ANGULAR DENSITY REPRESENTS AN EMBEDDING'S TYPICALITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised pretraining has become the dominant paradigm for learning general-purpose representations across image, multi-modal, and language settings. Whether a given input is well-represented is normally assessed only indirectly, through labels, class structure, or expensive downstream evaluation. We show that an embedding's own geometry offers a cheap, label-free substitute: an embedding's *angular density* and, less consistently, its norm predict representation quality across paradigms. We argue that angular density measures an embedding's local *typicality* and does so even where no label- or class-based notion of typicality is available to begin with, e.g., individual LLM prompts or one side of a cross-modal image-text pair. Across vision, vision-language, and language models, embeddings with high angular density are more likely to be classified or answered correctly and, in vision and vision-language settings, represented similarly by other models. We additionally find that, in multi-modal models, the geometry of a *text* embedding predicts how reliably the corresponding *image* embedding is classified, despite no parameters being shared between the two encoders. We show that this local signal succeeds where a natural alternative, distance to a dataset centroid, does not: the global notion of typicality is inconsistent across models and often inverts, while angular density is monotonically predictive throughout.

1 INTRODUCTION

Pretraining and self-supervised learning have become the dominant paradigm for learning general-purpose representations at scale, across vision (Chen et al., 2020a; He et al., 2020; Dwibedi et al., 2021; Chen et al., 2020b; Fan et al., 2021; Grill et al., 2020; Chen & He, 2021; He et al., 2022; Assran et al., 2023; Caron et al., 2021; Oquab et al., 2023; Siméoni et al., 2025; Bardes et al., 2022; Zbontar et al., 2021; Balestriero & LeCun, 2025), vision-language (Radford et al., 2021; Zhai et al., 2023; Chen et al., 2024; Liu et al., 2023; Cherti et al., 2023; Yuan et al., 2021; Girdhar et al., 2023), and language settings (Olmo Team, 2025; Qwen Team, 2025b;a; GLM Team, 2024; Jiang et al., 2023; Gemma Team & Google DeepMind, 2025; 2024). Although methods within each paradigm differ widely in loss function, architecture, and training regime, they tend to produce representations that are consistent with one another and largely interchangeable for finetuning (Ericsson et al., 2021b; Goldblum et al., 2023). Correspondingly, the field evaluates these models at the *model-level*: classification accuracy for vision models (Chen et al., 2020a), benchmark suites for language ones (Clark et al., 2018; Chollet, 2019; Chollet et al., 2025; Hendrycks et al., 2021; Zellers et al., 2019; Sakaguchi et al., 2020; Bisk et al., 2020), and increasingly, cross-model representations alignment, formalized by the Platonic Representation Hypothesis (PRH) (Huh et al., 2024; Edamadaka et al., 2025; Tjandrasuwita et al., 2025; Ziyin & Chuang, 2025; Ruthardt et al., 2024; Jha et al., 2025). These measures characterize models across datasets rather than individual samples.

This model-level view leaves practitioners with no way to answer a question that matters at deployment time: *is this specific input* one the model is likely to handle well? Classification accuracy needs labels unavailable at inference time; cross-model alignment needs a second trusted model, running alongside every prediction; and class-conditional typicality needs class structure that many settings, like an LLM benchmark prompt, simply do not have. Without a per-input, label-free alternative, practitioners deploying pretrained encoders have no cheap way to flag which predictions to trust, which inputs to route to a stronger model or a human reviewer, or which examples in a dataset are poorly represented and worth relabeling or updating. This raises the question we study: *without*

