FINE-TUNING CAN DISTORT PRETRAINED FEATURES AND UNDERPERFORM OUT-OF-DISTRIBUTION

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, Percy Liang
Stanford University, Computer Science Department

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 → STL, CIFAR-10.1, 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. 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). 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).

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; Zhai et al., 2020; He et al., 2020), and even when testing OOD, prior work usually fine-tunes all parameters of their model (Hendrycks et al., 2019a; Miller 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 freezes these 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 a large distribution shift. We experiment on ten distribution shift benchmarks (BREEDS-Living17, BREEDS-Entity30, DomainNet, CIFAR → 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).

When and why 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
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.

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 accommodates 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 extrapolates better OOD because it preserves pretrained features, but does worse than 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; Ji 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 underperforms 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 rather 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 find that fine-tuning fails and LP-FT works, for the reasons predicted by our feature distortion theory: (1) fine-tuning changes the features for ID examples more than for OOD examples, leading to distortions; (2) LP-FT indeed changes both ID and OOD features 10–100× less than fine-tuning does; (3) LP-FT gets the best of both worlds, achieving 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 $f : \mathbb{R}^d \rightarrow \mathcal{Y}$ 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 $\ell$, we evaluate classifiers on:

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

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 3.1), $g_B$ is a deep network and in our theory (Section 3.2), $g_B$ 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). 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 error but worse ID error than fine-tuning (Section 3.2). Finally, we explain why linear probing then fine-tuning can mitigate this ID-OOD tradeoff (Section 3.3).

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. 2021b). 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 $\ell(\hat{y}, y) = (\hat{y} - y)^2$ is the squared loss.

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

Good pretrained features. For simplicity, we assume the models are well-specified i.e. $y = v^\top B_* x$ where $v_* \in \mathbb{R}^k$ and $B_* \in \mathbb{R}^{k \times d}$.

Note that $B_*$ and $v_*$ are only unique up to rotations, i.e., for any rotation matrix $U$, $((Uv_*)^\top U B_*) x = v_*^\top B x$. As in prior work (Tripuraneni et al. 2020) suppose $B_*$

---

1Our main contribution, analysis of fine-tuning (Theorem 3.2), does not require well-specification. We compare FT with LP by adapting earlier work on linear probing which requires well-specification.
and $B_0$ have been orthogonalized to have orthonormal rows. Suppose we have a pretrained feature extractor $B_0$ close to $B$, so $d(B_0, B) \leq \epsilon$ where the distance $d$ is defined as (where the min is over rotation matrices $U \in \mathbb{R}^{k \times k}$):

$$d(B, B') = \min_U \|B - UB'\|_2.$$  

(3.1)

**Training data.** Let $X \in \mathbb{R}^{n \times d}$, $X \neq 0$ be a matrix encoding $n$ training examples from $P_d$ where each of the $n$ rows is a training input. Let $Y \in \mathbb{R}^n$ be the corresponding outputs. Let $S =$ 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).

**Large OOD shift.** We assume that the OOD data contains examples outside the span of the training data. Formally, let $P_{	ext{ood}}$ have second moment $\Sigma = \mathbb{E}[xx^\top]$ where $x \sim P_{\text{ood}}$, for invertible $\Sigma$.

**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 $\hat{L}(v, B) = \|XB^\top v - Y\|_2^2$, the gradient flow differential equations for LP and FT are as follows:

$$\partial_t v_{t\text{lp}}(t) = -\nabla_B \hat{L}(v_{t\text{lp}}(t), B_{t\text{lp}}(t)), \quad \partial_t B_{t\text{lp}}(t) = -\nabla_B \hat{L}(v_{t\text{lp}}(t), B_{t\text{lp}}(t)),$$

(3.2)

$$\partial_t v_{t\text{ft}}(t) = -\nabla_B \hat{L}(v_{t\text{ft}}(t), B_{t\text{ft}}(t)), \quad \partial_t B_{t\text{ft}}(t) = 0,$$

(3.3)

initialized with $B_{t\text{lp}}(0) = B_{t\text{lp}}(0) = B_0$ and $v_{t\text{lp}}(0) = v_{t\text{lp}}(0) = v_0$. 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 $\sigma^2$, 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:

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

(3.4)

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

(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 that FT can distort features leading to poor OOD performance. We first explain the key intuitions and then present our formal theorem lower bounding the OOD error of FT (Section 3.2.2).

#### 3.2.1 Key Intuitions

We use two main observations to characterize when and why FT has higher OOD error than LP.

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 $\hat{L}(v, B) = \|XB^\top v - Y\|_2^2$ with respect to the feature extractor parameter $B$:

$$\nabla_B \hat{L}(v, B) = 2v(Y - XB^\top v)^\top X.$$  

(3.6)

By definition, if $u$ is a direction orthogonal to the training subspace $S =$ rowspace($X$), then $\nabla_B \hat{L}(v, B)u = 0$, 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
Figure 2: A toy example of our theory illustrating why fine-tuning distorts features, with inputs in 2D. Given input $x$, the ground truth output is $y = w_0^* x$. 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 $w_*$ 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.

shown in the Figure—both fine-tuned and linear probed estimators match the true parameter in the ID subspace (since $w_{lp}, w_{ft}, w_*$ 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 $B_{ft}$ 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. However, if the angle between $B_0$ and the $x$-axis is non-zero, 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 3.1** (largest principal angle). Let $A$ and $B$ be arbitrary subspaces, and $E$ and $F$ be matrices with orthonormal columns that span $A$ and $B$ respectively, with $r = \min(\dim(A), \dim(B))$. Then $\cos \theta_{\max}(A, B) = \sigma_r(E^T F)$, which is the $r$-th largest singular value of $E^T F$.

**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.

**Theorem 3.2.** In the overparameterized linear setting, let $S^\perp = \text{rowspace}(X)^\perp$, $R_0 = \text{rowspace}(B_0)$, and $v_*, B_*$ be the optimal parameters with $w_* = B_* v_*$. If $\cos \theta_{\max}(R_0, S^\perp) > 0$, then for all time steps $t$, the OOD error of the fine-tuning iterates $(B_{ft}(t), v_{ft}(t))$ is lower bounded:

$$\sqrt{L_{ood}(v_{ft}(t), B_{ft}(t))} \geq \sigma_{\min}(\Sigma) \left( \frac{\cos \theta_{\max}(R_0, S^\perp)}{\sqrt{k}} \right) \min(\varphi, \varphi^2 / \|w_*\|_2) - \epsilon), \quad (3.7)$$

where $\varphi^2 = (v_0^T v_*)^2 - (v_*^T v_*)^2$ is defined to be initial head alignment error and $\epsilon \geq d(B_0, B_*)$ 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 $v_{ft}(t)$ would have to become very similar to the optimal $v_*$ at some time $t$. The head initialization $v_0$ is random (or zero) and likely to be far from $v_*$ (measured by the alignment error $\varphi$), so the head would have to change a lot to get close to $v_*$. As we see from the fine-tuning gradient flow (12), $v_{ft}(t)$ and $B_{ft}(t)$ change in a “coupled” manner, and a “balancedness” invariant in [Du et al. 2018] holds across the fine-tuning trajectory. Correspondingly, if $v_{ft}(t)$ changes a lot and gets close to $v_*$, the features $B_{ft}(t)$ 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 ($\epsilon$).** To unpack the bound consider a special case where the pretrained features are perfect ($\epsilon = 0$). With perfect features, Proposition A.4 shows that linear probing gets zero OOD error. Theorem 3.2 shows that $L_{\text{ood}}(v_\infty(t),B_0(t)) > 0$ at all times $t$—so fine-tuning underperforms when the features are perfect.

**Alignment error of random head initialization ($\varphi^2$).** The lower bound (Equation A.4) increases as $\varphi^2$ increases, because the gradient updates to the head and feature extractor are coupled. If the head were somehow initialized perfectly at $v_\star$, fine-tuning updates may not increase the OOD error. However, when the head is randomly initialized as is standard in fine-tuning, the alignment error is high, leading to high OOD error. We use this insight in Section A.3 to show that better head initialization (via linear probing) improves OOD performance of fine-tuning.

### 3.3 Linear Probing vs. Fine-Tuning

In this section, we use our main theorem on fine-tuning (Theorem 3.2) 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 $B_\star$.

**Assumption 3.3 (ID subspace assumption).** We assume that the ID data lies on an $m$-dimensional subspace $S$ where $k < m < d - k$, and we have $n \ge m$ training examples. Formally, let $P_*$ 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 $z \sim P_*$.

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

$$L_{id}(v,B) = \mathbb{E}_{x \sim P_{id}}[(v_\star^T B^* x - v^T Bx)^2]$$  (3.8)

**OOD comparison:** Under mild non-degeneracy conditions, we show that as the feature extractor error $\epsilon$ 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 $\cos \theta_{\text{max}}(R_\star,S)$ and $\cos \theta_{\text{max}}(R_\star,S^\perp)$ are not 0 where $R_\star = \text{rowspace}(B_\star)$.

**Theorem 3.4 (Informal version of Theorem 3.2).** In the linear overparameterized setting, under the ID subspace assumption (Assumption 3.3), if $\cos \theta_{\text{max}}(R_\star,S) \neq 0$ and $\cos \theta_{\text{max}}(R_\star,S^\perp) \neq 0$ where $R_\star = \text{rowspace}(B_\star)$, then,

$$\frac{L_{ood}(v_\infty,B_0)}{L_{ood}(v_\infty(t),B_0(t))} \to 0 \text{ as } B_0 \to B_\star.$$  (3.9)

This holds for all times $t$ for FT (and therefore also for the limit $v_\infty^\star, B_\infty^\star$) and the LP iterates converge to $v_\infty^\star, B_\infty^\star$ 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 ID error (Lemma A.11) but fine-tuning has high OOD error (Theorem 3.4). We give a more formal version of Theorem 3.4 and a proof in Appendix A.3.

**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. The non-degeneracy condition on $R_{\text{aug}}$ below is similar to our previous results, and holds with probability 1 if the ID subspace is chosen randomly, from Lemma A.5.

**Proposition 3.5.** In the linear overparameterized setting, under the ID subspace assumption (Assumption 3.3), let $R_0 = \text{rowspace}(B_0)$, and $R_{\text{aug}} = \text{Span}\{w_\star\} \cup R_0$. Suppose $w_\star \not\in R_0$, $\cos \theta_{\text{max}}(S,R_{\text{aug}}) \neq 0$, and that fine-tuning converges to a local minimum of its loss, then fine-tuning does better ID almost surely: $L_{id}(v_\infty^\star, B_\infty^\star) < L_{id}(v_\infty^0, B_0)$ 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 the feature extractor 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 3.4, we see that the alignment error in the head initialization 
\[ \varphi^2 = |(v_0^\top v_s)^2 - (v_s^\top v_s)^2| \] plays an important role. The issue with FT was that under random or zero initialization, \( \varphi^2 \) 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 \( v_s \) 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. Note that in this case, linear probing alone gets zero OOD error—so Proposition 3.6 is just a first cut result to illustrate that if initialized well, full fine-tuning does not distort features.

