrices U: $d(B_0, B_\star) = \min_U \|UB_0 - B_\star\|_2$.

Bounding key singular values: Before proceeding with the proof, we examine a key quantity $XB_0^\top = ZF^\top B_0^\top$ which comes up in the Hessian of the loss function. We will show that this is invertible almost surely, and get a lower bound on its min singular value.

First, we examine the shapes of the matrices. $ZF^\top B_0^\top$ has shape (n, d) where Z has shape (n, m) and $F^\top B_0^\top$ has shape (m, k). Since $n \geq m > k$ we have that Z and $F^\top B_0^\top$ are tall matrices, and so from Lemma A.13 we can write the min singular value of $ZF^\top B_0^\top$ as:

$$\sigma_{\min}(ZF^\top B_0^\top) \geq \sigma_{\min}(Z)\sigma_{\min}(F^\top B_0^\top) \tag{A.134}$$

Now from the definion of the principal angle (Definition 3.2), we have:

$$\sigma_{\min}(F^\top B_0^\top) = \cos\theta_{\mathsf{max}}(R,S) > 0. \tag{A.135}$$

Since we assumed P_z has density in the ID subspace assumption, from Lemma 3 in Xie et al. (2021a) we get that for some $c'_\delta > 0$ that depends on δ and P_z, with probability at least $1-\delta$:

$$\sigma_{\min}(Z) \geq c'_\delta \tag{A.136}$$

Note that this also means that $\sigma_{\min}(ZF^\top B_0^\top) > 0$ and so $XB_0^\top = ZF^\top B_0^\top$ has full rank k almost surely. This also implies that $B_0X^\top XB_0^\top$ is a matrix of shape (k, k) that is invertible almost surely.

Main proof Since $B_0X^\top XB_0^\top$ is invertible almost surely, there is a unique global minimum (minimizing over v) to the loss optimized by linear-probing:

$$\arg\min_v \|XB_0^\top v - XB_\star^\top v_\star\|_2^2 = (B_0X^\top XB_0^\top)^{-1}B_0X^\top XB_\star^\top v_\star \tag{A.137}$$

We can see this by noting that the loss function on the LHS is strongly convex in v since the Hessian $B_0X^\top XB_0^\top$ is invertible. Then, gradient flow converges to the unique minimizer on the RHS, so:

$$v_{\mathsf{lp}}^\infty = (B_0X^\top XB_0^\top)^{-1}B_0X^\top XB_\star^\top v_\star \tag{A.138}$$

We now bound the square-root OOD error (taking the square root makes it easier to apply triangle inequalities), starting with the definition:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty}, B_0)} = \|B_\star^\top v_\star - B_0^\top v_{\mathsf{lp}}^{\infty}\|_2 \tag{A.139}$$

$$\leq \|(B_\star^\top v_\star - B_0^\top v_\star) + (B_0^\top v_\star - B_0^\top v_{\mathsf{lp}}^{\infty})\|_2 \tag{A.140}$$

$$\leq \underbrace{\|B_\star^\top v_\star - B_0^\top v_\star\|_2}_{(1)} + \underbrace{\|B_0^\top v_\star - B_0^\top v_{\mathsf{lp}}^{\infty})\|_2}_{(2)} \tag{A.141}$$

We bound each term on the RHS of the last line. For term (1):

$$\|B_\star^\top v_\star - B_0^\top v_\star\|_2 \leq \sigma_{\max}(B_\star - B_0)\|v_\star\|_2 \tag{A.142}$$

$$\leq \epsilon\|v_\star\|_2 \tag{A.143}$$

$$= \epsilon\|w_\star\|_2. \tag{A.144}$$

Where we note that $\|v_\star\|_2 = \|w_\star\|_2$ because $w_\star = B_\star^\top v_\star$ where the rows of $B_\star$ (columns of $B_\star^\top$) are orthonormal.

Let $\Sigma = X^\top X$. For term (2), we first subtitute v_{lp}^{∞} and do some algebra (again noting that $\|v_\star\|_2 = \|w_\star\|_2$) to get:

$$\|B_0^\top v_\star - B_0^\top v_{\mathsf{lp}}^{\infty}\|_2 = \|B_0^\top (B_0\Sigma B_0^\top)^{-1} B_0\Sigma B_0^\top v_\star - B_0^\top v_{\mathsf{lp}}^{\infty}\|_2 \tag{A.145}$$

$$= \|B_0^\top (B_0\Sigma B_0^\top)^{-1} B_0\Sigma (B_0 - B_\star)^\top v_\star\|_2 \tag{A.146}$$

$$\leq \sigma_{\max}(B_0^\top (B_0\Sigma B_0^\top)^{-1} B_0\Sigma)\sigma_{\max}(B_0 - B_\star)\|w_\star\|_2 \tag{A.147}$$

$$\leq \sigma_{\max}(B_0^\top (B_0\Sigma B_0^\top)^{-1} B_0\Sigma)\epsilon\|w_\star\|_2 \tag{A.148}$$

$$\leq \sigma_{\max}(B_0)^2\sigma_{\max}(\Sigma)\frac{1}{\sigma_{\min}(B_0\Sigma B_0^\top)}\epsilon\|w_\star\|_2 \tag{A.149}$$

$$\leq \frac{\sigma_{\max}(B_0)^2\sigma_{\max}(X)^2}{\sigma_{\min}(XB_0^\top)^2}\epsilon\|w_\star\|_2 \tag{A.150}$$

$$= \frac{\sigma_{\max}(B_0)^2\sigma_{\max}(ZF^\top)^2}{\sigma_{\min}(ZF^\top B_0^\top)^2}\epsilon\|w_\star\|_2 \tag{A.151}$$

$$\leq \frac{\sigma_{\max}(B_0)^2\sigma_{\max}(Z)^2}{\sigma_{\min}(Z)^2(\cos\theta_{\mathsf{max}}(R,S))^2}\epsilon\|w_\star\|_2 \tag{A.152}$$

$$\tag{A.153}$$

Where in the first line we subtituted in the closed form for v_{lp}^{∞} from Equation A.137, and in the last line we used the fact that $\sigma_{\max}(ZF^\top) \leq \sigma_{\max}(Z)$ since $F^\top$ has orthonormal rows, and $\sigma_{\min}(ZF^\top B^\top) = \sigma_{\min}(Z)\cos\theta_{\mathsf{max}}(R,S)$ as explained in Equation A.134 and Equation A.135.

So it suffices to bound the quantities in the RHS. Since B_0 has orthonormal rows, $\sigma_{\max}(B_0) = 1$.

No Gaussian assumption: For the first part of the Theorem (Equation A.132 where we make no Gaussian assumptions, but give a less quantitative bound), we just use the fact that $\sigma_{\max}(Z)$ is upper bounded almost surely, and $\sigma_{\min}(Z) \geq c'_\delta$ with probability at least $1-\delta$. This implies that for some $c_\delta > 0$ with probability at least $1-\delta$:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty}, B_0)} \leq \Big(\frac{c_\delta}{\cos\theta_{\mathsf{max}}(S,R)}\Big)^2\epsilon||w*||_2, \tag{A.154}$$

where $\epsilon = \|B_0 - B_\star\|_2$.