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

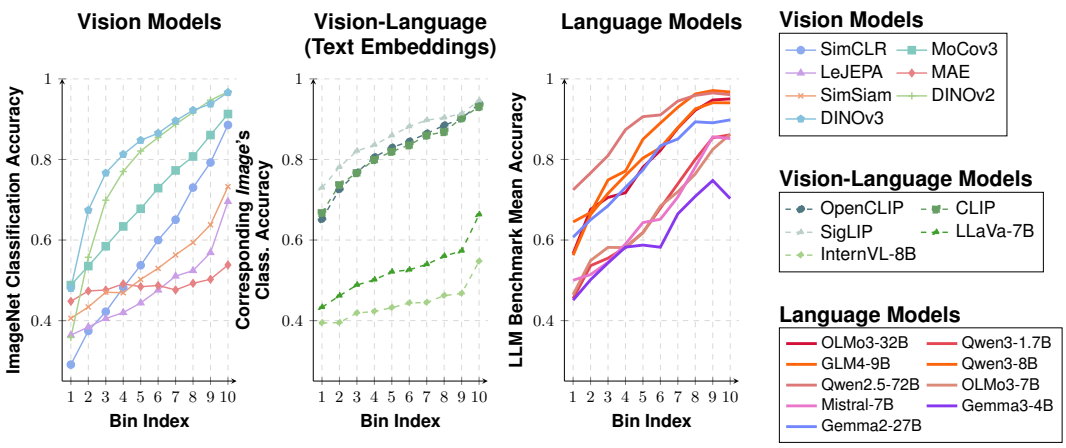


Figure 1: **Across pretrained models, an embedding's angular density predicts how reliably its input is handled.** For a large suite of pretrained vision (left), vision-language (middle) and language models (right), we sort embeddings by their angular density (mean similarity to k-nearest neighbors after normalization). We then split them into ten bins and evaluate the mean accuracy of each: k-NN classification for vision and vision-language, and mean benchmark accuracy for language. Across models and modalities, an embedding's local typicality, measured by its angular density, tracks downstream correctness.

labels, without a reference model, and without class structure, can an embedding's own geometry tell us whether the corresponding input is typical for the model, and thus likely to be well-represented?

Our Contributions. In this paper, we show that the answer is yes. Across seven SSL vision models, five vision-language models, and nine large language models, an embedding's *angular density*, i.e., its mean cosine similarity to its k nearest neighbors after normalization, behaves as a local, label-free measure for how typical the corresponding input is relative to the data distribution, and typical inputs are consistently better represented. Here, typicality is local and model-relative: an input is typical if its embedding lies in a region the encoder populates densely, unlike class prototypicality (Carlini et al., 2019) or distance to the distribution as a whole. Indeed, this measure remains computable and predictive in settings where no alternative notion of typicality (class- or label-based) is even defined.

To our knowledge, no prior work has established a diagnostic that (1) acts per-*input* to distinguish between inputs which are reliably vs. unreliably represented; (2) requires no labels and remains well-defined even when class-structure does not exist; (3) transfers across modalities; and (4) is specifically *local*, succeeding where a global notion of typicality fails.

To relate embedding properties to model quality quantitatively, we introduce the *Property-Quality Correlation* index (PQC), which scores a property highly when it *both* ranks embeddings by quality *and* separates high- from low-quality embeddings by a meaningful margin. This is unlike standard rank-correlation statistics, which can be high even when that separation is negligible.

We first study vision SSL models trained from scratch, where we can observe typicality emerge over training. Relating angular density to two measures of representation quality—k-NN classification accuracy (Chen et al., 2020a) and cross-model alignment (Huh et al., 2024)—we find that typical (high-density) embeddings are both more likely to be classified correctly and more likely to be represented consistently by other models. Embedding norm, which prior work links to quality under cosine-similarity losses (Draganov et al., 2025), carries a related but less stable signal: its direction reverses during training for several models.[1] The same holds for VLMs and LLMs (Figure 1): sorting embeddings by angular density into ten bins yields a near-monotonic increase in accuracy across vision SSL methods, VLM text embeddings, and LLMs. Notably, a VLM's *text* embedding predicts how reliably the corresponding *image* is classified, despite the two encoders sharing no parameters.

[1]We also study other embedding properties, but the angular density and norm show the strongest result.

A natural question raised by these results is whether "local typicality" is simply a more specific name for an already-known, coarser notion: whether a sample sits near the center of the overall data distribution. We show that it is not: a global, class-independent typicality baseline behaves inconsistently across models, sometimes even inverted, while angular density does not. We discuss this and related constructs, e.g., directional embedding structure and neural collapse, in Section 2.

We conclude by discussing our hypothesis for the mechanism behind these results and the concrete ways we believe this cheap diagnostic can be used in practice: flagging low-confidence predictions, routing inputs toward stronger models or human reviewers, and identifying poorly-represented samples reflecting bias or imbalance in datasets.

2 Related Work

The pretraining paradigms we study — vision, multi-modal, and language—share little at the level of training objectives or architecture (Chen et al., 2020a; He et al., 2020; Chen et al., 2020b; Fan et al., 2021; Dwibedi et al., 2021; Chen & He, 2021; Grill et al., 2020; He et al., 2022; Balestriero & LeCun, 2025; Caron et al., 2021; Oquab et al., 2023; Siméoni et al., 2025), yet each has produced its own literature on the properties of their learned representations. We summarize the relevant strands below; an extended discussion of paradigm-specific background is provided in Appendix A. We close by relating our framing to three related constructs: directional structure in embeddings, other notions of typicality, and neural collapse.

SSL Quality Metrics. SSL methods are typically evaluated using *downstream classification accuracy*: the model's weights are frozen, and a linear probe or k-NN classifier is trained on the embeddings (Marks et al., 2025). High classification scores suggest that the model developed a representation which naturally separates class boundaries. Building off previous ideas, research around the Platonic Representation Hypothesis (PRH) showed that networks trained with different objectives and modalities converge toward similar representations (Grigg et al., 2021; Bansal et al., 2021; Moschella et al., 2022), with the alignment growing between more powerful models (Edamadaka et al., 2025; Tjandrasuwita et al., 2025; Ziyin & Chuang, 2025; Ruthardt et al., 2024). This is measured using *overlap of nearest neighbors* between different models' embeddings, implying that high k-NN overlap with respect to a state-of-the-art model is a separate measure of model quality. Both metrics, however, are computed at the model level, over a full dataset.