**Proposition 3.6.** Given perfect pretrained features \( B_0 = U B_* \) for some rotation \( U \). Let \( R_0 = \text{rowspace}(B_0) \). Under the non-degeneracy conditions \( \cos \theta_{\max}(R_0,S) \neq 0, \cos \theta_{\max}(R_0,S^\perp) \neq 0 \):

\[ \forall t, L_{\text{ood}}(B_0(t)^\top v_0(t)) > 0, \text{if } v_0 \sim \mathcal{N}(0,\sigma^2 I) \text{ is randomly initialized (FT)}, \]  
\[ \forall t, L_{\text{ood}}(B_0(t)^\top v_0(t)) = 0, \text{if } v_0 \text{ is initialized to } v_0^\infty \text{ (LP-FT)}. \]

4 Experiments

We run experiments on ten benchmark datasets with deep neural networks and see that given good pretrained features, fine-tuning (FT) does better ID but worse OOD than linear probing (LP). As predicted by the theory, we find that LP-FT does better than both methods. Finally, we see that a number of predictions from the feature distortion theory hold up in practice. For more details on datasets, pretraining models, and experiment protocols, see Appendix B.

We use standard distribution shift datasets: DomainNet (Peng et al., 2019; Tan et al., 2020), BREEDS-Living-17 (Santurkar et al., 2020), BREEDS-Entity-30 (Santurkar et al., 2020), CIFAR-10 → STL (Krizhevsky, 2009), COIL-100 (Coates et al., 2011), French et al. (2018), CIFAR-10 → CIFAR-10.1 (Recht et al., 2018), ImageNet-IK (Russakovsky et al., 2015)—where the OOD test sets are ImageNetV2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2020), ImageNet-A (Hendrycks et al., 2019b), and ImageNet-Sketch (Wang et al., 2019)—, and FMoW Geo-shift which is adapted from the satellite remote sensing dataset Functional Map of the World (Christie et al., 2018; Koh et al., 2021). See Appendix B for more details on the datasets.

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., 2020), 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.

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 \( \ell_2 \)-regularized logistic regression classifier on frozen features from the penultimate layer of the pretrained model, selecting the best \( \ell_2 \)-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 \( \ell_2 \)-regularization), and ran each hyperparameter configuration once. In all cases, OOD data was only used for evaluation.

**Results.** Fine-tuning (FT) does better than linear probing (LP) on 5 out of 6 ID datasets (average accuracy of 85.1% for FT vs. 82.9% for LP, see Table C). 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 LP vs. 59.3% for FT, see Table C)—LP does better on all datasets except CIFAR-10.1 and ImageNetV2, where the OOD is designed to closely replicate the ID dataset. This matches
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.

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

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 similar (consistent with our theory). LP-FT does the best on all 10 datasets.

<table>
<thead>
<tr>
<th></th>
<th>STL</th>
<th>CIFAR-10.1</th>
<th>Ent-30</th>
<th>Liv-17</th>
<th>DomainNet</th>
<th>FMoW</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT</td>
<td>82.4 (0.4)</td>
<td>92.3 (0.4)</td>
<td>60.7 (0.2)</td>
<td>77.8 (0.7)</td>
<td>55.5 (2.2)</td>
<td>32.0 (3.5)</td>
<td></td>
</tr>
<tr>
<td>LP</td>
<td>85.1 (0.2)</td>
<td>82.7 (0.2)</td>
<td>63.2 (1.3)</td>
<td>82.2 (0.2)</td>
<td>79.7 (0.6)</td>
<td>36.6 (0.0)</td>
<td></td>
</tr>
<tr>
<td>LP-FT</td>
<td>90.7 (0.3)</td>
<td>93.5 (0.1)</td>
<td>62.3 (0.9)</td>
<td>82.6 (0.3)</td>
<td>80.7 (0.9)</td>
<td>36.8 (1.3)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ImNetV2</th>
<th>ImNet-R</th>
<th>ImNet-Sk</th>
<th>ImNet-A</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT</td>
<td>71.5 (-)</td>
<td>52.4 (-)</td>
<td>40.5 (-)</td>
<td>27.8 (-)</td>
<td>59.3</td>
</tr>
<tr>
<td>LP</td>
<td>69.7 (-)</td>
<td>70.6 (-)</td>
<td>46.4 (-)</td>
<td>45.7 (-)</td>
<td>66.2</td>
</tr>
<tr>
<td>LP-FT</td>
<td>71.6 (-)</td>
<td>72.9 (-)</td>
<td>48.4 (-)</td>
<td>49.1 (-)</td>
<td>68.9</td>
</tr>
</tbody>
</table>

our theoretical predictions. Our training datasets vary in size from 20K examples to over a million examples, so LP does not appear to perform better than FT 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 the pretrained features are not good.

### 4.3 Examining the Feature Distortion Theory

**Early stopping does not mitigate feature distortion.** 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. 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 says that linear probing does better than fine-tuning OOD, but only if the OOD and ID data are quite different, and the pretrained features are good—otherwise fine-tuning can do better OOD by adjusting the feature extractor ID.
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).

$\text{ID} \approx \text{OOD}$: 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. There is substantial evidence of FT outperforming LP in distribution (ID) including recent large-scale investigations [Kornblith et al. 2019; Chen et al. 2021; Zhong et al. 2020; Chen et al. 2020] (the only notable exception is in Peters et al. 2019). In FT when using ELMo representations, but worse using BERT). 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 (Gao 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 is updated, can perform 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. 2021b; Itama et al. 2021; Zhou et al. 2021) have been shown to improve OOD performance. Andressen et al. (2021) observe that through the course of fine-tuning, ID accuracy increases but OOD accuracy plateaus.

Mitigating ID-OOD tradeoffs. While LP-FT has sometimes been used as a fine-tuning heuristic (Levine et al. 2019; 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; Tripathi et al. 2020; Du et al. 2020). Recent works (Chua et al. 2021; Shachar et al. 2021) study fine-tuning, but in the underparameterized regime (where there is a unique global optimum) or assuming a balanced initialization. Prior works also focus on ID error, while we analyze OOD error. See Section 6 for additional related work on theory of overparameterized models.

6 CONCLUSION.

There is a strong trend towards leveraging pre-trained models to improve downstream performance, and whenever feasible, it is common to fine-tune all model parameters. In this work, we study 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.

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). In NLP, linear probing is not as good — here we could first prompt-tune (Lester et al. 2021) and then fine-tune the entire model. 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. Updated code is available at https://github.com/AnanyaKumar/transfer_learning and this CodaLab worksheet.

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 Philanthropy 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.
REFERENCES


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


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 $c \text{expr1}$ for some universal constant such that the statement holds. As an example, we can say $5x^2 \leq O(x^3)$ because there exists some universal constant ($c = 5$) such that $5x^2 \leq 5x^2$. More examples: we can also say $5x^2 \geq O(x^3)$ 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 $\Pi_R$ 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:

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

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

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

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

$$\cos \theta_{\max}(R,S) = \min_{r \in R, \|r\|_2 = 1} \|F^\top(r)\|_2 \tag{A.9}$$
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 $\dim(R) \leq \dim(S)$ (a crucial condition!), $F^\top E$ is a ‘tall’ matrix (it has more rows than columns) so we have:

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

The result now follows from some algebra:

$$
\cos \theta_{\max}(R,S) = \sigma_{\min}(F^\top E) = \min_{\|v\|_2=1} \|F^\top Ev\|_2 = \min_{r \in R, \|r\|_2=1} \|F^\top (r)\|_2. \tag{A.12}
$$

A.2 Feature distortion theorem

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

Restatement of Theorem 3.2. In the overparameterized linear setting, let $S^\perp = \text{rowspace}(X)^\perp$, $R_0 = \text{rowspace}(B_0)$, and $v_*, B_*$ be the optimal parameters with $w_* = B_* v_*$. If $\cos \theta_{\max}(R_0, S^\perp) > 0$, then for all time steps $t$, the OOD error of the fine-tuning iterates $(B_n(t), v_n(t))$ is lower bounded:

$$
\sqrt{L_{\text{ood}}(v_n(t), B_n(t))} \geq \sigma_{\min}(\Sigma) \left( \frac{\cos \theta_{\max}(R_0, S^\perp) \min(\phi, \varphi^2/\|w_*\|_2^2)}{\sqrt{k}} - \epsilon \right), \tag{A.14}
$$

where $\varphi^2 = [(v_0^\top v_*)^2 - (v_*^\top v_*)^2]$ is defined to be initial head alignment error and $\epsilon \geq d(B_0, B_*)$ 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 $v_{ft}^t B_{ft}^t x$ 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 $B_n(0) = B_0$. 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:

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

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:

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

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:

$$
v_0 v_0^\top - B_0 B_0^\top = v_{ft}^t v_{ft}^t - B_{ft}^t B_{ft}^t \tag{A.19}
$$
Proof. This follows by showing that the derivative is 0:

\[ \partial_{t} [\frac{\partial}{\partial_{t}} \frac{e_{f}^{T} e_{f}}{2} - B_{i}^{T} B_{i}] = 0 \]  

(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 \( u \in S^{+} \). We will show that this is implied by the condition on the principal angle: \( \cos \theta_{\text{max}}(R, S^{+}) > 0 \) where \( R = \text{rowspace}(B_{0}) \), which we assumed in Theorem 3.2. 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 \( \text{dim}(R) \leq \text{dim}(S) \). For all \( r \in R \) with \( \| r \|_{2} = \cos \theta_{\text{max}}(R, S) \), 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 \( c = \cos \theta_{\text{max}}(R, S) \). First, 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 \( E \in \mathbb{R}^{d \times \text{dim}(S)} \), \( F \in \mathbb{R}^{d \times \text{dim}(R)} \) 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 \( z \in \mathbb{R}^{\text{dim}(R)} \), \( c = \sigma_{\text{min}}(E^{T} F) > 0 \), which means that \( E^{T} F \in \mathbb{R}^{\text{dim}(R) \times \text{dim}(S)} \) has rank \( \text{dim}(R) \) since \( \text{dim}(R) \leq \text{dim}(S) \)—in other words, \( E^{T} F \) has full column rank since the column dimension is smaller than the row dimension. So \( z = E^{T} Fw \) for some \( w \in \text{rowspace}(E^{T} F) \). 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 \( r = Ez = EE^{T} Fw = EE^{T} s \). We note that \( \Pi_{R} = EE^{T} \) 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 \( z = E^{T} Fw \) — since \( w \in \text{rowspace}(E^{T} F) \), from Lemma A.4 we have:

\[ \| z \|_{2} \geq \sigma_{\text{min}}(E^{T} F) \| w \|_{2} = c \| w \|_{2}. \]  

(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 \( x \in \text{rowspace}(A) \), 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 \( A \):

\[ A = UDV^{T} \]  

(A.22)

Where \( U \in \mathbb{R}^{m \times r}, D \in \mathbb{R}^{r \times r}, V^{T} \in \mathbb{R}^{r \times n} \), where \( U \) and \( V \) have orthonormal columns, and \( D = \text{diag}(\sigma_{1}, ..., \sigma_{r}) \) is a diagonal matrix with \( \sigma_{1} \geq ... \geq \sigma_{r} \geq 0 \).

\[ \| Ax \|_{2} = \| UDV^{T} x \|_{2} \]  

(A.23)

\[ = \| DV^{T} x \|_{2} \]  

(A.24)

\[ \geq \sigma_{r} \| V^{T} x \|_{2} \]  

(A.25)

\[ = \sigma_{r} \| x \|_{2} \]  

(A.26)

\[ = \sigma_{r}(A) \| x \|_{2} \]  

(A.27)
Where for the fourth step, we used the fact that if \( x \in \text{rowspace}(V^\top) \) 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. \( \square \)

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

**Lemma A.7.**

\[
L_{ood}(v, B) = (B^\top v_* - B^\top x) \Sigma (B^\top v_* - B^\top x)^\top \leq \sigma_{\min}(\Sigma) \|B^\top v_* - B^\top x\|^2_2. \tag{A.28}
\]

**Proof.** Let \( x \sim P_{ood} \). We have,

\[
L_{ood}(v, B) = \mathbb{E}[(v^\top B_x - v^\top x B_x)^2] = \mathbb{E}[(B^\top v_* - B^\top x) xx^\top (B^\top v_* - B^\top x)] = (B^\top v_* - B^\top x)^\top \mathbb{E}[xx^\top](B^\top v_* - B^\top x) = (B^\top v_* - B^\top x)^\top \Sigma(B^\top v_* - B^\top x). \tag{A.29}
\]

The inequality follows immediately because \( \sigma_{\min}(A) \) (for a square matrix \( A \)) is simply the min over \( x \) with unit \( \ell_2 \) norm of \( x^\top A x \). \( \square \)

We now prove Theorem 4.2, following the 3 steps outlined in the main text.