Gaussian assumption: For the second part of the Theorem (Equation A.133 where we assume P_z is Gaussian), we use results in random matrix theory to lower bound and upper bound $\sigma_{\min}(Z)$. For the lower bound we use a result from Rudelson & Vershynin (2009) (see page 4, in the equation below Equation 1.11), since $Z \in \mathbb{R}^{n \times m}$ is a matrix with each entry sampled from $\mathcal{N}(0,1)$, we get for all $t > 0$:

$$\mathbb{P}(\sigma_{\min}(Z) \leq \sqrt{n} - \sqrt{m} - t) \leq e^{-t^2/2} \tag{A.155}$$

With a bit of algebra, this gives us that with probability at least $1 - \delta$:

$$\sigma_{\min}(Z) \geq \sqrt{n} - \sqrt{m} - \sqrt{2 \log \frac{1}{\delta}} \tag{A.156}$$

We assumed $n \geq 5m$ and $n \geq 10 \log \frac{1}{\delta}$, so we get:

$$\sigma_{\min}(Z) \geq O(\sqrt{n}) \tag{A.157}$$

The upper bound is a standard matrix concentration bound—we use the high probability bound in Theorem 4.1.1 from Tropp (2015) (see Section 4.2.2 which calculates the variance statistic for rectangular Gaussian matrices, also notice the square on the LHS below):

$$\sigma_{\max}(Z)^2 \leq O(n \log \frac{n}{\delta}) \tag{A.158}$$

Substituting the lower and upper bounds on $\sigma_{\min}(Z)$ into Equation A.145 we get:

$$\|B_0^\top v_\star - B_0^\top v_{\mathsf{lp}}^\infty\|_2 \leq O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R,S))^2}\epsilon\|w_\star\|_2\Big) \tag{A.159}$$

Substituting into equation A.139, we have:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)} \leq O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R,S))^2}\epsilon\|w_\star\|_2\Big), \tag{A.160}$$

where $\epsilon = \|B_0 - B_\star\|_2$. Which completes the proof of the second part (Equation A.133).

Handling the rotation matrix U: We now handle the fact that the feature extractor distance involves the min over rotation matrices U: $d(B_0, B_\star) = \min_U \|UB_0 - B_\star\|_2$. Let $v_{\mathsf{lp}}^\infty(B_0)$ denote the linear probing head solution if we use a pretrained feature extractor B_0. We first note that for any k-by-k rotation matrix U, we have:

$$L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty(B_0), B_0) = L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty(UB_0), UB_0). \tag{A.161}$$

This follows from using the closed form we derived above for $v_{\mathsf{lp}}^\infty(B_0)$ and some simple algebraic manipulation (e.g., recall that $U^{-1} = Y^\top$):

$$\begin{aligned}
(UB_0)^\top v_{\mathsf{lp}}^\infty(UB_0) &= (UB_0)^\top (UB_0 X^\top X B_0^\top U^\top)^{-1} UB_0 X^\top X B_\star^\top v_\star && \text{(A.162)}\\
&= B_0^\top U^\top U (B_0 X^\top X B_0^\top)^{-1} U^\top U B_0 X^\top X B_\star^\top v_\star && \text{(A.163)}\\
&= B_0^\top (U^\top U)(B_0 X^\top X B_0^\top)^{-1} (U^\top U) B_0 X^\top X B_\star^\top v_\star && \text{(A.164)}\\
&= B_0^\top (B_0 X^\top X B_0^\top)^{-1} B_0 X^\top X B_\star^\top v_\star && \text{(A.165)}\\
&= B_0^\top v_{\mathsf{lp}}^\infty(B_0) && \text{(A.166)}
\end{aligned}$$

So the final predictors in both cases, $(UB_0)^\top v_{\mathsf{lp}}^\infty(UB_0)$ and $B_0^\top v_{\mathsf{lp}}^\infty(B_0)$ are identical. This means that the OOD error $L_{\mathsf{ood}}(v, B) = \|B^\top v - B_\star^\top v_\star\|_2$ is the same in both cases.

This means that we can just take the min over all rotation matrices U (where the first step follows since the identity matrix is a rotation matrix, and the second step is from Equation A.154):

$$L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty(B_0), B_0) \leq \min_U L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty(UB_0), UB_0) \tag{A.167}$$

$$\leq \min_U \Big(\frac{c(\delta)}{\cos\theta_{\mathsf{max}}(S, R)}\Big)^2 \|UB_0 - B_\star\|_2 ||w*||_2^2 \tag{A.168}$$

$$= \Big(\frac{c(\delta)}{\cos\theta_{\mathsf{max}}(S, R)}\Big)^2 d(B_0, B_\star) ||w*||_2^2, \tag{A.169}$$

which is as desired. We repeat the same thing for Equation A.160 to get Equation A.133 in the Theorem statement. □

A.4 LP vs. FT (OOD), non-asymptotic result for Gaussian covariates

Theorem 3.5 showed an asymptotic result: if the error $d(B_0, B_\star) \to 0$, then linear probing (LP) achieves better out-of-distribution (OOD) error than fine-tuning (FT). Here we give a more quantitative version of Theorem 3.5 for Gaussian covariates. The result can be extended to the case there each entry of P_z is independent and identically distributed, mean-zero, constant non-zero variance, but instead of Gaussian is sub-Gaussian with constant sub-Gaussian variance / moment—this can be shown using Theorem 1.1 in Rudelson & Vershynin (2009), which is a different matrix concentration inequality.

We show that LP does better than FT out-of-distribution if the error is less than a specific quantity (in terms of the representation dimension k, and the angles between the ID subspace S and the important pretrained directions $R_* = \text{rowspace}(B_\star)$).

Theorem A.15. *In the linear overparameterized setting, under the ID subspace assumption, assume the non-degeneracy conditions* $\cos\theta_{\mathsf{max}}(R_*, S) > 0$ *and* $\cos\theta_{\mathsf{max}}(R_*, S^\perp) > 0$ *where* $R_* = \textit{rowspace}(B_\star)$. *Suppose the covariates are generated from a Gaussian distribution on the ID subspace* S, *so* $P_z = \mathcal{N}(0, I_m)$. *Let* $\|w_\star\|_2$ *be a fixed constant. Given failure probability* $1 \leq \delta > 0$, *for all* $w_\star, B_0, n, d, k, \epsilon$, *if* $n \geq 5m$, *and* $n \geq 10 \log \frac{1}{\delta}$, *if the error of the pretrained representation is not too high:*

$$d(B_0, B_\star) < O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp)(\cos\theta_{\mathsf{max}}(R_*, S))^2\delta^2}{\sqrt{k}\log(n/\delta)}\Big), \tag{A.170}$$

then with probability at least $1 - \delta$, *the OOD error of linear probing is lower (better) than for fine-tuning at all time steps* $t \geq 0$ *in the fine-tuning trajectory:*

$$L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i) < \inf_{t \geq 0} L_{\mathsf{ood}}(v_{\mathsf{lp}}^{\infty i}, B_0^i). \tag{A.171}$$

Proof. Let $\epsilon = d(B_0, B_\star)$. We first note that the condition in Equation A.170 implies that $d(B_0, B_\star) < O(\cos\theta_{\mathsf{max}}(R_*, S^\perp))$ and $d(B_0, B_\star) < O(\cos\theta_{\mathsf{max}}(R_*, S))$. This is because the cosine angles are between 0 and 1, δ is between 0 and 1, and k and n are at least 1. We now simplify and combine the linear probing and fine-tuning bounds.

Let $R_0 = \text{rowspace}(B_0)$. Warning: note that the Equation A.170 in the Theorem statement assumes

conditions on the angles between R_* (corresponding to the optimal representation) and the ID subspace S. However, our results that bounded the fine-tuning (Theorem 3.3) and linear probing (Lemma A.133) errors require conditions on the angles between R_0 (corresponding to the representation that linear probing and fine-tuning use) and S. So we have to be careful about this distinction, and use Lemma A.9 to relate the two, which we do below.