Embedding Norms. An embedding's *norm* has been suggested to represent a contrastive SSL model's certainty in the corresponding input (Scott et al., 2021; Kirchhof et al., 2022; 2023). In particular, Draganov et al. (2025) showed that, under cosine similarity losses, embedding norms naturally grow during training and correlate with classification accuracy. To our knowledge, no prior work has studied this phenomenon on non-cosine-similarity loss functions or language models.

Angular Density. A line of work in *supervised* settings has shown that per-input geometric quantities such as local density and Mahalanobis distance to class centroids can flag inputs likely to be misclassified (Mandelbaum & Weinshall, 2017; Jiang et al., 2018; Sun et al., 2022; Papernot & McDaniel, 2018; Lee et al., 2018a). These measures, however, require supervision and class-labels. In the unsupervised setting, there is less evidence of embedding density as a predictor of whether the model understood an input. Wang & Isola (2020) theoretically showed that, under idealized conditions, normalized embeddings should distribute uniformly over the hypersphere while nonetheless clustering by classes, with subsequent work supporting this (Jing et al., 2021; Fang et al., 2024; Liu et al., 2018).

Vision-Language and Large-Language Models. Cross-modal contrastive learning aligns images and captions in a shared embedding space (Radford et al., 2021; Jia et al., 2021; Yuan et al., 2021; Girdhar et al., 2023), producing its own characteristic geometry: Liang et al. (2022) document a persistent "modality gap" between image and text embeddings, and Xu et al. (2023) examine how data composition shapes the learned space. Prior work has shown that text-derived signals can predict zero-shot classification accuracy at the dataset (Zohar et al., 2023) or per-class (Robbins et al., 2025) level, but whether geometric properties of *individual* text embeddings predict the reliability of the corresponding image classifications has, to our knowledge, not been studied. Separately, LLM activation geometry is known to encode qualities such as truthfulness (Azaria & Mitchell, 2023; Marks & Tegmark, 2024), and per-prompt hallucination likelihood (Kossen et al., 2024). Whether such geometric signals predict per-prompt correctness *within* a single model, as opposed to

correlating with which model produced the embedding in the first place, has, to our knowledge, not been established.

Angular Density in Context. Angular density touches on three related constructs, discussed in full in Appendix A. Directional structure in embedding spaces has been shown to be semantically meaningful enough to control outputs in diffusion models (Qiu et al., 2023); we instead ask whether local angular density diagnoses per-input reliability, with no notion of direction relative to a learned axis. Local density is known to correlate with prototypicality in supervised settings (Carlini et al., 2019). We argue that angular density is best understood as a label-free, *local* measure of typicality, and we show in Section 4 that this locality is essential: a global notion of typicality, distance to a single dataset-wide centroid, is inconsistent across models and often inverted. It is conceptually adjacent to *neural collapse*(Yang et al., 2023), though collapse is defined via class labels while angular density has no such conditioning; we return to this direction in Section 4 as well.

3 Property-Quality Correlation Methodology

Typicality in our local, model-relative sense has no ground-truth label, so we assess a candidate property through the representation quality it should predict if it tracks typicality. For an embedding property to serve as a useful measure of this kind, two things have to be true: sorting embeddings by the property has to track quality monotonically, *and* the quality difference between low-property and high-property embeddings has to be substantial. We summarize these two notions into a metric, which we term the Property-Quality Correlation (PQC); see Figure 2 for an illustration.

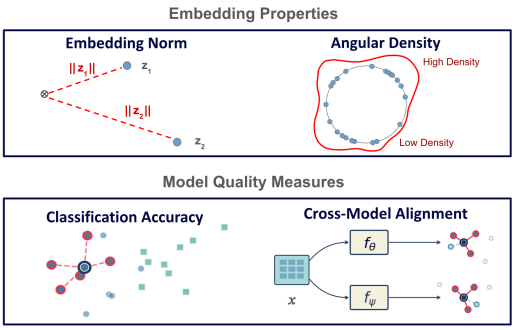


Figure 2: Visualization of our primary tools for comparing embedding properties to model quality in the SSL domain.

Notation. We denote a pretrained encoder $f_\theta : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{Z}$ is the space from which embeddings are taken for downstream applications. We index checkpoints by t and write the encoder as $f_\theta^{(t)}$. Following Huh et al. (2024), we use *embedding* to refer to a single vector-valued output of the frozen encoder; dense or token-level representations are not considered. For embeddings $\{z_i\}_{i=1}^N$ from a fixed model and dataset, we mean-center them as $\tilde{z}_i = z_i - \bar{z}$ and, for simplicity, write z_i for the centered embedding hereafter.