**Proof of Theorem 4.2.** Let \( c = \cos \theta_{\text{max}}(R, S^\perp) \). From Lemma A.7, we have \( L_{ood}(v_{f,t}^t, B_{f,t}^t) \leq \sigma_{\min}(\Sigma) \|B^\top_* v_* - B^\top_{f,t} v_{f,t}^t\|^2_2 \) so it suffices to bound \( \|B^\top_* v_* - B^\top_{f,t} v_{f,t}^t\|_2 \).

Because it makes the proof much easier, we will prove the contrapositive, and then convert back to the original theorem statement. We assume \( \|B^\top_* v_* - B^\top_{f,t} v_{f,t}^t\|_2 \leq \Delta \), and will show that:

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

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 \( \|B_0 - U B_*\|_2 \leq \epsilon \) for some \( U \). We note that the setup is rotationally symmetric so without loss of generality we can suppose \( \|B_0 - B_*\|_2 \leq \epsilon \). This is because we can let \( B_{f,t} = U B_* \) and \( v_{f,t} = U v_* \), and we have \( w_* = B_{f,t}^\top v_* = (U B_* )^\top (U v_*) \), where \( w_* \) is the optimal classifier—so we can now write the entire proof in terms of \( B_* \) and \( v_* \).

**Step 1: Show that \( \|v_{f,t}^t - v_*\|_2 \leq \Delta / c \):** We first give intuition and then dive into the math. The key insight is to use the fact that in ‘many’ directions \( B_{f,t}^\perp \) and \( B_* \) are the same (formally, for all \( x \in S^\perp \), \( B_{f,t}^\perp x = B_0 x \)). But \( B_0 \) and \( B_* \) are close by assumption, which means that \( B_{f,t}^\perp \) and \( B_* \) are close in ‘many’ directions. Since we assumed in the contrapositive that \( v_{f,t}^t \perp B_{f,t}^\perp \) and \( v_* \perp B_* \), we get that \( v_{f,t}^t \) and \( v_* \) 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 \( v_* \), so the two vectors \( v_{f,t}^t \) and \( v_* \) 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:

\[
z = \frac{c}{\|v_{f,t}^t - v_*\|_2} (v_{f,t}^t - v_*) \tag{A.34}
\]

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 \( c = \cos \theta_{\max}(R, S^\perp) > 0 \), and \( y \in R \) with \( \|y\| = c \), from Lemma A.3 we can choose \( x \in S^\perp \) with \( \|x\| \leq 1 \) and \( f_R(x) = y \). Then, we have \( B_0x = z \).

From Proposition A.3, since \( x \in S^\perp \), \( B_0 \) does not change in directions of \( x \) when fine-tuning so we have: \( B_0x = B^t_{ft}x \).

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.

\[
\|v^t_{ft} - v_*\|_2 = \frac{1}{c}(v^t_{ft} - v_*)^\top \left( \frac{c(v^t_{ft} - v_*)}{\|v^t_{ft} - v_*\|_2} \right) \tag{A.35}
\]

[Algebra]

\[
= \frac{1}{c}(v^t_{ft} - v_*)^\top z \tag{A.36}
\]

[Definition of \( z \)]

\[
= \frac{1}{c}(v^t_{ft} - v_*)^\top B_0x \tag{A.37}
\]

[Since \( B_0x = z \)]

\[
= \frac{1}{c}(v^t_{ft} B^t_{ft} - v_* B_0)x \tag{A.38}
\]

[Algebra]

\[
= \frac{1}{c}(v^t_{ft} B^t_{ft} - v_* B_0)x \tag{A.39}
\]

\[
\leq \frac{1}{c} \|v^t_{ft} B^t_{ft} - v_* B_0\|_2 \|x\|_2 \tag{A.40}
\]

[Cauchy-Schwarz]

\[
\leq \frac{1}{c} \|v^t_{ft} B^t_{ft} - v_* B_0\|_2 \tag{A.41}
\]

[Since \( \|x\| \leq 1 \)]

\[
\leq \frac{1}{c} \|v^t_{ft} B^t_{ft} - v_* B_0\|_2 \tag{A.42}
\]

[Triangle inequality]

\[
\leq \frac{1}{c} \|v^t_{ft} B^t_{ft} - v_* B_0\|_2 + \frac{1}{c} \|v_* B_* - v_* B_0\|_2 \tag{A.43}
\]

[Taking transpose]

\[
= \frac{1}{c} \|B^t_{ft} v_* - B_* v_*\|_2 + \frac{1}{c} \|v_* B_* - v_* B_0\|_2 \tag{A.44}
\]

[Definition of max singular value]

\[
= \frac{1}{c} \|B^t_{ft} v_* - B_* v_*\|_2 + \frac{1}{c} \|v_* B_* - v_* B_0\|_2 \tag{A.45}
\]

\[
= \frac{1}{c} \|B^t_{ft} v_* - B_* v_*\|_2 \tag{A.46}
\]

[Since \( \sigma_{\max}(B_0 - B_0) \leq \epsilon \)]

\[
\leq \frac{\Delta + \epsilon \|v_*\|_2}{c} \tag{A.47}
\]

[Since \( \|B^t_{ft} v_* - B^t_{ft} v_*\|_2 \leq \Delta \)]

\[
\leq \frac{\Delta + \epsilon \|v_*\|_2}{c} \tag{A.48}
\]

Which shows that \( \|v^t_{ft} - v_*\|_2 \leq (\Delta + \epsilon \|v_*\|_2)/c \).

**Step 2A: Show that \( ||B^t_{ft}||_F^2 \) is small:** The key insight is to take the trace on both sides of Proposition A.3, which bounds the Frobenius norm of \( B^t_{ft} \) and therefore the operator norm.

Rearranging Proposition A.3, we have:

\[
B^t_{ft} B^t_{ft} = B_0 + v_* v_*^\top - v_0 v_0^\top \tag{A.49}
\]

Taking the trace everywhere, we get:

\[
\text{Tr}(B^t_{ft} B^t_{ft}) = \text{Tr}(B_0 B_0^\top) + \text{Tr}(v_* v_*^\top) - \text{Tr}(v_0 v_0^\top) \tag{A.50}
\]

For any matrix \( A \), \( \text{Tr}(AA^\top) = \|A\|_F^2 \), and for a vector \( v \) the Frobenius norm is just the \( \ell_2 \)-norm, so \( \text{Tr}(vv^\top) = \|v\|_2^2 \). So we have:

\[
||B^t_{ft}||_F^2 = \|B_0\|_F^2 + \|v_*\|_2^2 - \|v_0\|_2^2 \tag{A.51}
\]

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

\[
||B^t_{ft}||_F^2 \leq \|B_0\|_F^2 + \|v_*\|_2^2 \tag{A.52}
\]
Step 2B: Show that $\|B_0^T v_*\|_2^2 - \|B_{ft}^T v_*\|_2^2$ is small: This step doesn’t involve much insight, and is standard perturbation analysis—we simply factor the difference of squares and bound each term.

First, we bound $\|B_{ft}^T v_{ft}^* - B_{ft}^T v_*\|_2$:

$$
\|B_{ft}^T v_{ft}^* - B_{ft}^T v_*\|_2 \leq \sigma_{\max}(B_{ft}^T) \|v_{ft}^* - v_*\|_2 \\
\leq \|B_{ft}^T\| F \|v_{ft}^* - v_*\|_2 \\
\leq \sqrt{\|B_0\|_F^2 + \|v_*\|_2^2} \|v_{ft}^* - v_*\|_2 \\
\leq \sqrt{\|B_0\|_F^2 + \|v_*\|_2^2} \left( \frac{\Delta + \epsilon \|v_*\|_2}{c} \right)
$$

(A.53) (A.54) (A.55) (A.56)

Next, we bound $\|B_0^T v_* - B_{ft}^T v_*\|_2$:

$$
\|B_0^T v_* - B_{ft}^T v_*\|_2 \leq \|B_0^T v_* - B_*^T v_*\|_2 + \|B_*^T v_* - B_{ft}^T v_*\|_2 \\
\leq \sigma_{\max}(B_0 - B_*) \|v_*\|_2 + \|B_*^T v_* - B_{ft}^T v_*\|_2 \\
\leq \epsilon \|v_*\|_2 + \|B_*^T v_* - B_{ft}^T v_*\|_2 \\
\leq \epsilon \|v_*\|_2 + \|B_{ft}^T v_{ft}^*\|_2 + \|B_{ft}^T v_{ft}^* - B_{ft}^T v_*\|_2 \\
\leq \epsilon \|v_*\|_2 + \Delta + \sqrt{\|B_0\|_F^2 + \|v_*\|_2^2} \left( \frac{\Delta + \epsilon \|v_*\|_2}{c} \right)
$$

(A.57) (A.58) (A.59) (A.60) (A.61) (A.62)

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

$$
\|u\|_2^2 - \|v\|_2^2 = \|(u - v)^T (u + v)\| \\
\leq \|u - v\|_2 \|u + v\|_2 \\
\leq \|u - v\|_2 (2 \|u\|_2 + \|u - v\|_2)
$$

(A.63) (A.64) (A.65)

Applying this:

$$
\|B_0^T v_*\|_2^2 - \|B_{ft}^T v_*\|_2^2 \leq \|B_0^T v_* - B_{ft}^T v_*\|_2 (2 \|B_0^T v_*\|_2 + \|B_0^T v_* - B_{ft}^T v_*\|_2) \\
\leq \Delta_2 (2 \|B_0^T v_*\|_2 + \Delta_2) \\
\leq \Delta_2 (2 \|B_0^T v_*\|_2 + 2 \|B_0^T v_* - B_*^T v_*\|_2 + \Delta_2) \\
\leq \Delta_2 (2 \|v_*\|_2 + 2 \epsilon \|v_*\|_2 + \Delta_2) \\
= : \Delta_3
$$

(A.66) (A.67) (A.68) (A.69) (A.70)

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

We start from Proposition A.4:

$$
v_0 v_0^T - B_0 B_0^T = v_{ft}^T v_{ft}^T - B_{ft}^T B_{ft}^T
$$

(A.71)

The key step is to left multiply both sides by $v_*^T$ and right multiply both sides by $v_*$ to get:

$$
(v_0^T v_*)^2 - \|B_0^T v_*\|_2^2 = (v_{ft}^T v_*)^2 - \|B_{ft}^T v_*\|_2^2
$$

(A.72)