Fine-tuning: From Theorem 3.3, we get:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \geq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_0, S^\perp)}{\sqrt{k}} \frac{\min(\varphi, \varphi^2/\|w_\star\|_2)}{(1+\|w_\star\|_2)^2}\Big) - \epsilon. \tag{A.172}$$

Where φ is the head-error, which we lower bounded in Lemma A.12—subtituting this bound and noting that $\min(\varphi, \varphi^2) = O(\varphi^2)$, $\|v_\star\|_2 = \|w_\star\|_2$ (which we assumed is a constant), this gives us:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \geq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_0, S^\perp)}{\sqrt{k}}\delta^2\Big) - \epsilon \tag{A.173}$$

Now, since $d(B_0, B_\star) = \epsilon$, we use Lemma A.9 to get that:

$$\cos\theta_{\mathsf{max}}(R_0, S^\perp) \geq \cos\theta_{\mathsf{max}}(R_*, S^\perp) - \epsilon \tag{A.174}$$

Subtituting this into Equation A.173, we get (notice the R_* instead of R_0 below):

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \geq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp) - \epsilon}{\sqrt{k}}\delta^2\Big) - \epsilon \tag{A.175}$$

Since $\epsilon \leq O(\cos\theta_{\mathsf{max}}(R_*, S^\perp))$, this can be simplified to:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{ft}}(t), B_{\mathsf{ft}}(t))} \geq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp)}{\sqrt{k}}\delta^2\Big) - \epsilon \tag{A.176}$$

Linear probing: From Lemma A.133, we get:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)} \leq O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R_0, S))^2}\epsilon\|w_\star\|_2\Big) \tag{A.177}$$

Again, we use Lemma A.9 to get:

$$\cos\theta_{\mathsf{max}}(R_0, S) \geq \cos\theta_{\mathsf{max}}(R_*, S) - \epsilon \tag{A.178}$$

Substituting into Equation A.177, and using the fact that $\epsilon \leq O(\cos\theta_{\mathsf{max}}(R_*, S))$, and since we assumed $\|w_\star\|_2$ is a constant, we get:

$$\sqrt{L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0)} \leq O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R_*, S))^2}\epsilon\Big) \tag{A.179}$$

Combining the two: We want to show that the OOD error of LP is less than for fine-tuning:

$$O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R_*, S))^2}\epsilon\Big) \leq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp)}{\sqrt{k}}\delta^2\Big) - \epsilon \tag{A.180}$$

We can bring the ϵ to the LHS, so this is equivalent to showing:

$$O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R_*, S))^2}\epsilon\Big) + \epsilon \leq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp)}{\sqrt{k}}\delta^2\Big) \tag{A.181}$$

Since $\log(n/\delta) \geq 1$ and $\cos\theta_{\mathsf{max}}(R_*, S))^2$ is between 0 and 1, this is equivalent to folding the ϵ inside the big-oh on the LHS:

$$O\Big(\frac{\log(n/\delta)}{(\cos\theta_{\mathsf{max}}(R_*, S))^2}\epsilon\|w_\star\|_2\Big) \leq O\Big(\frac{\cos\theta_{\mathsf{max}}(R_*, S^\perp)}{\sqrt{k}}\delta^2\Big) \tag{A.182}$$

But assuming the condition on ϵ in Equation A.170 of the Theorem statement, this is easy to show with a bit of algebra. □

A.5 Principal angles are likely non-zero

In Theorems 3.3, 3.5, and 3.6, we assumed the cosine of the largest principal angle between the representations and ID subspace (or complement of the ID subspace) was non-zero. For example, Theorem 3.5 assumed the largest principal angle between $R_* = \text{rowspace}(B_\star)$ and the ID subspace S is non-zero, and similarly for the angle between R_* and $S^\perp$. Having an angle of 0 is a degenerate condition. As an example, look at Figure 2—here the input dimension $d = 2$, the representation dimension $k = 1$, and the ID subspace S has dimension 1. The only way these angles can be 0 is if $B_\star^\top$ is exactly in the same direction as S or $S^\perp$, which seems like too much of a coincidence. intuitively, if nature introduces even a small amount of randomness in either the optimal representation or ID subspace, the angle will be non-zero.

This example was in two dimensions—to make this intuition a bit more formal in higher dimensions, we prove a simple claim. Lemma A.16 shows that if the S is a randomly selected m dimensional subspace, then the angles $\cos\theta_{\mathsf{max}}(R_*, S)$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp)$ are non-zero (and we get quantitative lower bounds on them).

Lemma A.16. *Let R be a fixed k dimensional subspace, and let S be a uniformly random m dimensional subspace (formally, a uniform measure on the Grassmannian manifold) in $\mathbb{R}^d$ with $m > k$. Then with probability at least $1 - \delta$,*

$$\cos\theta_{\mathsf{max}}(R, S) \geq \frac{\sqrt{m} - \sqrt{k} - \sqrt{2\log\frac{1}{\delta}}}{\sqrt{d\log\frac{2d}{\delta}}} \tag{A.183}$$

In addition, we get that $\cos\theta_{\mathsf{max}}(R, S) > 0$ almost surely (with probability 1).

If $m \geq 5k$ and $m \geq 10\log\frac{1}{\delta}$, then we get with probability at least $1 - \delta$:

$$\cos\theta_{\mathsf{max}}(R, S) \geq O\Big(\sqrt{\frac{m}{d\log\frac{2d}{\delta}}}\Big) \tag{A.184}$$

Recall that big-oh notation here means that the RHS is true for some universal constant (independent of any other problem parameters).

Proof. Note that principal angles are invariant if we rotate R and S by the same rotation matrix U. That is, if we let $U \in R^{d\times d}$ be a rotation matrix, and $E \in \mathbb{R}^{d\times k}$, $F \in \mathbb{R}^{d\times m}$ have orthonormal columns which form a basis for R and S respectively, then we have:

$$\cos\theta_{\mathsf{max}}(R, S) = \sigma_k(E^\top F) = \sigma_k((UE)^\top(UF)) \tag{A.185}$$

This symmetry means that we can fix S and instead consider R to be a uniform random k dimensional subspace on the Grassmannian manifold. Without loss of generality, we can also fix S to be the span of the first m standard basis vectors: $(e_1, \ldots, e_m)$, where $e_i \in \mathbb{R}^d$ has a 1 in the i-th entry and a 0 in every other entry.

Equivalently, let M_R be a d-by-k matrix, where each column is sampled independently from $N(0, I_d)$—since the columns of M_R span a uniformly random k-dimensional subspace, we can let R be range of M_R. This is equivalent to sampling each entry of M_R from $N(0, 1)$.

Let $c = \cos\theta_{\mathsf{max}}(R, S)$. From Lemma A.2, c can be written as:

$$c = \min_{r \in R, \|r\|_2 = 1} \|F^\top r\|_2 = \min_{r \in R, \|r\|_2 \geq 1} \|F^\top r\|_2 \tag{A.186}$$

Since R is the range of M_R, any $r \in R$ can be written as $r = M_R\lambda$ for some $\lambda \in \mathbb{R}^k$. We first show that $\|\lambda\|_2$ cannot be much smaller than $\|r\|_2$. This is because:

$$\|r\|_2 = \|M_R\lambda\|_2 \leq \sigma_{\max}(M_R)\|\lambda\|_2 \tag{A.187}$$

So this gives us:

$$\|\lambda\|_2 \geq \frac{\|r\|_2}{\sigma_{\max}(M_R)} \tag{A.188}$$

So every $r \in R$ can be written as $M_R\lambda$ where $\|\lambda\|_2$ is lower bounded as above.

We now simplify the definition of c, starting from Equation A.186.

$$c = \min_{r \in R, \|r\|_2 \geq 1} \|F^\top r\|_2 \tag{A.189}$$

$$\geq \min_{\|\lambda\| \geq 1/\sigma_{\max}(M_R)} \|F^\top M_R\lambda\|_2 \tag{A.190}$$

$$\geq \min_{\|\lambda\| \geq 1/\sigma_{\max}(M_R)} \sigma_{\min}(F^\top M_R)\|\lambda\|_2 \tag{A.191}$$

$$= \frac{\sigma_{\min}(F^\top M_R)}{\sigma_{\max}(M_R)} \tag{A.192}$$

So now we want to lower bound the ratio of two random matrices. We note that $F^\top M_R$ is a matrix of size (m, k) with each entry sampled independently from $N(0, 1)$ (this is because $F^\top$ simple selects the first m rows of M_R). M_R is a matrix of size (d, k) with each entry sampled independently from $N(0, 1)$.

Now, as in the Gaussian assumption step of the proof of Lemma A.14, we can apply standard matrix concentration bounds (page 4, below Equation 1.11, in Rudelson & Vershynin (2009) for the bound on $\sigma_{\min}$, and Theorem 4.1.1 in Tropp (2015) for the bound on $\sigma_{\max}$). We get that with probability at least $1 - \delta$:

$$\sigma_{\min}(F^\top M_R) \geq \sqrt{m} - \sqrt{k} - \sqrt{2\log\frac{1}{\delta}} \tag{A.193}$$

$$\sigma_{\max}(M_R) \leq \sqrt{d\log\frac{2d}{\delta}} \tag{A.194}$$

Note that we can use alternate bounds for $\sigma_{\min}$ in Rudelson & Vershynin (2009) that are sometimes tighter.