Embedding Properties. We focus on two embedding-level geometric properties p in the main body of the paper: the embedding's *norm* $p_{\text{norm}}(z_i) = \|z_i\|$ and *angular density* (p_{ang}). We measure angular density using the mean cosine similarity to the top-k nearest neighbors by cosine similarity:

$$p_{\text{ang}}(z_i) = \frac{1}{k} \sum_{j \in \text{kNN}(i)} \frac{z_i^\top z_j}{\|z_i\| \|z_j\|}.$$

In Appendix B, we also describe how we evaluate embeddings according to invariance to image augmentations and Mahalanobis distance to class centroids, both of which are vision-specific and serve as supporting analyses. Unless otherwise stated, all properties are computed on mean-centered embeddings, as defined above.

Model Quality Measures. We evaluate if an embedding property p correlates with a model-quality measure q. We consider two measures of representation quality that depend on the training paradigm.

Downstream accuracy. For vision SSL and VLM image encoders, we fit a k-NN classifier on ImageNet-1k training embeddings and evaluate on validation embeddings, with $k = 10$ (Lee et al., 2023). For language models, we average the model's accuracy across five benchmarks: MMLU, Hellaswag, Arc-AGI, PIQA, and Winogrande (Hendrycks et al., 2021; Zellers et al., 2019; Chollet, 2019; Chollet et al., 2025; Bisk et al., 2020; Sakaguchi et al., 2020). When computing LLM angular densities, nearest neighbors are drawn from the full set of embeddings for each corresponding evaluation dataset. More details for LLM accuracy are provided in Section C.6.

Cross-model alignment. We compute the k-nearest-neighbor (k-NN) overlap between two embedding spaces, following Huh et al. (2024): for models A and B, with embeddings z_i^A and z_i^B computed on shared inputs, we define $k\text{-NN-overlap}_k(A, B) = |S|^{-1} \sum_{i=1}^{|S|} |k\text{-NN}_k^A(i) \cap k\text{-NN}_k^B(i)|/k$ for a subset $S \subseteq \mathcal{Z}$, which ranges from 0 (indicating no neighborhood overlap) to 1 (indicating perfect overlap). We use $k = 10$ throughout. For each model A, we report a weighted mean of its overlap with the other models from the same training paradigm: each reference model M contributes with weight $\omega_M = \text{acc}(M)$, equaling the reference's final accuracy. The weighted overlap for model A is:

$$k\text{nn overlap}_{\text{weighted}}(A) = \frac{\sum_{M \neq A} \omega_M \cdot k\text{nn overlap}_k(A, M)}{\sum_{M \neq A} \omega_M}.$$

This emphasizes alignment with high-quality reference models, since alignment with lower-performing references is less informative as a quality signal.

Measurement Protocol. Measuring PQC involves four steps. First, we compute the norm p_{norm} and angular density p_{ang} on mean-centered embeddings z_i. For all subsequent steps, we additionally unit-normalize: $\hat{z}_i = z_i/\|z_i\|$. Second, we compute the paradigm-dependent quality measure or pairwise k-NN overlaps between models (for cross-model alignment). Since both quality measures rely on k-nearest neighbor structure over the same unit-normalized embeddings that define angular density, we separately validate our findings with linear-probe accuracy, which does not rely on neighborhood structure at all, and confirm the relationship holds (Section D.1). Third, we sort each model's validation embeddings by the chosen per-input property $p(z_i)$, and partition the embeddings into ten equally-sized bins $\{D_1, \ldots, D_{10}\}$, from lowest (D_1) to highest (D_{10}) property value. Finally, we compute the quality measure per bin: $\text{acc}(D) = \frac{1}{|D|} \sum_{z \in D} \mathbb{1}[h(z) = y(z)]$ for classification, where $y(z)$ is the true label of z, or $\text{acc}(D) = \frac{1}{|D|} \sum_{z \in D} \mathbb{1}[\text{correct}(z)]$ for LLM benchmarks, where $\text{correct}(z)$ indicates whether the model's answer matches the benchmark's reference answer. For cross-model overlap, since $D_b^A \neq D_b^B$ in general, we evaluate overlap on the union $U_b = D_b^A \cup D_b^B$.

The above procedure produces ten quality scores relating the property to the chosen paradigm-specific quality measure. To summarize this relationship as a single score, we combine two complementary terms: a magnitude term (Information) and a monotonicity term (Monotonicity).

Information. The *Information* term captures the magnitude of the quality gap between embeddings with high and low property values:

$$\text{Information}(p, q) = |q(D_{10}) - q(D_1)|, \tag{1}$$

where q is the quality measure. Larger values indicate that the property strongly discriminates between bin extremes.