Rearranging, and then using Equation A.56, we get:

$$
\|v_{ft}^T v_*\|_2^2 - \|v_0^T v_*\|_2^2 = \|B_{ft}^T v_*\|_2^2 - \|B_0^T v_*\|_2^2 \leq \Delta_3
$$

(A.73)

This is close to what we want, except we have $\|v_{ft}^T v_*\|_2^2$ on the LHS instead of $(v_0^T v_*)^2$. We previously showed that $v_{ft}^T$ and $v_*$ are close, in Step 1, so with some algebra we can bound the
For Now for \( \Delta \) Using the fact that So for \( B \) \( \Delta \) like to express \( \Delta \) 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 \(|(v_\ast^T v_\ast)^2 - (v_0^T v_\ast)^2|\):

\[
|(v_\ast^T v_\ast)^2 - (v_0^T v_\ast)^2| \leq |(v_\ast^T v_\ast)^2 - (v_{f_\ell}^T v_\ast)^2| + |(v_{f_\ell}^T v_\ast)^2 - (v_0^T v_\ast)^2| \leq \Delta_4 + \Delta_3
\]  

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

\[
\Delta_4 = (\Delta/c)\|v_\ast\|_2^2 (2\|v_\ast\|_2 + (\Delta/c)) \\
\Delta_3 = \Delta_2 (2\|w_\ast\|_2 + 2\epsilon\|v_\ast\|_2 + \Delta_2) \\
\Delta_2 = \epsilon\|v_\ast\|_2 + \Delta + \sqrt{\|B_0\| F^2 + \|v_\ast\|_2^2} \left( \frac{\Delta + \epsilon\|v_\ast\|_2}{c} \right)
\]

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

\[
\epsilon\|v_\ast\|_2 + \Delta \leq \left( \frac{\Delta + \epsilon\|v_\ast\|_2}{c} \right)
\]

So for \( \Delta_2 \), up to constant factors we can ignore the \( \epsilon\|v_\ast\|_2 + \Delta \) term—this means we get:

\[
\Delta_2 \leq O \left( (\sqrt{k} + \|w_\ast\|_2) \left( \frac{\Delta + \epsilon\|w_\ast\|_2}{c} \right) \right)
\]

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

\[
\Delta_2 \leq O \left( \sqrt{k}(1 + \|w_\ast\|_2) \left( \frac{\Delta + \epsilon\|w_\ast\|_2}{c} \right) \right)
\]

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

\[
\Delta_2 \leq O \left( \sqrt{k}(1 + \|w_\ast\|_2)^2 \left( \frac{\Delta + \epsilon}{c} \right) \right)
\]

Now for \( \Delta_3 \), first note that \( \epsilon \leq 2 \), since \( B_\ast \) and \( B_0 \) have orthonormal rows so \( \|B_\ast - B_0\|_2 \leq 2 \). This means that \( \epsilon\|w_\ast\|_2 \leq \|v_\ast\|_2 \), so \( \Delta_3 \) simplifies to:

\[
\Delta_3 \leq O(\Delta_2(\|w_\ast\|_2 + \Delta_2)) = O(\Delta_2\|w_\ast\|_2 + \Delta_2)
\]

Substituting the bound for \( \Delta_2 \) into \( \Delta_3 \), we get:

\[
\Delta_3 \leq O \left( \sqrt{k}\|w_\ast\|_2 (1 + \|w_\ast\|_2) \left( \frac{\Delta + \epsilon\|w_\ast\|_2}{c} \right) + k(1 + \|w_\ast\|_2)^4 \left( \frac{\Delta + \epsilon\|w_\ast\|_2}{c} \right)^2 \right)
\]

For \( \Delta_4 \), we get:

\[
\Delta_4 \leq O \left( \|w_\ast\|_2 \left( \frac{\Delta}{c} + \|w_\ast\|_2 \left( \frac{\Delta}{c} \right) \right) \right)
\]

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

\[
\Delta_3 + \Delta_4 \leq \sqrt{k}w(1 + \|w_\ast\|_2)^2 \left( \frac{\Delta + \epsilon}{c} \right) + k(1 + \|w_\ast\|_2)^4 \left( \frac{\Delta + \epsilon}{c} \right)^2
\]
Wrap up i.e., taking the contrapositive: So we’ve shown that if \( \|B_i^T v_* - B_{ft}^T v_{ft}\|_2 \leq \Delta \), then:
\[
|v_i^T v_* - (v_0^T v_*)|^2 \leq \frac{\Delta + \epsilon}{c} \frac{w(1+\|w_*\|_2^2)}{\sqrt{k}} \frac{(\Delta + \epsilon)^2}{c^2} (1+\|w_*\|_2^2)^4 \frac{k}{4}
\]
(A.92)

We’d like to flip this around: suppose \( |v_i^T v_* - (v_0^T v_*)|^2 \geq \varphi^2 \) for some \( \varphi \). To lower bound \( \|B_i^T v_* - B_{ft}^T v_{ft}\|_2 \), we simply take the contrapositive of what we have proved. Let \( \Delta \) be given by:
\[
\Delta = \frac{c}{w(1+\|w_*\|_2^2)\sqrt{k}} \frac{(\Delta + \epsilon)^2}{c^2} (1+\|w_*\|_2^2)^4 \frac{k}{4}
\]
(A.93)

In this case with some algebra, we can show that:
\[
|v_i^T v_* - (v_0^T v_*)|^2 \geq \varphi^2 \geq \frac{\Delta + \epsilon}{c} \frac{w(1+\|w_*\|_2^2)}{\sqrt{k}} \frac{(\Delta + \epsilon)^2}{c^2} (1+\|w_*\|_2^2)^4 \frac{k}{4}
\]
(A.94)

To see this, we bound each of the terms in the RHS separately using our definition of \( \Delta \). Then, from the contrapositive of what we proved (compare with Equation A.92), we get:
\[
\|B_i^T v_* - B_{ft}^T v_{ft}\|_2 \geq \Delta
\]
(A.95)

Finally, we can massage \( \Delta \) to combine terms and make it look slightly nicer:
\[
\Delta \geq \frac{c}{\sqrt{k}} \frac{\min(\varphi, \varphi^2)}{(1+\|w_*\|_2^2)^2} - \epsilon
\]
(A.96)

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

A.3 LP vs. FT (OOD)

We now prove Theorem 5.4, which compares linear probing and fine-tuning in the linear overparameterized setting, when the ID data lies in a lower dimensional subspace. We first define a distance (formally, a pseudometric) between pretrained feature extractors, which measures the operator norm difference between them up to rotations (e.g., if \( B = U B' \) for a rotation matrix \( U \) then \( B \) and \( B' \) are equivalent since this means we can obtain one feature extractor’s representations from another just by rotating):

**Definition A.8 (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 \( U \in \mathbb{R}^{k \times k} \)):
\[
d(B, B') = \min_U \|B - U B'\|_2.
\]
(A.97)

We state a more precise version of Theorem 5.4—basically we fix all problem parameters except \( B_0 \) (which limits to \( B_* \)). 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 \( v_{ft}^i(t), B_{ft}^i \) denote the parameters at time \( t \) of fine-tuning if we initialize with \( v_0, B_0 \) (see Equation 5.3 for the fine-tuning updates). Let \( v_{lp}^{\infty}, B_0^\infty \) be the linear probing solution when initialized with \( v_0, B_0 \) (see Equation 5.3 for the linear probing updates). We note that the LP iterates converge to \( v_{lp}^{\infty}, B_0^\infty \) as a result of gradient flow on a convex problem.

Finally, Theorem 5.4 says that as the pretrained representations get better, linear probing does much better than fine-tuning OOD:

**Theorem A.9 (Formal statement of Theorem 5.4).** 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 \( v_*, B_* \). Assume the non-degeneracy conditions \( \cos \theta_{max}(R_*, S) > 0 \) and \( \cos \theta_{max}(R_*, S^\perp) > 0 \) where \( R_*= \text{rowspace}(B_*) \). Given a sequence of pretrained feature extractors \( \{B_0^i\}_{i=1}^{\infty} \) with \( B_0^i \rightarrow B_* \), where the limit is in the pseudometric given by Definition A.8, the ratio of OOD errors of linear probing and fine-tuning converges in probability to 0:
\[
\frac{L_{ood}(v_{lp}^{\infty}, B_0^\infty)}{\inf_{t \geq 0} L_{ood}(v_{ft}^i(t), B_{ft}^i)} \rightarrow 0 \text{ as } i \rightarrow \infty.
\]
(A.98)
Recall that we say a sequence of real-valued random variables converges in probability to $X_i$, we have that

$$P(|X_i| > \epsilon') \leq \delta.$$

(A.99)

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

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

$$\cos \theta_{\text{max}}(R^i, S_*^\perp) > \cos \theta_{\text{max}}(R_*, S_*^\perp)/2.$$  

(A.100)

Next, from Lemma A.13, we have that with probability at least $1 - \delta/2$, Head-Error($v_0$, $v_\ast$) = $|\langle v_0^\top u \rangle - \langle v_\ast^\top u \rangle|^2 \geq c_\delta$ for some $c_\delta > 0$. Plugging this into the fine-tuning bound in Theorem 2.3, this means that for all large enough $i$ with probability at least $1 - \delta/2$:

$$\inf_{t \geq 0} L_{\text{ood}}(v_0^i(t), B_0^i(t)) \geq c_\delta' - d(B_0^i, B_*),$$

(A.101)

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

$$\inf_{t \geq 0} L_{\text{ood}}(v_0^i(t), B_0^i(t)) \geq c_\delta'',$$

(A.102)

for some $c_\delta'' > 0$.

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

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

(A.103)

Plugging this into the RHS of Lemma A.13, Equation A.13, which upper bounds the OOD error of linear probing, we get that for all large enough $i$, with probability at least $1 - \delta/2$:

$$L_{\text{ood}}(v_{lp}^\infty i, B_0^i) \leq u_\delta(d(B_0^i, B_*))^2,$$

(A.104)

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

$$L_{\text{ood}}(v_{lp}^\infty i, B_0^i) \leq c_\delta' \epsilon.$$  

(A.105)

**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 $1 - \delta$:

$$\frac{L_{\text{ood}}(v_{lp}^\infty i, B_0^i)}{\inf_{t \geq 0} L_{\text{ood}}(v_0^i(t), B_0^i(t))} \leq \epsilon,$$

(A.106)

as desired.

\[\square\]

We now prove the Lemmas that we used in the above proof.
A.3.1 Convergence of Principal Angle

Theorem 5.2 assumes conditions on the angle between the perfect feature extractor $B_*$ 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 $B_*$. We show that if $B_0$ and $B_*$ are close, then the angles between their rowspaces to a third subspace $T$ (which could be the ID subspace $S$) is similar.