For the ratio of the two, we get that with probability at least $1-\delta$, we have:

$$c \geq \frac{\sigma_{\min}(F^\top M_R)}{\sigma_{\max}(M_R)} \geq \frac{\sqrt{m} - \sqrt{k} - \sqrt{2\log\frac{2}{\delta}}}{\sqrt{d\log\frac{2d}{\delta}}} \tag{A.195}$$

For interpretability, ignoring log factors this is approximately:

$$c \gtrapprox \frac{\sqrt{m} - \sqrt{k}}{\sqrt{d}} \tag{A.196}$$

The result when $m \geq 5k$ and $n \geq 10\log\frac{2}{\delta}$ follows with simple algebra.

For the result where we show $\cos\theta_{\mathsf{max}}(R, S) > 0$ almost surely, we recall that $F^\top M_R$ is a matrix of size (m, k) with each entry sampled independently from $N(0, 1)$. Then applying Lemma 3 in Xie et al. (2021a), we get that $\sigma_{\min}(F^\top M_R) > 0$ almost surely. Since $\sigma_{\max}(M_R)$ is finite, this gives us $\cos\theta_{\mathsf{max}}(R, S) > 0$ almost surely.

□

In our case, the dimension of the ID subspace S is m, and the dimension of $R_* = \text{rowspace}(B_\star)$ is k, with $k < m$ and $k < d - m$. If S is a uniformly random m-dimensional subspace, then $S^\perp$ is a uniformly random $d - m$ dimensional subspace. In this case, Lemma A.16 tells us that $\cos\theta_{\mathsf{max}}(R_*, S) > 0$ and $\cos\theta_{\mathsf{max}}(R_*, S^\perp) > 0$ almost surely, and gives us quantitative lower bounds for these angles.

A.6 LP vs. FT (ID)

We prove Proposition 3.6, where we show that if the representation is imperfect, then fine-tuning does better than linear probing, in-distribution.

Restatement of Proposition 3.6. *In the linear overparameterized setting, under the ID subspace assumption (Assumption 3.4), let $R_0 = \textit{rowspace}(B_0)$, and $R_{\mathsf{aug}} = \textit{Span}(\{w_\star\} \cup R_0)$. Suppose $w_\star \notin R_0$, $\cos\theta_{\mathsf{max}}(S, R_{\mathsf{aug}}) \neq 0$, and that fine-tuning converges to a local minimum of its loss, then fine-tuning does better ID almost surely: $L_{\mathsf{id}}(v_{\mathsf{ft}}^\infty, B_{\mathsf{ft}}^\infty) < L_{\mathsf{id}}(v_{\mathsf{lp}}^\infty, B_0)$ with probability 1 (over the randomness of the training examples).*

Proof. **Fine-tuning gets** 0 **ID loss**: It is well known from prior work (Laurent & von Brecht, 2018) that all local minima are global for optimizing two layer linear networks under convex losses (which is our setting), so if fine-tuning converges to a local minimum, it actually converges to a global minimum of the train loss. Since there exists parameters that achieve 0 loss on the training data (namely, $B_\star, v_\star$), this means fine-tuning gets 0 loss on the training data as well. So for all training examples x (that is, rows of X):

$$v_{\mathsf{ft}}^{\infty\top} B_{\mathsf{ft}}^\infty x = w_\star^\top x. \tag{A.197}$$

Since the models are linear, this implies that fine-tuning gets all examples in the span of the training examples correct as well. Since P_z has density, and the number of training examples n is at least as large as the ID subspace dimension m, the training examples span the ID subspace almost surely, so fine-tuning gets every example in $x \in S$ correct almost surely, giving us:

$$L_{\mathsf{id}}(v_{\mathsf{ft}}^\infty, B_{\mathsf{ft}}^\infty) = 0 \tag{A.198}$$

Linear probing gets positive ID loss: Lemma A.19 shows that the ID error of linear probing is greater than zero under the same assumptions as this Proposition, so

$$L_{\mathsf{id}}(v_{\mathsf{lp}}^{\infty}, B_0) > 0, \tag{A.199}$$

which finishes the proof.

□

We now state and prove the Lemmas that we used to lower bound the ID error of linear probing.

Lemma A.17 gives conditions for when the projection $F^\top w$ of a vector w is not contained in the projection $\text{Range}(F^\top E_0)$ of the column space of a matrix E_0.

Lemma A.17. *Let $w \in \mathbb{R}^d$ be a vector and $F \in \mathbb{R}^{d\times m}, E_0 \in \mathbb{R}^{d\times k}, E_{\mathsf{aug}} \in \mathbb{R}^{d\times(k+1)}$ have orthonormal columns, with Range(E_{aug}) = Span$(\{w\} \cup$ Range$(E_0))$. If $m > k$, we have:*

$$F^\top E_{\mathsf{aug}} \textit{ is full rank} \tag{A.200}$$

$$\overset{(a)}{\Longrightarrow} F^\top E_{\mathsf{aug}} \textit{ has higher rank than } F^\top E_0 \tag{A.201}$$

$$\overset{(b)}{\Longleftrightarrow} F^\top w \notin \textit{Range}(F^\top E_0) \tag{A.202}$$

Proof. The proof of (a) is clear—$F^\top E_{\mathsf{aug}} \in \mathbb{R}^{m\times(k+1)}$ has rank $k+1$ (since it is full rank and $m \geq k+1$), but $F^\top E_{\mathsf{aug}} \in \mathbb{R}^{m\times k}$ has rank at most k and is therefore lower rank. The assumption that $m > k$ is crucial here.

For (b), let $a_1, \ldots, a_k$ be the columns of E_0, which form a basis for $\text{Range}(E_0)$. Then $F^\top a_1, \ldots, F^\top a_k, F^\top w$ spans $\text{Range}(F^\top E_{\mathsf{aug}})$, while $F^\top a_1, \ldots, F^\top a_k$ spans $\text{Range}(F^\top E_0)$. So (notice the first list of vectors has an additional $F^\top w$) this means that $\dim(\text{Range}(F^\top E_{\mathsf{aug}})) \neq \dim(\text{Range}(F^\top E_0))$ iff $F^\top w$ is linearly independent from the rest, that is, $F^\top w \notin \text{Range}(F^\top E_0)$. Note that the rank of a matrix is the dimension of its range (column space), that is, $\dim(\text{Range}(A)) = \text{rank}(A)$ so this is what we wanted to show. □

The next Lemma says that if the projection $F^\top w_\star$ of the optimal linear model $w_\star$ onto the ID subspace S, is not contained in the projection $\text{Range}(F^\top E_0)$ of the features, then linear probing incurs non-zero ID error.

Lemma A.18. *In the linear overparameterized setting, under the ID subspace assumption, if $F^\top w_\star \notin$ Range$(F^\top E_0)$, then $L_{\mathsf{id}}(v_{\mathsf{lp}}^{\infty}, B_0) > 0$, where $E_0 \in \mathbb{R}^{d\times k}$ and $F \in \mathbb{R}^{d\times m}$ have orthonormal columns that form a basis for the feature rowspace $R_0 =$ rowspace(B_0) and ID subspace S respectively.*

Proof. We prove the contrapositive. Suppose $L_{\mathsf{id}}(v_{\mathsf{lp}}^{\infty}, B_0) = 0$. This means that:

$$L_{\mathsf{id}}(v_{\mathsf{lp}}^{\infty}, B_0) = \mathop{\mathbb{E}}_{x\sim P_{\mathsf{id}}}[(v_\star^\top B_\star x - v_{\mathsf{lp}}^{\infty\top} B_0 x)^2] = 0 \tag{A.203}$$

Since the squared error is always non-negative, this means that $v_{\mathsf{lp}}^{\infty\top} B_0 x = w_\star^\top x$ almost surely when $x \sim P_{\mathsf{id}}$ (recall that we defined $w_\star = B_\star^\top v_\star$). Recall P_{id} is defined as: first pick $z \in P_z$ (which has density) and then output $x = Fz$. Since P_z has density, this implies that we get all examples in the ID subspace S correct:

$$v_{\mathsf{lp}}^{\infty\top} B_0 x = w_\star^\top x \text{ for all } x \in S. \tag{A.204}$$