Monotonicity. The *Monotonicity* term measures how consistently quality changes with the property. Let $\sigma = \text{sign}(q(D_{10}) - q(D_1))$ indicate the dominant direction of the relationship. We define

$$\text{Monotonicity}(p, q) = \frac{1}{10} \sum_{i=1}^{10} \mathbb{1}\left[\sigma \cdot q(D_i) = \max_{j \leq i} \sigma \cdot q(D_j)\right], \tag{2}$$

where 10 is the number of bins. This is the fraction of bins at which quality reaches a new running extremum in the direction set by σ: a running maximum if quality increases overall with the property, a running minimum if it decreases. The score lies in $[0, 1]$ in principle and approaches 1 for strictly monotonic relationships.

Property-Quality Correlation (PQC).

Definition 3.1 (Property-Quality Correlation)**.** The PQC of a property p with respect to a quality measure q is the product of the two terms:

$$\boxed{\text{PQC}(p, q) = \text{Information}(p, q) \times \text{Monotonicity}(p, q).}$$

Since both terms lie in $[0, 1]$ for the quality measures q we use here (accuracy and k-NN overlap, both bounded in $[0, 1]$), we have $\text{PQC} \in [0, 1]$. A score near 1 indicates that the property *both* discriminates strongly between high- and low-quality embeddings and predicts quality monotonically across the full property range. We choose this formulation over standard rank correlations, such as Spearman's ρ or AUROC, which discard the magnitude of the relationship; both are reported alongside PQC in Section D.5, where the property rankings are consistent with those produced by PQC.

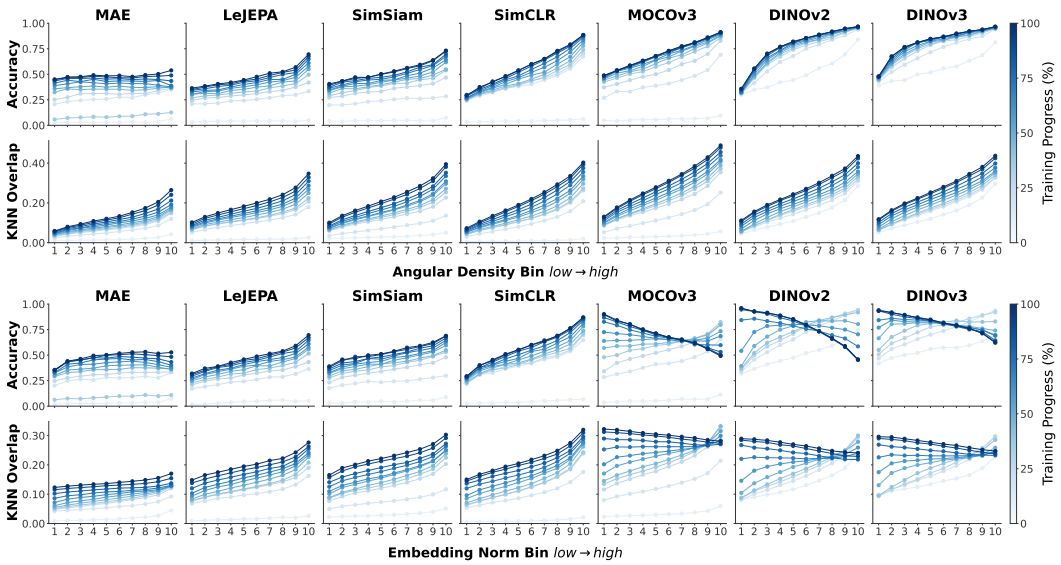


Figure 3: **Model-quality measures as a function of embedding property.** Each column corresponds to one SSL method. **Top:** sorted by angular density. **Bottom:** sorted by embedding norm. Curves show the property vs. quality measure at different training checkpoints (colored from light to dark as training progresses). Each point represents one bin D_i, plotted at its mean property (x-axis) against its median classification accuracy (top row) and weighted-average k-NN overlap (bottom row). Angular density's slopes are steeper and more consistently monotonic than the norm's.

4 RESULTS

Across every paradigm we study, an embedding's local angular density predicts whether that specific input is well represented, without reference to labels, class structure, or a second model. We interpret this as evidence that a representation's local geometry encodes typicality: dense regions correspond to inputs the model treats as typical, and this local notion of typicality, not a global one and not a distinct "quality axis", tracks downstream correctness and cross-model alignment. We establish this in three steps. First, in SSL models trained from scratch, we show the relationship is strong, holds under model accuracy and cross-model alignment, and is specifically local rather than a re-derivation of global typicality. Then, we show that it transfers to VLMs, including a cross-modal case where text-embedding geometry predicts image-embedding reliability. Finally, we show that it also holds in LLMs, where no label- or class-based notion of typicality is even defined.