**Lemma A.10.** Given two feature extractors $B_0, B_* \in \mathbb{R}^{k \times d}$ with orthonormal rows, where $R_0 = \text{rowspace}(B_0), R_* = \text{rowspace}(B_*)$, and a subspace $T$ with dimension at least 1, we have:

$$|\cos \theta_{\text{max}}(R_0, T) - \cos \theta_{\text{max}}(R_*, T)| \leq d(B_0, B_*)$$

(A.107)

**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 $U \in \mathbb{R}^{k \times k}$:

$$\cos \theta_{\text{max}}(R_0, T) = \sigma_r(U B_0 F)$$

(A.108)

$$= \sigma_r(U B_0 F)$$

(A.109)

$$\geq \sigma_r(B_* F) - \sigma_1((U B_0 - B_*) F)$$

(A.110)

$$\geq \sigma_r(B_* F) - \sigma_1(U B_0 - B_*)$$

(A.111)

$$= \sigma_r(B_* F) - \|U B_0 - B_*\|_2$$

(A.112)

$$= \cos \theta_{\text{max}}(R_*, T) - \|U B_0 - B_*\|_2$$

(A.113)

$$\geq \min_U \|U B_0 - B_*\|_2$$

(A.114)

Here in the first step we used the definition of $\cos \theta_{\text{max}}$ (Definition 3.11), and the fact that $B_0^T$ has orthonormal columns which form a basis for $R_0$ (the rowspace of $B_0$), so in Definition 3.1 we can substitute $E = B_0^T$. To get Equation A.111 we used Weyl’s theorem, which bounds the singular value under perturbations: $\sigma_r(A + B) \geq \sigma_r(A) - \sigma_1(B)$. To get Equation A.113 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:

$$\cos \theta_{\text{max}}(R_0, T) \geq \cos \theta_{\text{max}}(R_*, T) - \min_U \|U B_0 - B_*\|_2 = \cos \theta_{\text{max}}(R_*, T) - d(B_0, B_*)$$

(A.115)

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

$$\cos \theta_{\text{max}}(R_0, T) - \cos \theta_{\text{max}}(R_*, T) \leq d(B_0, B_*)$$

(A.116)

\[\square\]

**Lemma A.11.** Given a sequence of pretrained feature extractors $\{B_i\}_{i=1}^\infty$ with $B_i^T \rightarrow B_*$, where $B_0, B_* \in \mathbb{R}^{k \times d}$ have orthonormal rows, let $R_i = \text{rowspace}(B_i^T), R_* = \text{rowspace}(B_*)$. Then for any subspace $T$, we have:

$$\cos \theta_{\text{max}}(R_i^T, T) \rightarrow \cos \theta_{\text{max}}(R_*, T), \text{as } i \rightarrow \infty.$$  

(A.117)

**Proof.** This follows directly from Lemma A.10. $B_0 \rightarrow B_*$ means $d(B_0, B_*) \rightarrow 0$. Then from Lemma A.10

$$|\cos \theta_{\text{max}}(R_i^T, T) - \cos \theta_{\text{max}}(R_*, T)| \rightarrow 0, \text{as } i \rightarrow \infty.$$  

(A.118)

This means $\cos \theta_{\text{max}}(R_i^T, T) \rightarrow \cos \theta_{\text{max}}(R_*, T)$ as $i \rightarrow \infty$. \[\square\]

A.3.2 Bounding the Head Error

We prove a lower bound on $\text{Head-Error}(v_0, v_*) = |(v_0^T v_*)^2 - (v_0^T v_*)^2|$, which was a key term in the fine-tuning lower bound (Theorem 5.2). Note that if the head is initialized as $v_0 = 0$, then $\text{Head-Error}(v_0, v_*) = |v_*^2| = |v_*|^2$. 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 $\sigma^2$, the head error will still typically be at least $\Omega(\|v_*\|_2)$. 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.12. For some universal constant \( c \), given \( a > 0 \), for all \( \nu^2 \) if \( u \sim N(0, \nu^2) \) then for all \( 0 \leq \delta \leq 1 \):

\[
P(|u - a| \leq c\delta a) \leq \delta \quad (A.119)
\]

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

\[
f(u) \leq O\left( \frac{1}{\nu} \exp\left( -\frac{9a^2}{2\cdot 10^2\nu^2} \right) \right) \quad (A.120)
\]

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 \( c' \geq 10 \) (it is OK to choose a larger universal constant than needed):

\[
f(u) \leq \frac{c'}{a} \quad (A.121)
\]

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

\[
P(|u - a| \leq \delta a) \leq \frac{2c'\delta a}{a} = 2c'\delta \quad (A.122)
\]

Now, we substitute \( \delta' = 2c'\delta \). We get for all \( \delta' \leq 2c'/10 \):

\[
P(|u - a| \leq \frac{1}{2c'}\delta'a) \leq \delta' \quad (A.123)
\]

Since \( c' \geq 10 \), \( 2c'/10 \geq 1 \), so the statement is true for all \( 0 \leq \delta' \leq 1 \).

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

**Lemma A.13.** For some universal constant \( c \), for all \( v_* \in \mathbb{R}^k \) with \( v_* \neq 0 \), \( \sigma \in \mathbb{R}^+ \), \( \delta \in [0, 1] \), if \( v_0 \sim N(0, \sigma^2 I_k) \), we have with probability at least \( 1 - \delta \):

\[
(\text{Head-Error}(v_0, v_*))^2 := |(v_0^T v_*)^2 - (v_*^T v_*)^2| \geq c\delta (v_*^T v_*)^2 \quad (A.124)
\]

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

**Suffices to bound \( |v_0^T v_* - v_*^T v_*| \):** We decompose the error:

\[
|(v_0^T v_*)^2 - (v_*^T v_*)^2| = |v_0^T v_* - v_*^T v_*|(|v_0^T v_* + v_*^T v_*|)
\]

\[
\geq |v_0^T v_* - v_*^T v_*|(|v_*^T v_*|) \quad (A.125)
\]

So we bound \( |v_0^T v_* - v_*^T v_*| \).

**\( v_0^T v_* \text{ is normally distributed:}** We note that \( v_0^T v_* \) is distributed as:

\[
v_0^T v_* \sim N(0, \sigma^2 v_*^T v_*) \quad (A.127)
\]

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

**Apply Gaussian anti-concentration lemma:** Then, from Lemma A.12, we have for some universal constant \( c \) that with probability at least \( 1 - \delta \):

\[
|v_0^T v_* - v_*^T v_*| \geq c\delta v_*^T v_* \quad (A.128)
\]

So substituting this back into Equation A.125, we get the desired result:

\[
|(v_0^T v_*)^2 - (v_*^T v_*)^2| \geq c\delta (v_*^T v_*)^2 \quad (A.129)
\]
A.3.3 Upper Bounding Linear Probing Error

We showed a lower bound for the OOD error of fine-tuning in Theorem 3.2. 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 \( \sigma_{\text{min}} \), unlike for \( \sigma_{\text{max}} \), and we included this lemma to be careful about these conditions):

**Lemma A.14.** 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:

\[
\sigma_{\text{min}}(AB) \geq \sigma_{\text{min}}(A)\sigma_{\text{min}}(B)
\]

**Proof.** For a tall matrix \( A \), we have:

\[
\sigma_{\text{min}}(A) = \min_{\|x\|_2 \leq 1} \|Ax\|_2
\]

So we have:

\[
\sigma_{\text{min}}(AB) = \min_{\|x\|_2 \leq 1} \|ABx\|_2 \geq \sigma_{\text{min}}(A)\sigma_{\text{min}}(B) \min_{\|x\|_2 \leq 1} \|x\|_2
\]

And \( \min_{\|x\|_2 \leq 1} \|x\|_2 = 1 \) which completes the proof. \( \square \)

**Lemma A.15.** In the linear overparameterized setting, under the ID subspace assumption, fix arbitrary \( P_z \). Then there exists \( c_\delta \) such that with probability at least \( 1 - \delta \), for all \( d,n,m,k,w \), feature extractors \( B_k, B_0 \), and ID subspaces \( S \) with corresponding \( F \) (whose columns are orthonormal and form a basis for \( S \)), if \( \cos \theta_{\text{max}}(S,R) > 0 \), we have:

\[
\sqrt{L_{\text{ood}}(v^\infty_{F},B_0)} \leq \left( \frac{c_\delta}{\cos \theta_{\text{max}}(S,R)} \right)^2 d(B_0, B_\star) \|w_\star\|_2
\]

If \( P_z \) is isotropic Gaussian so \( \mathcal{N}(0, I_m) \), then we derive a bound for \( c_\delta \) 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:

\[
\sqrt{L_{\text{ood}}(v^\infty_{F},B_0)} \leq O\left( \frac{\log(n/\delta)}{(\cos \theta_{\text{max}}(R,S))^2} d(B_0, B_\star) \|w_\star\|_2 \right)
\]

**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 \( \epsilon = \|B_\star - B_0\|_2 \). We first prove the bounds for \( \epsilon \), 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.14 we can write the min singular value of \( ZF^\top B_0^\top \) as:

\[
\sigma_{\text{min}}(ZF^\top B_0^\top) \geq \sigma_{\text{min}}(Z)\sigma_{\text{min}}(F^\top B_0^\top)
\]

Now from the definition of the principal angle (Definition 3.1), we have:

\[
\sigma_{\text{min}}(F^\top B_0^\top) = \cos \theta_{\text{max}}(R,S) > 0.
\]

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 \( \delta \) and \( P_z \), with probability at least \( 1 - \delta \):

\[
\sigma_{\text{min}}(Z) \geq c_\delta'
\]

Note that this also means that \( \sigma_{\text{min}}(ZF^\top B_0^\top) > 0 \) and so \( XB_0^\top =ZF^\top B_0^\top \) has full rank \( k \) almost surely. This also implies that \( B_0X \) is a matrix of shape \((k,k)\) that is invertible almost surely.
Main proof Since \( B_0 X^T X B_0^T \) is invertible almost surely, there is a unique global minimum (minimizing over \( v \)) to the loss optimized by linear probing:

\[
\arg\min_v \| X B_0^T v - X B_0^T v_\star \|_2^2 = (B_0 X^T X B_0^T)^{-1} B_0 X^T X B_0^T v_\star
\] (A.138)

We can see this by noting that the loss function on the LHS is strongly convex in \( v \) since the Hessian \( B_0 X^T X B_0^T \) is invertible. Then, gradient flow converges to the unique minimizer on the RHS, so:

\[
v_\text{lp}^\infty = (B_0 X^T X B_0^T)^{-1} B_0 X^T X B_0^T v_\star
\] (A.139)

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

\[
\sqrt{L_{\text{ood}}(v_\text{lp}^\infty, B_0)} = \| B_\star^T v_\star - B_0^T v_\text{lp}^\infty \|_2
\] (A.140)

\[
\leq \| (B_\star^T v_\star - B_0^T v_\star) + (B_0^T v_\star - B_0^T v_\text{lp}^\infty) \|_2
\] (A.141)

\[
\leq \| B_\star^T v_\star - B_0^T v_\star \|_2 + \| B_0^T v_\star - B_0^T v_\text{lp}^\infty \|_2
\] (A.142)

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

\[
\| B_\star^T v_\star - B_0^T v_\star \|_2 \leq \sigma_{\max}(B_\star - B_0) \| v_\star \|_2
\] (A.143)

\[
\leq \epsilon \| v_\star \|_2
\] (A.144)

\[
= \epsilon \| w_\star \|_2.
\] (A.145)

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

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

\[
\| B_0^T v_\star - B_0^T v_\text{lp}^\infty \|_2 = \| B_0^T (B_0 \Sigma B_0^T)^{-1} B_0 \Sigma B_0^T v_\star - B_0^T v_\text{lp}^\infty \|_2
\] (A.146)

\[
= \| B_0^T (B_0 \Sigma B_0^T)^{-1} B_0 \Sigma (B_0 - B_\star)^T v_\star \|_2
\] (A.147)

\[
\leq \sigma_{\max}(B_0^T (B_0 \Sigma B_0^T)^{-1} B_0 \Sigma (B_0 - B_\star)^T) \| v_\star \|_2
\] (A.148)

\[
\leq \sigma_{\max}(B_0^T (B_0 \Sigma B_0^T)^{-1} B_0 \Sigma) \sigma_{\max}(B_0 - B_\star) \| v_\star \|_2
\] (A.149)

\[
\leq \sigma_{\max}(B_0^T (B_0 \Sigma B_0^T)^{-1} B_0 \Sigma) \frac{1}{\sigma_{\min}(B_0 \Sigma B_0^T)} \| v_\star \|_2
\] (A.150)

\[
\leq \sigma_{\max}(B_0^T)^2 \sigma_{\max}(X^2) \sigma_{\min}(B_0 \Sigma B_0^T)^{-2} \| w_\star \|_2
\] (A.151)

\[
= \sigma_{\max}(B_0^T)^2 \sigma_{\max}(Z F^T B_0^T)^2 \sigma_{\min}(Z F^T B_0^T)^{-2} \| w_\star \|_2
\] (A.152)

\[
\leq \sigma_{\max}(B_0^T)^2 \sigma_{\max}(Z^2) \sigma_{\min}(Z F^T B_0^T)^{-2} \| w_\star \|_2
\] (A.153)

Where in the first line we substituted in the closed form for \( v_\text{lp}^\infty \) from Equation A.138, and in the last line we used the fact that \( \sigma_{\max}(Z F^T B_0^T) \leq \sigma_{\max}(Z) \) since \( F^T \) has orthonormal rows, and \( \sigma_{\min}(Z F^T B_0^T) = \sigma_{\min}(Z) \cos \theta_{\max}(R, S) \) as explained in Equation A.135 and Equation A.136.