Since the columns of F form an orthonormal basis for S, this gives us (since each column of F is in S):

$$v_{\mathsf{lp}}^{\infty\top} B_0 F = w_\star^\top F. \tag{A.205}$$

Note that the rows of B_0 also form an orthonormal basis for R_0 just like the columns of E_0. So we can choose v with $v^\top E_0^\top = v_{\mathsf{lp}}^{\infty\top} B_0$. Then we have:

$$v^\top E_0^\top F = w_\star^\top F \Leftrightarrow F^\top E_0 v = F^\top w_\star \tag{A.206}$$

$$\Leftrightarrow F^\top w_\star \in \text{Range}(F^\top E_0), \tag{A.207}$$

where we took the transpose of both sides in the first step. This finishes the proof of the contrapositive. □

Finally, Lemma A.19 combines Lemma A.17 and Lemma A.18 to give a more interpretable condition for the ID error of linear probing: when the ID subspace S has some components along the optimal linear model $w_\star$ and the feature rowspace R_0, then linear probing has non-zero error. This is measured in terms of the principal angle $\cos\theta_{\mathsf{max}}(R_{\mathsf{aug}}, S)$ between the ID subspace S and R_{aug} which is the span of R_0 combined with $w_\star$. This angle will typically be non-zero—as an illustrative example, from Lemma A.16 we have that this angle will be non-zero almost surely if the ID subspace S is a uniformly random subspace.

Lemma A.19. *In the linear overparameterized setting, under the ID subspace assumption, let $R_0 =$ rowspace(B_0), and $R_{\mathsf{aug}} =$ Span$(\{w_\star\} \cup R_0)$. If $w_\star \notin R_0$ and $\cos\theta_{\mathsf{max}}(R_{\mathsf{aug}}, S) > 0$, then $L_{\mathsf{id}}(v_{\mathsf{lp}}^\infty, B_0) > 0$.*

Proof. After a bit of setup, the proof simply combines Lemma A.17 and Lemma A.18. If $w_\star \notin R_0$, then R_{aug} has dimension $k+1$. Let $E_{\mathsf{aug}} \in \mathbb{R}^{d\times(k+1)}, F \in \mathbb{R}^{d\times m}$ have orthonormal columns which form a basis for R_{aug} and S respectively. We assumed $\cos\theta_{\mathsf{max}}(R_{\mathsf{aug}}, S) = \sigma_{\min}(F^\top E_{\mathsf{aug}}) > 0$ which means that $F^\top E_{\mathsf{aug}}$ is full rank. The ID subspace assumption assumes that $m > k$. So from Lemma A.17, $F^\top w_\star \notin \text{Range}(F^\top E_0)$ where $E_0 \in \mathbb{R}^{d\times k}$ has orthonormal columns that form a basis for R_0. Then from Lemma A.18, $L_{\mathsf{id}}(v_{\mathsf{lp}}^\infty, B_0) > 0$. □

A.7 LP-FT

We start by showing a simple proposition, that if the initial feature extractor is perfect, then linear probing recovers the optimal weights.

Proposition A.20. *In the overparameterized linear setting, let $R =$ rowspace(B_0). If $B_0 = B_\star$, and $\cos\theta_{\mathsf{max}}(S, R) > 0$, then $L_{\mathsf{ood}}(v_{\mathsf{lp}}^\infty, B_0) = 0$ for all t.*

Proof. We first show that because $\cos\theta_{\mathsf{max}}(R, S) > 0$, the training loss for linear probing is strongly convex. Recall that the training loss is:

$$\widehat{L}(v, B) = \|XB^\top v - Y\|_2^2 \tag{A.208}$$

Linear probing keeps B fixed as $B_0 = B_\star$ and only tunes v, so we are interested in the Hessian of the loss with respect to v evaluated at $v, B_\star$:

$$\text{Hess}_v\, \widehat{L}(v, B_\star) = 2(B_\star X^\top)(B_\star X^\top)^\top \tag{A.209}$$

For strong convexity, it suffices to show that the min singular value of the Hessian is bounded away from 0 by a constant. Recall the definition of $\cos\theta_{\mathsf{max}}(R, S)$. For some F whose columns form an orthonormal basis for S, we have (since the rows of $B_\star$ form an orthonormal basis for R):

$$\sigma_k(B_\star F) = \cos\theta_{\mathsf{max}}(R, S) > 0 \tag{A.210}$$

Note that $B_\star F$ is a k-by-n matrix, so if the k-th singular value is positive it must be full rank. Since the columns of $X^\top$ span F (since we defined F to be such that the columns of F are an orthonormal basis for S, i.e. the rows of X), this means $B_\star X^\top$ is rank k. But that means the Hessian $(B_\star X^\top)(B_\star X^\top)^\top$ is rank k as well. So the linear probing loss is strongly convex.

Since the loss is strongly convex, there is a unique minimizer, and gradient flow converges to that. However, since we are in the well-specified setting, we know the training loss is:

$$\widehat{L}(v, B_\star) = \|XB_\star^\top v - XB_\star^\top v_\star\|_2^2 \tag{A.211}$$

So $v = v_\star$ achieves 0 loss and must be the (unique) minimizer. Therefore we have shown that linear probing converges to the unique minimizer $v_{\mathsf{lp}}^\infty = v_\star$, which attains 0 loss, as desired.

Note that the entire proof works out if $B_0 = UB_\star$ for some rotation matrix U. In that case, the Hessian becomes $2U(B_\star X^\top)(B_\star X^\top)^\top U^\top$ which is still rank k, since multiplying by square rotation matrices does not change the rank. In this case, the minimizer of the loss is $v = Uv_\star$, since $(UB_\star)^\top(Uv_\star) = B_\star^\top v_\star$. So linear probing converges to $v_{\mathsf{lp}}^\infty = Uv_\star$, which achieves 0 loss, as desired. □

Restatement of Proposition 3.7. *Suppose we have perfect pretrained features $B_0 = UB_\star$ for some rotation U. Let $R_0 =$ rowspace(B_0). Under the non-degeneracy conditions $\cos\theta_{\mathsf{max}}(R_0, S) \neq 0, \cos\theta_{\mathsf{max}}(R_0, S^\perp) \neq 0$:*

$$\forall t, L_{\mathsf{ood}}(B_{\mathsf{ft}}(t)^\top v_{\mathsf{ft}}(t)) > 0, \text{ if } v_0 \sim \mathcal{N}(0, \sigma^2 I) \text{ is randomly initialized (FT)}, \tag{A.212}$$

$$\forall t, L_{\mathsf{ood}}(B_{\mathsf{ft}}(t)^\top v_{\mathsf{ft}}(t)) = 0, \text{ if } v_0 \text{ is initialized to } v_{\mathsf{lp}}^\infty \text{ (LP-FT)}. \tag{A.213}$$

Proof. We first use Proposition A.20, which in the proof we showed still works if $B_0 = UB_\star$ for some rotation matrix U (which doesn't have to be identity). We get that $v_{\mathsf{lp}}^\infty = Uv_\star$. Then we have $B_0^\top v_{\mathsf{lp}}^\infty = B_\star^\top v_\star = w_\star$.

We now just show that the gradients with respect to the training loss $\widehat{L}$ at $(v_{\mathsf{lp}}^\infty, B_0)$ is 0, so gradient flow does not update the parameters at all.

The training loss is:

$$\widehat{L}(v, B) = \|XB^\top v - XB_\star^\top v_\star\|_2^2 \tag{A.214}$$

The derivative with respect to v is:

$$\partial_v \widehat{L}(v, B) = 2BX^\top(XB^\top v - XB_\star^\top v_\star) \tag{A.215}$$

Then since $B_0^\top v_{\mathsf{lp}}^\infty = B_\star^\top v_\star$, we have:

$$\partial_v \widehat{L}(v_{\mathsf{lp}}^\infty, B_0) = 0 \tag{A.216}$$

Next, the derivative with respect to B is:

$$\partial_B \widehat{L}(v, B) = 2v(XB^\top v - XB_\star^\top v_\star)^\top X \tag{A.217}$$

Table 3: **OOD accuracies** with 90% confidence intervals over 3 runs, for each of the three OOD domains in the split of DomainNet used by Tan et al. (2020); Prabhu et al. (2021). LP does better than FT across the board, and LP-FT does the best.