Table 1: **PQC scores for the embedding norm and angular density with respect to accuracy.** Models sorted by k-NN accuracy (SSL; VLM) and mean benchmark score (LLM). VLM PQCs use text embeddings.

Model	Norm	Ang. Density
MAE (0.47)	0.14	0.06
LeJEPA (0.49)	0.38	0.33
SimSiam (0.56)	0.30	0.29
SimCLR (0.59)	0.58	0.59
MoCov3 (0.69)	0.41	0.42
DINOv2 (0.78)	0.50	0.61
DINOv3 (0.82)	0.31	0.49
Mean ($\pm$ std)	**0.37** ($\pm$ 0.13)	**0.40** ($\pm$ 0.18)
InternVL 8B (0.44)	0.01	0.15
LLaVA 7B (0.53)	0.00	0.23
CLIP ViT-L/14 (0.82)	0.05	0.26
OpenCLIP ViT-L/14 (0.82)	0.11	0.29
SIGLIP ViT-L/16 (0.86)	0.08	0.22
Mean ($\pm$ std)	**0.05** ($\pm$ 0.04)	**0.23** ($\pm$ 0.05)
Gemma3 4B (0.61)	0.31	0.19
Mistralv0.3 7B (0.66)	0.35	0.31
OLMo3 7B (0.67)	0.29	0.33
Qwen3 1.7B (0.67)	0.39	0.34
Gemma2 27B (0.78)	0.19	0.24
OLMo3 32B (0.80)	0.19	0.34
Qwen3 8B (0.80)	0.32	0.33
GLM4 9B (0.84)	0.21	0.27
Qwen2.5 72B (0.88)	0.14	0.20
Mean ($\pm$ std)	**0.26** ($\pm$ 0.08)	**0.28** ($\pm$ 0.06)

4.1 SSL RESULTS

We evaluate the following SSL models trained from scratch on ImageNet1K (Deng et al., 2009): MAE (He et al., 2022), LeJEPA (Balestriero & LeCun, 2025), SimSiam (Chen & He, 2021), SimCLR (Chen et al., 2020a), MoCov3 (Fan et al., 2021), DINOv2 and v3 (Oquab et al., 2023; Siméoni et al., 2025), letting us track whether these property-quality relationships *emerge* over training. Architectures, training procedures, k-NN and linear-probe accuracies are available in Appendix C.3.

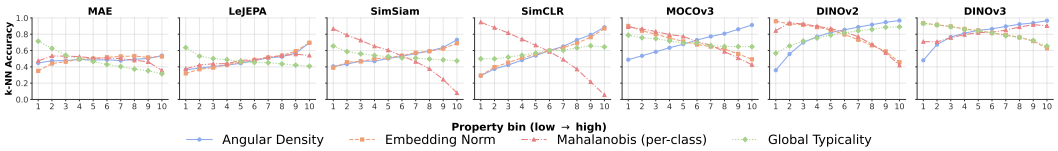


Figure 4: **k-NN classification accuracy as a function of property bin,** comparing angular density, embedding norm, per-class Mahalanobis distance, and the global typicality baseline, across all seven SSL models at final checkpoints.

Angular density predicts representation quality; embedding norm is predictive but unreliable. Figure 3 (top) shows that high-angular-density (i.e., typical) bins usually achieve higher classification accuracy and higher cross-model alignment: embeddings in regions of high angular density are more likely to be classified correctly and represented consistently by an independently trained model. The relationship sharpens over training (Appendix Figure 31), with meaningful neighborhood structure emerging progressively: typicality becomes visible in the geometry as the model learns. Embedding norm (Figure 3 bottom) also predicts representation quality across many SSL methods, but unstably: DINOv2, DINOv3, and MoCov3 all reverse trend halfway through training, and norm predicts cross-model alignment far less consistently than angular density, making it an unreliable signal in practice.

The relevant typicality is local, not global. A natural question is whether angular density is distinct from a broader, class-independent notion of data typicality, or merely a geometric proxy for it. We test this with a *global typicality* baseline: Mahalanobis distance to a single centroid over the entire reference set, using no labels or local structure. As Figure 35 shows, global typicality does not consistently track representation quality, and its direction of correlation is unstable across models. For five of the seven models (MAE, DINOv3, LeJEPA, MoCov3, SimSiam), accuracy *decreases* nearly monotonically as embeddings approach the global centroid - the opposite of what global typicality would predict. Only DINOv2 and SimCLR show the expected positive trend. Angular density, evaluated identically, is monotonically positive in all seven models. Typicality therefore predicts representation quality more consistently when it is measured locally.

Angular density matches a supervised baseline while using no labels. Table 6 further benchmarks against Mahalanobis distance to the class's centroid, an established, label-dependent measure of class typicality with the advantage of using ground truth assignment. Mean classification-accuracy PQC is nearly identical (0.39 ± 0.31 for true-class Mahalanobis vs. 0.40 ± 0.18 for angular density) despite one requiring labels and the other requiring none. On cross-model alignment, angular density is markedly *better* (0.30 vs. 0.12). Our unsupervised properties thus match established supervised ones for evaluating representations (additional results in Sections E.1 to E.3).