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.133 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 \( 1 - \delta \):

\[
\sqrt{L_{\text{ood}}(v_\text{lp}^\infty, B_0)} \leq \left( \frac{c_\delta}{\cos \theta_{\max}(S, R)} \right)^2 \epsilon \| w_\star \|_2,
\] (A.155)
where \( \epsilon = \|B_0 - B_*\|_2 \).

**Gaussian assumption:** For the second part of the Theorem (Equation 4.13a), 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 (2008)](http://rpubs.com/paper) (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 \( N(0,1) \), we get for all \( t > 0 \):

\[
\mathbb{P}(\sigma_{\min}(Z) \leq \sqrt{n} - \sqrt{m} - t) \leq e^{-t^2/2}
\]

(A.156)

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

\[
\sigma_{\min}(Z) \geq \sqrt{n} - \sqrt{m} - \sqrt{2\log \frac{1}{\delta}}
\]

(A.157)

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

\[
\sigma_{\min}(Z) \geq O(\sqrt{n})
\]

(A.158)

The upper bound is a standard matrix concentration bound—we use the high probability bound in Theorem 4.1.1 from [Tropp (2012)](http://rpubs.com/paper) (see Section 4.2.2 which calculates the variance statistic for rectangular Gaussian matrices, also notice the square on the LHS below):

\[
\sigma_{\max}(Z)^2 \leq O(n\log \frac{n}{\delta})
\]

(A.159)

Substituting the lower and upper bounds on \( \sigma_{\min}(Z) \) into Equation 4.13b we get:

\[
\|B_0^T v_* - B_0^T v_{\infty}\|_2 \leq O \left( \frac{\log(n/\delta)}{(\cos \theta_{\max}(R,S))^2} \epsilon \|w_*\|_2 \right)
\]

(A.160)

Substituting into Equation 4.13c we have:

\[
L_{ood}(v_{\infty}^\infty(B_0),B_0) = O \left( \frac{\log(n/\delta)}{(\cos \theta_{\max}(R,S))^2} \epsilon \|w_*\|_2 \right).
\]

(A.161)

where \( \epsilon = \|B_0 - B_*\|_2 \). Which completes the proof of the second part (Equation 4.13b).  

**Handling the rotation matrix** \( U \): We now handle the fact that the feature extractor distance involves the \( \min \) over rotation matrices \( U \): \( d(B_0,B_*) = \min_U \|UB_0 - B_*\|_2 \). Let \( v_{\infty}^\infty(B_0) \) 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:

\[
L_{ood}(v_{\infty}^\infty(B_0),B_0) = L_{ood}(v_{\infty}^\infty(U B_0),U B_0).
\]

(A.162)

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

\[
(U B_0)^T v_{\infty}^\infty(U B_0) = (U B_0)^T(U B_0 X^T X B_0^T U^T)^{-1} U B_0 X^T X B_0^T v_*
\]

(A.163)

\[
= B_0^T (B_0 X^T X B_0^T)^{-1} U B_0 X^T X B_0^T v_*
\]

(A.164)

\[
= B_0^T (B_0 X^T X B_0^T)^{-1} (U^T U) B_0 X^T X B_0^T v_*
\]

(A.165)

\[
= B_0^T (B_0 X^T X B_0^T)^{-1} B_0 X^T X B_0^T v_*
\]

(A.166)

\[
= B_0^T v_{\infty}^\infty(B_0)
\]

(A.167)

So the final predictors in both cases, \( (U B_0)^T v_{\infty}^\infty(U B_0) \) and \( B_0^T v_{\infty}^\infty(B_0) \) are identical. This means that the OOD error \( L_{ood}(v,B) = \|B^T v - B_*^T v_*\|_2 \) 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 4.13b):

\[
L_{ood}(v_{\infty}^\infty(B_0),B_0) \leq \min_U L_{ood}(v_{\infty}^\infty(U B_0),U B_0)
\]

(A.168)

\[
\leq \min_U \left( \frac{c(\delta)}{\cos \theta_{\max}(S,R)} \right)^2 \|U B_0 - B_*\|_2 \|w_*\|_2
\]

(A.169)

\[
= \left( \frac{c(\delta)}{\cos \theta_{\max}(S,R)} \right)^2 d(B_0,B_*) \|w_*\|_2
\]

(A.170)

which is as desired. We repeat the same thing for Equation 4.13c to get Equation 4.13d in the Theorem statement.
A.4 LP vs. FT (OOD), Non-Asymptotic Result for Gaussian Covariates

Theorem A.13 showed an asymptotic result: if the error $d(B_0, B_*) \rightarrow 0$, then linear probing (LP) achieves better out-of-distribution (OOD) error than fine-tuning (FT). Here we give a more quantitative version of Theorem A.13 for Gaussian covariates. The result can be extended to the case where 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 (2008), 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 $R_* = \text{rowspace}(B_*)$).

**Theorem A.16.** In the linear overparameterized setting, under the ID subspace assumption, assume the non-degeneracy conditions $\cos \theta_{\max}(R_*, S) > 0$ and $\cos \theta_{\max}(R_*, S^\perp) > 0$ where $R_* = \text{rowspace}(B_*)$. Suppose the covariates are generated from a Gaussian distribution on the ID subspace $S$, so $P_z = N(0, I_m)$. Let $\|w_*\|_2$ be a fixed constant. Given failure probability $1 - \delta > 0$, for all $w_*, B_0, n, d, k, \epsilon$, if $n \geq 5m$, and $n \geq 10 \log \frac{1}{\delta}$, if the error of the pretrained representation is not too high:

$$d(B_0, B_*) < O\left(\frac{\cos \theta_{\max}(R_*, S^\perp)(\cos \theta_{\max}(R_*, S))^2 \delta^2}{\sqrt{k} \log(n/\delta)}\right),$$

(A.171)

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:

$$L_{ood}(v_{lp}^{\infty_i}, B_0^i) < \inf_{t \geq 0} L_{ood}(v_{lp}^{\infty_i}, B_0^i).$$

(A.172)

**Proof.** Let $\epsilon = d(B_0, B_*)$. We first note that the condition in Equation A.171 implies that $d(B_0, B_*) < O(\cos \theta_{\max}(R_*, S^\perp))$ and $d(B_0, B_*) < O(\cos \theta_{\max}(R_*, S))$. This is because the cosine angles are between 0 and 1, $\delta$ 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.171 in the Theorem statement assumes conditions on the angles between $R_0$ (corresponding to the optimal representation) and the ID subspace $S$. However, our results that bounded the fine-tuning (Theorem A.13) and linear probing (Lemma A.14) 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.10 to relate the two, which we do below.

**Fine-tuning:** From Theorem A.2, we get:

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

(A.173)

Where $\varphi$ is the head-error, which we lower bounded in Lemma A.13—substituting this bound and noting that $\min(\varphi, \varphi^2) = O(\varphi^2)$, $\|w_*\|_2 = \|w_0\|_2$ (which we assumed is a constant), this gives us:

$$\sqrt{L_{ood}(v_{ft}(t), B_{ft}(t))} \geq O\left(\frac{\cos \theta_{\max}(R_0, S^\perp)}{\sqrt{k}}\delta^2\right) - \epsilon$$  

(A.174)

Now, since $d(B_0, B_*) = \epsilon$, we use Lemma A.10 to get that:

$$\cos \theta_{\max}(R_0, S^\perp) \geq \cos \theta_{\max}(R_*, S^\perp) - \epsilon$$

(A.175)

Substituting this into Equation A.174, we get (notice the $R_*$ instead of $R_0$ below):

$$\sqrt{L_{ood}(v_{ft}(t), B_{ft}(t))} \geq O\left(\frac{\cos \theta_{\max}(R_*, S^\perp) - \epsilon}{\sqrt{k}}\delta^2\right) - \epsilon$$

(A.176)

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

$$\sqrt{L_{ood}(v_{ft}(t), B_{ft}(t))} \geq O\left(\frac{\cos \theta_{\max}(R_*, S^\perp)\delta^2}{\sqrt{k}}\right) - \epsilon$$

(A.177)
Linear probing: From Lemma A.17.1, we get:
\[ \sqrt{L_{ood}(v_{lp}^\infty, B_0)} \leq O\left( \frac{\log(n/\delta)}{\cos\theta_{\max}(R_0, S)}^2 \epsilon \|w_s\|_2 \right) \]  \hspace{1cm} (A.178)

Again, we use Lemma A.10 to get:
\[ \cos\theta_{\max}(R_0, S) \geq \cos\theta_{\max}(R_*, S) - \epsilon \]  \hspace{1cm} (A.179)

Substituting into Equation A.178 and using the fact that \( \epsilon \leq O(\cos \theta_{\max}(R_*, S)) \), and since we assumed \( \|w_s\|_2 \) is a constant, we get:
\[ \sqrt{L_{ood}(v_{lp}^\infty, B_0)} \leq O\left( \frac{\log(n/\delta)}{(\cos\theta_{\max}(R_*, S))^2} \epsilon \right) \]  \hspace{1cm} (A.180)

Combining the two: We want to show that the OOD error of LP is less than for fine-tuning:
\[ O\left( \frac{\log(n/\delta)}{(\cos\theta_{\max}(R_*, S))^2} \epsilon \right) \leq O\left( \frac{\cos\theta_{\max}(R_*, S^\perp)}{\sqrt{k}} \delta^2 \right) - \epsilon \]  \hspace{1cm} (A.181)

We can bring the \( \epsilon \) to the LHS, so this is equivalent to showing:
\[ O\left( \frac{\log(n/\delta)}{(\cos\theta_{\max}(R_*, S))^2} \right) + \epsilon \leq O\left( \frac{\cos\theta_{\max}(R_*, S^\perp)}{\sqrt{k}} \delta^2 \right) \]  \hspace{1cm} (A.182)

Since \( \log(n/\delta) \geq 1 \) and \( \cos\theta_{\max}(R_*, S)^2 \) is between 0 and 1, this is equivalent to folding the \( \epsilon \) inside the big-oh on the LHS:
\[ O\left( \frac{\log(n/\delta)}{(\cos\theta_{\max}(R_*, S))^2} \epsilon \|w_s\|_2 \right) \leq O\left( \frac{\cos\theta_{\max}(R_*, S^\perp)}{\sqrt{k}} \delta^2 \right) \]  \hspace{1cm} (A.183)

But assuming the condition on \( \epsilon \) in Equation A.17.1 of the Theorem statement, this is easy to show with a bit of algebra. \( \square \)

A.5 PRINCIPAL ANGLES ARE LIKELY NON-ZERO

In Theorems 5.2, 5.3, and 5.5, 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 5.2 assumed the largest principal angle between \( R_0 = \text{rowspace}(B_0) \) 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 4—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 \( B_0^\perp \) 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.17 shows that if the \( S \) is a randomly selected \( m \) dimensional subspace, then the angles \( \cos\theta_{\max}(R_*, S) \) and \( \cos\theta_{\max}(R_*, S^\perp) \) are non-zero (and we get quantitative lower bounds on them).