	Real	Painting	Clipart
Fine-tuning	55.29 (0.52)	50.26 (0.98)	60.93 (2.15)
Linear probing	**87.16 (0.18)**	74.50 (0.58)	77.29 (0.12)
LP-FT	**86.82 (0.51)**	**75.91 (0.73)**	**79.48 (0.90)**

Then since $B_0^\top v_{\mathsf{lp}}^\infty = B_\star^\top v_\star$, we have:

$$\partial_B \widehat{L}(v_{\mathsf{lp}}^\infty, B_0) = 0 \tag{A.218}$$

So since both the derivatives are 0, we have $\partial_t v_{\mathsf{ft}}(t) = 0$ and $\partial_B B_{\mathsf{ft}}(t) = 0$, which means the parameters don't change at all—at all times t we have $v_{\mathsf{ft}}(t) = U v_\star$ and $B_{\mathsf{ft}}(t) = U B_\star$ which gives us zero OOD loss: $L_{\mathsf{ood}}(B_{\mathsf{ft}}(t)^\top v_{\mathsf{ft}}(t)) = 0$ as desired. □

B More information on experiments

In this Appendix, we include more details on the datasets, pretraining methods, and adaptation methods. We also include the OOD accuracies for fine-tuning and linear-probing if we early stop and choose the learning rate based on OOD data, where we see that linear-probing is still typically better than fine-tuning OOD. Finally, we include results for additional baselines, pretraining models, and conclude with a discussion about the effective robustness of LP-FT.

B.1 Dataset and method details

We use a diverse range of datasets and pretraining strategies.

- **CIFAR-10 → STL**: We fine-tune or linear probe on CIFAR-10 (Krizhevsky, 2009) and test on STL (Coates et al., 2011). This is a benchmark used in domain adaptation papers (French et al., 2018). CIFAR-10 and STL share 9 classes, so we follow the common practice of omitting the unshared class in STL (which is the 'monkey' class) when reporting accuracies. We use a publicly available MoCo-v2 ResNet-50 checkpoint pretrained on unlabeled examples from ImageNet-1k (Russakovsky et al., 2015), and fine-tune for 20 epochs.

- **DomainNet**: We use the dataset splits in Tan et al. (2020) which is also used by follow-up work, e.g., in Prabhu et al. (2021). This is different from the original version of the DomainNet dataset (Peng et al., 2019), specifically Tan et al. (2020) note that some domains and classes contain many mislabeled outliers, so they select the 40 most common classes from the 'sketch', 'real', 'clipart' and 'painting' domains. We use the 'sketch' domain as ID, and all other domains ('real', 'clipart', 'painting') as OOD, and in the main paper we report the average accuracies across the OOD domains. In Table 3 we see that the *same trends hold for each of the three OOD domains*. We use a CLIP (Radford et al., 2021) pretrained ResNet-50 model, and fine-tune for 50 epochs (since this is a smaller dataset).

- **Living-17** and **Entity-30**: We use a publicly available MoCo-v2 ResNet-50 checkpoint pretrained on unlabeled examples from ImageNet-1k (Russakovsky et al., 2015), and fine-tune for 20 epochs. Note that Living-17 and Entity-30 are subpopulation shifts derived from ImageNet, but the pretraining is done on unlabeled data and does not see any OOD labels, following the pretraining and fine-tuning strategy in Cai et al. (2021). Entity-30 is a relatively large dataset that contains around 140K training examples.

- **FMoW Geo-shift**: We adapt the version of the dataset from (Koh et al., 2021). We use training data from 'North America' to fine-tune or linear probe, and then evaluate on validation data from Africa and Europe. We use a MoCo-TP (Ayush et al., 2020) checkpoint, pretrained on unlabeled FMoW satellite images. We fine-tune for 50 epochs here since the ID training dataset is smaller (around 20K examples).

- **CIFAR-10 $\rightarrow$ CIFAR-10.1** (Recht et al., 2018): We follow the same protocols as CIFAR-10 $\rightarrow$ STL, except we test on CIFAR-10.1.

- **ImageNet**: we linear probe or fine-tune on ImageNet (Russakovsky et al., 2015), and evaluate on **ImageNetV2** (Recht et al., 2019), **ImageNet-R** (Hendrycks et al., 2020), **ImageNet-A** (Hendrycks et al., 2019b), and **ImageNet-Sketch** (Wang et al., 2019). We use a CLIP pretrained ViT-B/16 (vision transformer), the largest publicly available CLIP model (Radford et al., 2021). We ran fine-tuning for 10 epochs, linear probing for 10 epochs. To equalize the runtime for LP-FT, we ran the linear probing stage for 5 epochs, and then the fine-tuning stage for 5 epochs. We used a batch size of 128 for all methods.

Tuning for ImageNet experiments. We swept over three learning rates for fine-tuning (0.0001, 0.0003, 0.001) and linear probing (0.01, 0.03, 0.1)—as is standard we use larger learning rates for linear probing. For LP-FT, we swept over 3 learning rates (0.01, 0.03, 0.1) for the 5-epoch linear probing step. We took the run that had the best ImageNet (ID) validation accuracy, and then swept over 3 learning rates (0.00001, 0.00003, 0.0001) for the 5-epoch fine-tuning step—we use a lower learning rate for LP-FT since the experiments on the other datasets suggested that the optimal learning rate that maximizes *ID validation accuracy* for LP-FT is smaller. We did not find the comparisons to be particularly sensitive to learning rate choice.

Augmentations for ImageNet experiments. We used augmentations for fine-tuning, and no augmentations for linear probing, following Kornblith et al. (2019). This might raise a question of whether linear probing and LP-FT do better OOD because of the lack of augmentations. So as an ablation we also tried fine-tuning without augmentations, however that led to worse accuracy (than fine-tuning with augmentations) both ID and OOD. We now give details on the preprocessing and augmentations that we used. On ImageNet, for linear probing and LP-FT, we used no augmentations—we just resized each image so that the smaller side has size 224 with bicubic interpolation, and then center-crop to a 224-by-224 image. For fine-tuning, we used augmentations: specifically we use RandomResizedCrop in TorchVision, with the default arguments and setting the size of the crop to 224, and then apply a random horizontal flip.

Notes on pretrained model choice. We note that our results say that the pretraining has to be good (e.g., at least get reasonable accuracy ID) for linear probing to outperform fine-tuning OOD. So, for example, we use a model pretrained on unlabeled satellite images for the satellite image dataset—if we pretrain the model on ImageNet, we expect that fine-tuning might do better. Similarly, for DomainNet we use a CLIP pre-trained

Table 4: **OOD accuracies** with 90% confidence intervals over 3 runs, when fine-tuning gets to choose learning rate and early stop, and linear probing gets to choose ℓ_2 regularization weights, on OOD data. We see that linear probing still typically does better OOD (the only flip from before is on FMoW).

	CIFAR-10.1	STL	Ent-30	Liv-17	DomNet	FMoW
FT	**92.27 (0.36)**	85.97 (0.38)	64.09 (0.19)	78.63 (0.53)	59.43 (2.49)	**40.23 (3.12)**
LP	82.67 (0.22)	**86.53 (0.01)**	**69.15 (0.13)**	**82.39 (0.14)**	**79.91 (0.24)**	37.12 (0.01)

	ImNetV2	ImNet-R	ImNet-Sk	ImNet-A	Average
FT	**71.5 (-)**	52.4 (-)	40.5 (-)	27.8 (-)	61.3
LP	69.7 (-)	**70.9 (-)**	**46.4 (-)**	**46.1 (-)**	67.1

model, which is pretrained on the very large WebImageText dataset, and sees a variety of photo and sketch like images. Pretraining on ImageNet alone does not lead to high accuracies on DomainNet (features are not very good), so we do not necessarily expect linear probing to outperform fine-tuning with these lower quality features (for example, see the MoCo ablation in our main paper where we used a worse pretrained model, and fine-tuning did better OOD).