Angular density's reliability degrades only under reference-set scarcity, not class imbalance or contamination per se (Appendix D.8), and is not a fixed proxy for neural collapse (Appendix E.5). Table 1 summarizes PQC across all models, properties, and quality measures at the end of training, The two properties are only weakly correlated with each other (Appendix D), and angular density's PQC broadly increases with k-NN probing accuracy, while norm's does not.

4.2 VLM Embedding Geometries

We examine whether these geometric trends extend to vision-language models (VLMs), evaluating five representative models: CLIP ViT-L/14, OpenCLIP ViT-L/14, SigLIP ViT-L/16, InternVL 8B, and LLaVA 7B (Radford et al., 2021; Zhai et al., 2023; Chen et al., 2024; Liu et al., 2023; Cherti et al., 2023). VLMs admit a unique analysis with no unimodal analog: for each model, we compute the embedding properties of its *text* and *image* embeddings and relate both to the classification accuracy of the *corresponding image* embedding, directly probing the Platonic Representation Hypothesis (Huh et al., 2024) at the level of embedding geometry, i.e., whether two independently parametrized encoders develop shared structure that is informative about representation quality.

Angular density remains predictive, whereas embedding norm does not. VLMs preserve the main qualitative SSL trend observed but with a shift in relative strength: mean PQC for norm and

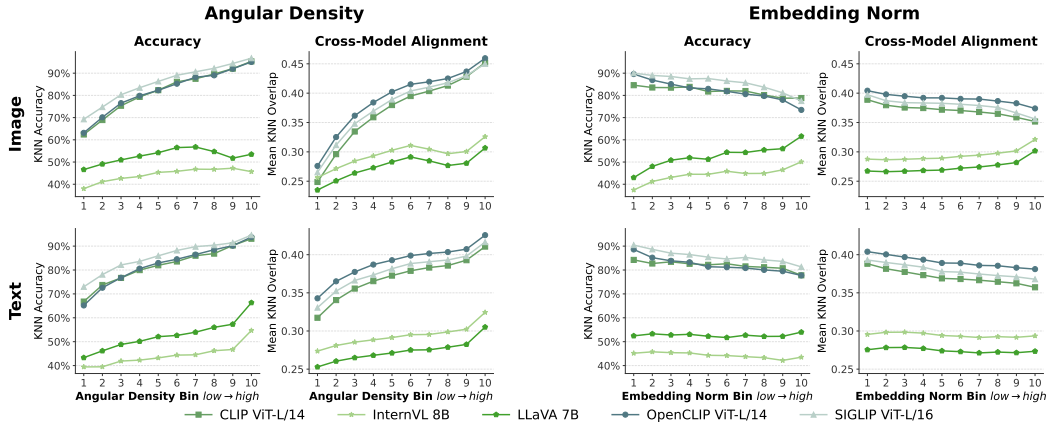


Figure 5: **Embedding properties vs. model quality. Two left columns:** angular density bins. **Two right columns:** embedding norm bins. Rows correspond to the data modality used to compute *properties*. Within each subplot, curves show the property–*image* accuracy curve for different VLMs.

angular density are comparable in SSL (0.37 vs. 0.40), but for VLM image embeddings, norm's mean PQC drops sharply to 0.11, while angular density remains 0.21 (Table 5).

Text-embedding typicality predicts image-embedding quality. We next ask whether text embeddings predict image-representation quality despite the encoders sharing no parameters. Figure 5 shows that binning embeddings by either image (top row) or text (bottom row) embedding properties yields similarly shaped accuracy curves, with angular-density bins showing clearer monotonic relationships than norm bins in both modalities. High-accuracy VLMs (CLIP, OpenCLIP, SigLIP) consistently occupy higher-quality regimes as angular density increases, suggesting that text embedding properties carry nearly as much predictive information about image quality as image embedding properties do.

4.3 LLM Embedding Geometries

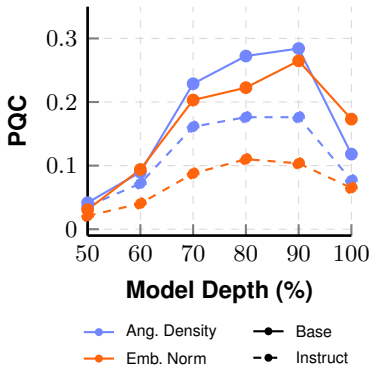
Finally, we relate embedding geometry to model quality in LLMs, using using nine models spanning providers, sizes, and architectures, each with pretrained and finetuned variants: OLMo3-7B, OLMo3-32B, GLM4-9B, Qwen2.5-72B, Qwen3-1.7B, Qwen3-8B, Mistralv0.3-7B, Gemma2-27B, and Gemma3-4B (Gemma Team & Google DeepMind, 2024; 2025; Qwen Team, 2025a;b; GLM Team, 2024; Olmo Team, 2025).This is the sharpest test of the local-typicality interpretation: LLM benchmark prompts have no natural class label and admit no global- or class-based typicality baseline, yet local angular density still requires only a reference set of embeddings.