**Lemma A.17.** 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 \( 1 - \delta \),
\[ \cos\theta_{\max}(R, S) \geq \frac{\sqrt{m} - \sqrt{k} - \sqrt{2 \log \frac{1}{\delta}}}{\sqrt{d \log \frac{2d}{\delta}}} \]  \hspace{1cm} (A.184)

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

If \( m \geq 5k \) and \( m \geq 10 \log \frac{1}{\delta} \), then we get with probability at least \( 1 - \delta \):
\[ \cos\theta_{\max}(R, S) \geq O\left( \sqrt{\frac{m}{\log \frac{2d}{\delta}}} \right) \]  \hspace{1cm} (A.185)

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 $U \in \mathbb{R}^{d \times d}$ 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:

$$
\cos \theta_{\text{max}}(R,S) = \sigma_k(E^TF) = \sigma_k((UE)^T(UF))
$$

(A.186)

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 $c = \cos \theta_{\text{max}}(R,S)$. From Lemma 3.1, $c$ can be written as:

$$
c = \min_{r \in R, \|r\|_2 = 1} \| F^T r \|_2 = \min_{r \in R, \|r\|_2 = 1} \| F^T r \|_2
$$

(A.187)

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:

$$
\|r\|_2 = \| M_R \lambda \|_2 \leq \sigma_{\text{max}}(M_R) \| \lambda \|_2
$$

(A.188)

So this gives us:

$$
\| \lambda \|_2 \geq \frac{\|r\|_2}{\sigma_{\text{max}}(M_R)}
$$

(A.189)

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.187).

$$
c = \min_{r \in R, \|r\|_2 = 1} \| F^T r \|_2 = \min_{\| \lambda \|_2 \geq 1} \| F^T M_R \lambda \|_2
$$

(A.190)

$$
\geq \min_{\| \lambda \|_2 \geq 1/\sigma_{\text{max}}(M_R)} \| F^T M_R \lambda \|_2
$$

(A.191)

$$
\geq \min_{\| \lambda \|_2 \geq 1/\sigma_{\text{max}}(M_R)} \sigma_{\text{min}}(F^T M_R) \| \lambda \|_2
$$

(A.192)

$$
= \sigma_{\text{min}}(F^T M_R) / \sigma_{\text{max}}(M_R)
$$

(A.193)

So now we want to lower bound the ratio of two random matrices. We note that $F^T M_R$ is a matrix of size $(m,k)$ with each entry sampled independently from $N(0,1)$ (this is because $F^T$ 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 3.13, we can apply standard matrix concentration bounds (page 4, below Equation 1.11, in Rudelson & Vershynin (2009) for the bound on $\sigma_{\text{min}}$, and Theorem 4.1.1 in Tropp (2015) for the bound on $\sigma_{\text{max}}$). We get that with probability at least $1 - \delta$:

$$
\sigma_{\text{min}}(F^T M_R) \geq \sqrt{m - k} - \sqrt{2 \log \frac{1}{\delta}}
$$

(A.194)

$$
\sigma_{\text{max}}(M_R) \leq \sqrt{d \log \frac{2d}{\delta}}
$$

(A.195)

Note that we can use alternate bounds for $\sigma_{\text{min}}$ 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:

$$
c \geq \frac{\sigma_{\text{min}}(F^T M_R)}{\sigma_{\text{max}}(M_R)} \geq \frac{\sqrt{m - k} - \sqrt{2 \log \frac{2d}{\delta}}}{\sqrt{d \log \frac{2d}{\delta}}}
$$

(A.196)
For interpretability, ignoring log factors this is approximately:
\[ c \geq \frac{\sqrt{m} - \sqrt{k}}{\sqrt{d}} \]  
(A.197)

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

For the result where we show \( \cos \theta_{\max}(R, S) > 0 \) 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 \textit{Xie et al.} (2021b), we get that \( \sigma_{\min}(F^\top M_R) > 0 \) almost surely. Since \( \sigma_{\max}(M_R) \) is finite, this gives us \( \cos \theta_{\max}(R, S) > 0 \) almost surely.

\[ \square \]

In our case, the dimension of the ID subspace \( S \) is \( m \), and the dimension of \( R_\ast = \text{rowspace}(B_\ast) \) 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 \[ \text{A.17} \] tells us that \( \cos \theta_{\max}(R_\ast, S) > 0 \) and \( \cos \theta_{\max}(R_\ast, S^\perp) > 0 \) almost surely, and gives us quantitative lower bounds for these angles.

### A.6 LP vs. FT (ID)

We prove \text{Restatement of Proposition \[ \text{A.5} \]} where we show that if the representation is imperfect, then fine-tuning does better than linear probing, in-distribution.

**Restatement of Proposition **\[ \text{A.5} \]. In the linear overparameterized setting, under the ID subspace assumption (Assumption \[ \text{A.3} \]), let \( R_0 = \text{rowspace}(B_0) \), and \( R_{\text{aug}} = \text{Span}(\{w_\ast\} \cup \{R_0\}) \). Suppose \( w_\ast \notin R_0 \), \( \cos \theta_{\max}(S, R_{\text{aug}}) \neq 0 \), and that fine-tuning converges to a local minimum of its loss, then fine-tuning does better ID almost surely: \( L_{\text{id}}(v^\infty_{\text{ft}}, B^\infty_{\text{ft}}) < L_{\text{id}}(v^\infty_{\text{lp}}, B_0) \) 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, \( B_\ast, v_\ast \)), this means fine-tuning gets 0 loss on the training data as well. So for all training examples \( x \) (that is, rows of \( X \)):

\[ v^\infty_{\text{ft}} \top B^\infty_{\text{ft}} x = w_\ast \top x. \]  
(A.198)

Since the models are linear, this implies that fine-tuning gets all examples in the span of the training examples correct as well. Since \( P_x \) 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:

\[ L_{\text{id}}(v^\infty_{\text{ft}}, B^\infty_{\text{ft}}) = 0 \]  
(A.199)

**Linear probing gets positive ID loss:** Lemma \[ \text{A.20} \] shows that the ID error of linear probing is greater than zero under the same assumptions as this Proposition, so

\[ L_{\text{id}}(v^\infty_{\text{lp}}, B_0) > 0, \]  
(A.200)

which finishes the proof.

\[ \square \]

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

**Lemma \[ \text{A.18} \]** 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 \[ \text{A.18} \].** Let \( w \in \mathbb{R}^d \) be a vector and \( F \in \mathbb{R}^{d \times m}, E_0 \in \mathbb{R}^{d \times k}, E_{\text{aug}} \in \mathbb{R}^{d \times (k + 1)} \) have orthonormal columns, with \( \text{Range}(E_{\text{aug}}) = \text{Span}(\{w\} \cup \text{Range}(E_0)) \). If \( m > k \), we have:

\[ F^\top E_{\text{aug}} \text{ is full rank} \]  
(A.201)

\[ \overset{(\text{w})}{\Rightarrow} F^\top E_{\text{aug}} \text{ has higher rank than } F^\top E_0 \]  
(A.202)

\[ \overset{(\text{b})}{\Leftrightarrow} F^\top w \notin \text{Range}(F^\top E_0) \]  
(A.203)
The proof of (a) is clear—\(F^T E_{\text{aug}} \in \mathbb{R}^{m \times (k+1)}\) has rank \(k + 1\) (since it is full rank and \(m \geq k + 1\), but \(F^T E_{\text{aug}} \in \mathbb{R}^{m \times k}\) 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 \(F^T a_1, \ldots, F^T a_k, F^T w\) spans \(\text{Range}(F^T E_{\text{aug}})\), while \(F^T a_1, \ldots, F^T a_k\) spans \(\text{Range}(F^T E_0)\). So (notice the first list of vectors has an additional \(F^T w\)) this means that \(\dim(\text{Range}(F^T E_{\text{aug}})) \neq \dim(\text{Range}(F^T E_0))\) iff \(F^T w\) is linearly independent from the rest, that is, \(F^T w \notin \text{Range}(F^T E_0)\). Note that the rank of a matrix is the dimension of its range (column space), and is therefore lower rank. The assumption that \(m > k\) is crucial here.

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

**Lemma A.19.** In the linear overparameterized setting, under the ID subspace assumption, if \(F^T w_* \notin \text{Range}(F^T E_0)\), then \(L_{\text{id}}(v_{\text{lp}}^\infty, B_0) > 0\), 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 \(R_0 = \text{rowspace}(B_0)\) and ID subspace \(S\) respectively.

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

\[
L_{\text{id}}(v_{\text{lp}}^\infty, B_0) = \mathbb{E}_{x \sim P_d} [(v_{\text{lp}}^\infty B_0 x - v_{\text{lp}}^\infty B_0 x)^2] = 0 \quad (A.204)
\]

Since the squared error is always non-negative, this means that \(v_{\text{lp}}^\infty B_0 x = w_*^T x\) almost surely when \(x \sim P_d\) (recall that we defined \(w_* = B_1^T u_*\)). Recall \(P_d\) is defined as: first pick \(z \in P_2\) (which has density) and then output \(x = F z\). Since \(P_2\) has density, this implies that we get all examples in the ID subspace \(S\) correct:

\[
v_{\text{lp}}^\infty B_0 x = w_*^T x \quad \text{for all } x \in S. \quad (A.205)
\]

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

\[
v_{\text{lp}}^\infty B_0 F = w_*^T F. \quad (A.206)
\]

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 \(v^T E_0^T = v_{\text{lp}}^\infty B_0\). Then we have:

\[
v^T E_0^T F = w_*^T F \iff F^T E_0 v = F^T w_*
\]

\[
\iff F^T w_* \in \text{Range}(F^T E_0), \quad (A.208)
\]

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

Finally, Lemma A.20 combines Lemma A.18 and Lemma A.19 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 \(w_*\) and the feature rowspace \(R_0\), then linear probing has non-zero error. This is measured in terms of the principal angle \(\cos \theta_{\text{max}}(R_{\text{aug}}, S)\) between the ID subspace \(S\) and \(R_{\text{aug}}\) which is the span of \(R_0\) combined with \(w_*\). This angle will typically be non-zero—as an illustrative example, from Lemma A.17 we have that this angle will be non-zero almost surely if the ID subspace \(S\) is a uniformly random subspace.