Sanity check of fine-tuning implementation. As a sanity check of our implementation, fine-tuning did substantially better than training from scratch on all datasets (both ID and OOD) and matched existing fine-tuning numbers where available (e.g. ResNet50 on CIFAR-10 (Chen et al., 2020b) and Entity-30 (Cai et al., 2021)). Fine-tuning and linear probing also both do substantially better than training from scratch, ID and OOD, across the datasets. For example, on Living-17, training from scratch gets 89.3% ID and 58.2% OOD (Santurkar et al., 2020) which is over 5% worse ID and nearly 20% worse OOD, than all the adaptation methods. For reference linear probing gets 96.5% ID and 82.2% OOD, and fine-tuning gets 97.1% ID and 77.8% OOD. This is even though training from scratch was run for 300 epochs, which is 15 times longer than fine-tuning and LP-FT.

B.2 Target early stopping

In the main paper, one ablation we mention is early stopping each fine-tuning method and choose the best learning rate based on target validation accuracy. As expected, fine-tuning does improve a little, but linear probing (average accuracy: 67.1%) is still better than fine-tuning (average accuracy: 61.3%). Table 4 shows the full results for all datasets.

B.3 Feature change

We examine how much the features changed for ID and OOD examples in each dataset. Specifically, for each dataset, for each input example in the held out validation set, we computed the Euclidean distance of the ResNet-50 features before and after fine-tuning. We averaged these numbers across the dataset, showing the results for ID validation examples in Table 5, and for OOD examples in Table 6.

The feature distortion theory predicts that the features for ID examples change more than for OOD examples. This bears out in 9 out of 10 cases, that is all cases except for FT on FMoW. To see this, compare each cell in Table 5 with the corresponding cell in Table 6—the former is higher in 9 out of 10 cases.

Table 5: **In-distribution (ID)**: Average distance that features move before and after fine-tuning or LP-FT, multiplied by 100 to make things easier to read. For linear probing the numbers are all 0, since the features are not tuned. As predicted by our theory, we see that features for ID examples (this table) move more than features for OOD examples (Table 6). Both sets of features change substantially less for LP-FT. As usual we show 90% confidence intervals over three runs.

	CIFAR-10	Entity-30	Living-17	DomainNet	FMoW
FT	2.23 (0.03)	3.05 (0.02)	1.88 (0.01)	207.6 (12.31)	4.87 (0.15)
LP-FT	0.07 (0.00)	0.03 (0.01)	0.11 (0.01)	0.19 (0.03)	0.57 (0.19)

Table 6: **Out-of-distribution (OOD)**: Average distance that features move before and after fine-tuning or LP-FT, multiplied by 100 to make things easier to read. For linear probing the numbers are all 0, since the features are not tuned. As predicted by our theory, we see that features for ID examples (Table 5) move more than features for OOD examples (this table). Both sets of features change substantially less for LP-FT. As usual we show 90% confidence intervals over three runs.

	STL	Entity-30	Living-17	DomainNet	FMoW
FT	1.70 (0.04)	2.60 (0.02)	1.67 (0.01)	159.97 (16.23)	5.62 (0.30)
LP-FT	0.04 (0.00)	0.02 (0.00)	0.09 (0.01)	0.18 (0.02)	0.54 (0.17)

The feature distortion theory says that this large feature change is caused because the head is randomly initialized—since the head needs to be updated by a large amount, the feature extractor is also updated a lot because the updates are coupled. Our theory predicts that if the head is initialized via linear probing then the feature extractor should change a lot less for both ID and OOD examples. As predicted by the theory, across all the datasets in Table 5 and Table 6, the features change a lot less for LP-FT than for FT. For example, on CIFAR-10, the features change 30$\times$ less for LP-FT than for FT.

These results suggest that fine-tuning underperforms OOD, and LP-FT does well ID and OOD, for the reasons predicted by the feature distortion theory.

B.4 Additional architectures, fine-tuning methods

The main contributions of our paper are conceptual understanding and theory. However, to strengthen the empirical investigation we ran two additional models (a CLIP vision transformer and CLIP ResNet-50), as well as three additional fine-tuning heuristics. We focus on the Living-17 dataset because some of these ablations require lots of compute and can take a long time to run on all the datasets.

Architectures and pretraining source: In the main paper, we showed results when initializing with a MoCo-v2 ResNet-50 model pretrained on unlabeled ImageNet examples. Here we examine how the results change when we 1. Use a ResNet-50 model pretrained on CLIP's WebImageText dataset (Table 7), and, 2. Use a much larger vision transformer model (ViT-B/16) pretrained on CLIP's WebImageText dataset (Table 8)—this is the largest publicly available CLIP model at the time of writing. We see that similar findings to our main paper hold—fine-tuning does better than linear probing ID, but does worse than linear probing

Table 7: ID and OOD accuracies on Living-17 using a CLIP ResNet-50 model pretrained on the WebImage-Text dataset, instead of unlabeled ImageNet examples. Similar findings hold—here fine-tuning does similarly to linear probing ID, but does worse than linear probing OOD. LP-FT does better than both ID, and closes 86% of the gap OOD. As usual we show 90% confidence intervals over three runs.

	ID	OOD
LP	94.7 (0.2)	**78.6 (0.5)**
FT	94.7 (0.1)	67.3 (0.8)
LP-FT	**95.6 (0.2)**	77.0 (0.6)

Table 8: ID and OOD accuracies on Living-17 using a CLIP ViT-B/16 (Vision Transformer) model pretrained on the WebImageText dataset, instead of unlabeled ImageNet examples. This is the largest publicly available CLIP model that we could find. The same findings hold—fine-tuning does better than linear probing ID, but does worse than linear probing OOD. LP-FT does better than both ID, and closes 75% of the gap OOD. As usual we show 90% confidence intervals over three runs.

	ID	OOD
LP	97.5 (0.1)	**87.6 (0.5)**
FT	97.8 (0.0)	81.5 (2.1)
LP-FT	**98.0 (0.0)**	86.1 (0.1)

('underperforms') OOD. Finally, LP-FT does better than both methods ID, and closes most (75%-90%) of the gap OOD.

These results are from early stopping on ID validation data. If we early stop on OOD validation data, LP-FT achieves $87.9 \pm 0.4\%$ OOD accuracy, and LP gets $88.3 \pm 0.2\%$ OOD accuracy and here there is no statistically significant difference between the two. On the other hand, even if we early stop on OOD validation data, fine-tuning gets $84.4 \pm 0.5\%$ OOD accuracy which is lower.

Fine-tuning heuristics: Transfer learning (initializing with a pretrained model, and then adapting it to a downstream task) is the standard way to build modern ML models, because it improves accuracy and speeds up training. Since this paradigm is so widely used, there are many heuristics people use when training their models (as mentioned in the main paper, LP-FT has sometimes been used as a heuristic as well, although not in the context of OOD). We showed that LP-FT is one way to do well ID and OOD, but we hope that our theory leads to even better fine-tuning algorithms.

In this section, we compare LP-FT with additional fine-tuning heuristics: using a larger learning rate for the head layer, regularizing the features towards their original values, and side-tuning (Zhang et al., 2020) where we freeze the features but add a side-network.

The intuitions from our theory suggest two other potential ways to improve OOD accuracy: 1. We could use a higher learning rate on the linear layer, so that the linear layer learns quicker and the features do not get as distorted, and 2. We could regularize the weights of the feature extractor towards the pretrained initialization, to prevent feature distortion. These heuristics have been used in prior work on fine-tuning as well, for example

method 2 corresponds to L2-SP in (Li et al., 2018).