**Lemma A.20.** In the linear overparameterized setting, under the ID subspace assumption, let \(R_0 = \text{rowspace}(B_0)\), and \(R_{\text{aug}} = \text{Span}(\{w_*\} \cup R_0)\). If \(w_* \notin R_0\) and \(\cos \theta_{\text{max}}(R_{\text{aug}}, S) > 0\), then \(L_{\text{id}}(v_{\text{lp}}^\infty, B_0) > 0\).

**Proof.** After a bit of setup, the proof simply combines Lemma A.18 and Lemma A.19. If \(w_* \notin R_0\), then \(R_{\text{aug}}\) has dimension \(k + 1\). Let \(E_{\text{aug}} \in \mathbb{R}^{d \times (k+1)}\), \(F \in \mathbb{R}^{d \times m}\) have orthonormal columns which form a basis for \(R_{\text{aug}}\) and \(S\) respectively. We assumed \(\cos \theta_{\text{max}}(R_{\text{aug}}, S) = \sigma_{\text{min}}(F^T E_{\text{aug}}) > 0\) which means that \(F^T E_{\text{aug}}\) is full rank. The ID subspace assumption assumes that \(m > k\). So from Lemma A.18, \(F^T w_* \notin \text{Range}(F^T E_0)\) where \(E_0 \in \mathbb{R}^{d \times k}\) has orthonormal columns that form a basis for \(R_0\). Then from Lemma A.19, \(L_{\text{id}}(v_{\text{lp}}^\infty, B_0) > 0\). \(\square\)
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.21.** In the overparameterized linear setting, let \( R = \text{rowspace}(B_0) \). If \( B_0 = B_* \), and \( \cos \theta_{\max}(S,R) > 0 \), then \( L_{ood}(v^\infty_{lp},B_0) = 0 \) for all \( t \).

**Proof.** We first show that because \( \cos \theta_{\max}(R,S) > 0 \), the training loss for linear probing is strongly convex. Recall that the loss is:
\[
\hat{L}(v,B) = \| XB^\top v - Y \|_2^2
\]
Linear probing keeps \( B \) fixed as \( B_0 = B_* \) and only tunes \( v \), so we are interested in the Hessian of the loss with respect to \( v \) evaluated at \( v,B_* \):
\[
\text{Hess}_v \hat{L}(v,B_*) = 2(B_* X^\top)(B_* X^\top)^\top
\]
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 \( \cos \theta_{\max}(R,S) \). For some \( F \) whose columns form an orthonormal basis for \( S \), we have (since the rows of \( B_* \) form an orthonormal basis for \( R \)):
\[
\sigma_k(B_* F) = \cos \theta_{\max}(R,S) > 0
\]
Note that \( B_* F \) is a \( k \)-by-\( \ell \) 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 \( B_* X^\top \) is rank \( k \). But that means the Hessian \( (B_* X^\top)(B_* X^\top)^\top \) 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:
\[
\hat{L}(v,B_*) = \| XB^\top v - X B_*^\top v_* \|_2^2
\]
So \( v = v_* \) achieves \( 0 \) loss and must be the (unique) minimizer. Therefore we have shown that linear probing converges to the unique minimizer \( v^\infty_{lp} = v_* \), which attains \( 0 \) loss, as desired.

Note that the entire proof works out if \( B_0 = U B_* \) for some rotation matrix \( U \). In that case, the Hessian becomes \( 2U(B_* X^\top)(B_* X^\top)^\top U^\top \) 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 \( v = U v_* \), since \( (U B_*^\top)(U v_*) = B_*^\top v_* \). So linear probing converges to \( v^\infty_{lp} = U v_* \), which achieves \( 0 \) loss, as desired.

**Restatement of Proposition 3.6.** Given perfect pretrained features \( B_0 = UB_* \) for some rotation \( U \). Let \( R_0 = \text{rowspace}(B_0) \). Under the non-degeneracy conditions \( \cos \theta_{\max}(R_0, S) \neq 0 \), \( \cos \theta_{\max}(R_0, S^\perp) \neq 0 \):
\[
\forall t, L_{ood}(B_t(t)^\top v_t(t)) > 0, \text{ if } v_0 \sim N(0, \sigma^2 I) \text{ is randomly initialized (FT),}
\]
\[
\forall t, L_{ood}(B_t(t)^\top v_t(t)) = 0, \text{ if } v_0 \text{ is initialized to } v^\infty_{lp} \text{ (LP-FT).}
\]

**Proof.** We first use Proposition A.21, which in the proof we showed still works if \( B_0 = U B_* \) for some rotation matrix \( U \) (which doesn’t have to be identity). We get that \( v^\infty_{lp} = U v_* \). Then we have \( B_0^\top v^\infty_{lp} = B_*^\top v_* = w_* \).

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

The training loss is:
\[
\hat{L}(v,B) = \| XB^\top v - X B_*^\top v_* \|_2^2
\]
The derivative with respect to \( v \) is:
\[
\partial_v \hat{L}(v,B) = 2B X^\top (X B^\top v - X B_*^\top v_*)
\]
Then since \( B_0^\top v^\infty_{lp} = B_*^\top v_* \), we have:
\[
\partial_v \hat{L}(v^\infty_{lp}, B_0) = 0
\]
Next, the derivative with respect to $B$ is:
\[
\frac{\partial_B \tilde{L}(v, B)}{v} = 2v (X B^T v - X B^T v_x) X
\]  
Then since $B_0^T v^\infty_p = B_0^T v_*$, we have:
\[
\frac{\partial_B \tilde{L}(v^\infty_p, B_0)}{v} = 0 \tag{A.219}
\]

So since both the derivatives are 0, we have $\partial_t v(t) = 0$ and $\partial_B B_\kappa(t) = 0$, which means the parameters don’t change at all—at all times $t$ we have $v_\kappa(t) = U v_*$ and $B_\kappa(t) = U B_*$ which gives us zero OOD loss: $L_{ood}(B_\kappa(t)^T v_\kappa(t)) = 0$ as desired. \hfill \square

## 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 Overview of Datasets

We first give an overview of the datasets used in our paper, before diving into more details of the exact training procedures (e.g., number of epochs, pretraining method, etc). 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 ([Sanurkar 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* ([Chinsue 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 → 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 → 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](#)).

### B.2 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.
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.

<table>
<thead>
<tr>
<th></th>
<th>Real</th>
<th>Painting</th>
<th>Clipart</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fine-tuning</td>
<td>55.29 (0.52)</td>
<td>50.26 (0.98)</td>
<td>60.93 (2.15)</td>
</tr>
<tr>
<td>Linear probing</td>
<td>87.16 (0.18)</td>
<td>74.50 (0.58)</td>
<td>77.29 (0.12)</td>
</tr>
<tr>
<td>LP-FT</td>
<td>86.82 (0.51)</td>
<td>75.91 (0.73)</td>
<td>79.48 (0.90)</td>
</tr>
</tbody>
</table>

• **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 5epoch 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 5epoch 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:
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 $\ell_2$ regularization weights, on OOD data. We see that linear probing still typically does better OOD (the only flip from before is on FMoW).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FT</th>
<th>LP</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR-10.1</td>
<td>92.27 (0.36)</td>
<td>82.67 (0.22)</td>
</tr>
<tr>
<td>STL</td>
<td>85.97 (0.38)</td>
<td>69.15 (0.13)</td>
</tr>
<tr>
<td>Ent-30</td>
<td>78.63 (0.53)</td>
<td>82.39 (0.14)</td>
</tr>
<tr>
<td>Liv-17</td>
<td>59.43 (2.49)</td>
<td>79.91 (0.24)</td>
</tr>
<tr>
<td>DomNet</td>
<td>40.23 (3.12)</td>
<td>37.12 (0.01)</td>
</tr>
<tr>
<td>FMoW</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FT</th>
<th>LP</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR-10</td>
<td>2.23 (0.03)</td>
<td>0.07 (0.00)</td>
</tr>
<tr>
<td>Entity-30</td>
<td>3.05 (0.02)</td>
<td>0.03 (0.01)</td>
</tr>
<tr>
<td>Living-17</td>
<td>1.88 (0.01)</td>
<td>0.11 (0.01)</td>
</tr>
<tr>
<td>DomainNet</td>
<td>207.6 (12.31)</td>
<td>0.19 (0.03)</td>
</tr>
<tr>
<td>FMoW</td>
<td>4.87 (0.15)</td>
<td>0.57 (0.19)</td>
</tr>
</tbody>
</table>

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 pretrained 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. 2020) 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.3 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.4 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.
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.

<table>
<thead>
<tr>
<th></th>
<th>STL</th>
<th>Entity-30</th>
<th>Living-17</th>
<th>DomainNet</th>
<th>FMoW</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT</td>
<td>1.70 (0.04)</td>
<td>2.60 (0.02)</td>
<td>1.67 (0.01)</td>
<td>159.97 (16.23)</td>
<td>5.62 (0.30)</td>
</tr>
<tr>
<td>LP-FT</td>
<td>0.04 (0.00)</td>
<td>0.02 (0.00)</td>
<td>0.09 (0.01)</td>
<td>0.18 (0.02)</td>
<td>0.54 (0.17)</td>
</tr>
</tbody>
</table>

Table 7: ID and OOD accuracies on Living-17 using a CLIP ResNet-50 model pretrained on the WebImageText 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.

<table>
<thead>
<tr>
<th></th>
<th>ID</th>
<th>OOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>LP</td>
<td>94.7 (0.2)</td>
<td><strong>78.6 (0.5)</strong></td>
</tr>
<tr>
<td>FT</td>
<td>94.7 (0.1)</td>
<td>67.3 (0.8)</td>
</tr>
<tr>
<td>LP-FT</td>
<td><strong>95.6 (0.2)</strong></td>
<td>77.0 (0.6)</td>
</tr>
</tbody>
</table>

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.

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× 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.5 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 3), 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 (‘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±0.4% OOD accuracy, and LP gets 88.3±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±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
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.

<table>
<thead>
<tr>
<th></th>
<th>ID</th>
<th>OOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>LP</td>
<td>97.5 (0.1)</td>
<td>87.6 (0.5)</td>
</tr>
<tr>
<td>FT</td>
<td>97.8 (0.0)</td>
<td>81.5 (2.1)</td>
</tr>
<tr>
<td>LP-FT</td>
<td>98.0 (0.0)</td>
<td>86.1 (0.1)</td>
</tr>
</tbody>
</table>

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 $\lambda$. 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.
Table 9: ID and OOD accuracies on Living-17 including three additional fine-tuning heuristics, where we (1) use a 10× 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.

<table>
<thead>
<tr>
<th></th>
<th>ID</th>
<th>OOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scratch</td>
<td>92.4 (1.3)</td>
<td>58.2 (2.4)</td>
</tr>
<tr>
<td>LP</td>
<td>96.5 (0.1)</td>
<td>82.2 (0.2)</td>
</tr>
<tr>
<td>FT</td>
<td>97.1 (0.1)</td>
<td>77.7 (0.7)</td>
</tr>
<tr>
<td>FT (10× Linear)</td>
<td>97.2 (0.2)</td>
<td>80.4 (0.3)</td>
</tr>
<tr>
<td>FT (regularized)</td>
<td>97.1 (0.2)</td>
<td>80.0 (0.4)</td>
</tr>
<tr>
<td>Side-tuning</td>
<td>95.5 (0.4)</td>
<td>81.0 (0.7)</td>
</tr>
<tr>
<td>LP-FT</td>
<td>97.8 (0.1)</td>
<td>82.6 (0.3)</td>
</tr>
</tbody>
</table>

B.6 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.

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

1. On CIFAR-10 → 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 → 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%).

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.
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 distribution 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.

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.