We run these two approaches on Living-17. For approach (1), we use a $10\times$ higher learning rate for the linear layer, and for approach (2) we regularize the Euclidean distance between the current feature extractor weights (so ignoring the linear head) from the pretrained weights, multiplying by a hyperparameter λ. We grid search over the same learning rates as fine-tuning for both methods, and in addition for (2) we grid search over $\lambda \in \{1.0, 0.1, 0.01, 0.001, 0.0001\}$, so this amounts to sweeping over 30 hyperparameters as opposed to just 6 for fine-tuning and LP-FT. For each hyperparameter configuration we run 3 replication runs with different seeds to reduce the estimation variance, and early stop and model select using ID data just like for fine-tuning and LP-FT. Just like for fine-tuning and LP-FT, we use a cosine learning rate decay and train for the same number of epochs. Indeed, we find that both (1) and (2) are able to close part of the OOD gap between fine-tuning and linear-probing. However, LP-FT does better than both methods ID and OOD. The full results are in Table 9.

We also compare with another method, (3) side-tuning (Zhang et al., 2020). Side-tuning freezes the pretrained features $g(x)$ but trains another 'side' model $s(x)$, and then outputs $v^\top(g(x) + h(x))$, where the head v and the parameters of the side model s are tuned. The intuition for trying this is that side-tuning also preserves the pretrained features which likely reduces feature distortion. In the supplementary of Zhang et al. (2020) they use a ResNet-50 for both the original model and the side model in their vision experiments, so we do the same. We sweep over twelve learning rates $(3 \cdot 10^{-5}, 1 \cdot 10^{-4}, 3 \cdot 10^{-4}, \ldots, 1.0, 3.0, 10.0)$, with three replication runs with different seeds for each learning rate. Just like for fine-tuning and LP-FT, we use a cosine learning rate decay and train for the same number of epochs, and we early stop and model select using ID validation data. We checked that the best learning rate was not at the boundary of the grid search. On OOD, side-tuning (81.0%) improves over fine-tuning (77.7%). However, side-tuning doesn't do as well ID. LP-FT did better ID and OOD. This could be because side-tuning does not get to refine the pretrained features for the ID task—while the side-network is powerful enough to learn good features, it is initialized randomly and effectively trained from scratch, so it might not be able to learn these good features on the limited sized training dataset (around 40K examples). The results are also in Table 9.

We also include results for training from scratch in Table 9—these results are from Santurkar et al. (2020). Note that training from scratch was done for 450 epochs, whereas fine-tuning was done for 20 epochs. As a sanity check, all the fine-tuning methods and linear probing do substantially better than training from scratch, both ID and OOD.

B.5 Discussion of effective robustness

LP-FT gets higher OOD accuracy than fine-tuning, but it sometimes gets higher ID accuracy as well. Taori et al. (2020) and Miller et al. (2021) show that OOD accuracy can often be correlated with ID accuracy, and suggest examining the effective robustness: intuitively the extra gain in OOD accuracy than can be predicted from improved ID accuracy alone. Is LP-FT simply better in-distribution, or does it have higher effective robustness as well?

We start out by noting that linear probing clearly has higher effective robustness in most of our datasets. Linear probing does worse than fine-tuning ID so based on the effective robustness framework we would expect it to do worse than fine-tuning OOD as well. However, linear probing does better than fine-tuning OOD and therefore has higher effective robustness.

Table 9: ID and OOD accuracies on Living-17 including three additional fine-tuning heuristics, where we (1) Use a $10\times$ larger learning rate for the head, or (2) Regularize the Euclidean distance of the feature extractor weights to the pretrained initialization, and (3) side-tuning where we freeze the pretrained model but add a side network that is fine-tuned. As a sanity check, all methods do better than training from scratch ID and OOD, and we show 90% confidence intervals over three runs. As per the intuitions from the feature distortion theory, these methods do mitigate feature distortion to some extent and improve OOD accuracy over fine-tuning. LP-FT does better than all methods ID and OOD—nonetheless, we believe that LP-FT is just the first step and hope that our theory can be used to inspire or derive better algorithms.

	ID	OOD
Scratch	92.4 (1.3)	58.2 (2.4)
LP	96.5 (0.1)	82.2 (0.2)
FT	97.1 (0.1)	77.7 (0.7)
FT (10x Linear)	97.2 (0.2)	80.4 (0.3)
FT (regularized)	97.1 (0.2)	80.0 (0.4)
Side-tuning	95.5 (0.4)	81.0 (0.7)
LP-FT	**97.8 (0.1)**	**82.6 (0.3)**

The solutions found by LP-FT also appear to have higher effective robustness than fine-tuning, because when they have similar ID accuracy, LP-FT does much better OOD. For a few pieces of evidence:

1. On CIFAR-10 $\rightarrow$ STL, there is no statistically significant difference between FT and LP-FT on ID, but LP-FT gets 8% higher accuracy OOD in Table 2.

2. If we look at checkpoints earlier in training for CIFAR-10 $\rightarrow$ STL we can exactly equalize ID accuracy and compare OOD accuracies. In-distribution, LP-FT and FT both get 97.2% accuracy, but OOD, LP-FT (90.2%) is much better than FT (81.8%).

3. Finally, in Figure 3 we plot the OOD accuracy against the ID accuracy for fine-tuning and LP-FT on Living-17. We plot these for three different pretrained models (CLIP ResNet-50, CLIP ViT-B/16, MoCo-V2 ResNet-50). We see that the ID-OOD line for LP-FT is above the line for FT indicating effective robustness.

Note that higher effective robustness does not mean a method is better. For example, a method A can have higher effective robustness B by doing a lot worse in-distribution even when they have the same OOD accuracy. In this case, A is clearly inferior since it does worse ID and same OOD, but has higher effective robustness because of its worse ID accuracy.

We believe the finding that linear probing and LP-FT has higher effective robustness than fine-tuning when the distributon shift is large is particularly interesting because Taori et al. (2020) and Miller et al. (2021) show that it is uncommon for methods to have higher effective robustness. In our case linear probing and LP-FT appear to consistently have higher effective robustness which suggests that with good transfer learning methods we can get both high in-distribution accuracy and higher effective robustness.

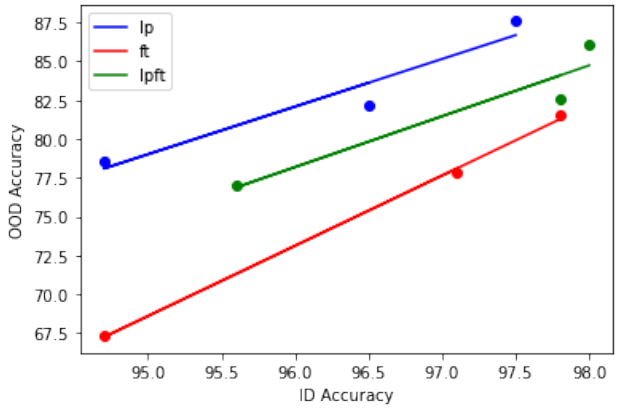


Figure 3: We plot the OOD accuracy against ID accuracy on Living-17 for the three methods we consider, when we start from three different pretrained models (CLIP ResNet-50, CLIP ViT-B/16, MoCo-V2 ResNet-50). The line for linear probing and LP-FT lie above fine-tuning which suggests that they have higher effective robustness. Each point is produced by averaging over three random seeds.

C Additional related work

Theoretical analysis of overparameterized models. Modern deep learning presents an interesting paradigm for theoretical analysis where the number of parameters is much larger than the number of training points. The model class is highly expressive and several solutions obtain zero training loss even in the presence of noise. Such overparameterized models have received a lot of interest recently especially with a focus on understanding "benign overfitting" or the phenomenon where fitting noisy training data to zero loss leads to classifiers that generalize well. By analyzing different linear overparameterized settings Belkin et al. (2019); Hastie et al. (2019); Bartlett et al. (2019); Muthukumar et al. (2020); Mei & Montanari (2019); Bibas et al. (2019) study various statistical properties such as the "double descent curve" in addition to benign overfitting. One important aspect of overparameterized models is that there is no unique minimizer of the training loss. We need some *inductive bias* which is typically implicit via the optimization procedure. Prior works study the statistical properties of the explicit inductive bias of minimum norm interpolation. In contrast, we study the effect of gradient based optimization from a particular pretrained initialization where we effectively capture the exact implicit inductive bias of gradient based fine tuning.