LLM PQC is affected by depth, and base models show a cleaner signal than finetuned ones. Which embedding layer to use is not obvious (Skean et al., 2025). Figure 6a shows norm and angular-density PQC at different layers against mean benchmark accuracy[2]: both peak at 90%-depth and drop at the output layer, so we use 90%-depth embeddings throughout unless stated otherwise. The same figure shows base-model PQC exceeding finetuned-model PQC on average; Figure 6b confirms this per-benchmark, with both properties monotonically tracking accuracy and base models showing a clearer relationship than finetuned ones. Figure 1 (right) shows per-model accuracy curves by angular-density bin, averaged over the five benchmarks; per-model PQC scores are in Table 1.

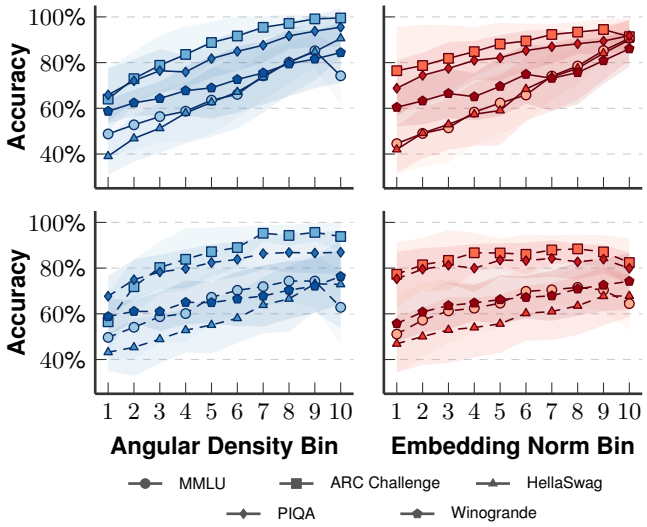
5 Hypothesis, Limitations and Implications

Hypothesis. Why would unrelated pretraining objectives turn typicality into angular density? We propose two coupled forces. First, pretraining objectives create natural attractors that pull frequent or easy inputs into dense regions; in effect, the objective converts how typical an input is into how densely its neighborhood is populated. Wang & Isola (2020) formalize one version of this for

[2]We include a version with standard deviations at Figure 33 in the appendix.



(a) **LLM PQC peaks at 90 % depth into the model.** For both of the embedding properties and LLM base/instruct model pairs, we plot the mean PQC across all 9 models and 5 benchmarks as a function of model depth. We see that pretrained models exhibit higher PQC scores than fine-tuned ones.



(b) **Embedding norms and angular densities encode LLM quality across benchmarks.** Mean and standard deviation per-bin accuracy across the 9 LLM models for both the base (top row) and instruct (bottom row) variants. Bins are sorted by embedding angular density (left column) and norm (right column).

Figure 6

contrastive losses, and HaoChen et al. (2021) sharpen the picture by showing that semantically-similar samples become tight clusters in the learned representation. The same coupling appears in language models, where frequent tokens occupy cones of the latent space while rare ones are pushed to the periphery (Gao et al., 2019; Cai et al., 2021). Second, the neural tangent kernel literature suggests that, because all embeddings share weights, a gradient step that affects one embedding necessarily affects its neighbors by a related amount (Jacot et al., 2018). Combined with Draganov et al. (2025)'s observation that gradient updates can directly affect embedding norms, dense embedding regions may accumulate aligned norm updates, producing the observed norm–quality relationship.

Limitations. Our per-bin PQC scores are moderate (0.2–0.6), which we view as the natural ceiling for an unsupervised, per-input signal. Indeed, angular density matches or exceeds the supervised Mahalanobis distance to true-class centroids on most models (Table 6), despite using neither labels nor class structure. The typicality we measure is relative to a specific encoder and a specific reference set: it describes how typical an input is *for this model, against this data*. Computing it therefore requires a reference set of embeddings; this is cheaper than labeling, but reliability depends on the set's coverage (Appendix D.8). Qualitatively, high-density embeddings are often prototypical class exemplars (Section E.7), consistent with prior findings that prototypical examples sit at the 'center' of a class (Wang & Isola, 2020; Mandelbaum & Weinshall, 2017).We argue this is important: an inference-time, unsupervised measurement recovers a signal previously exploited in supervised settings. Finally, LLM results use last-token activations on benchmark prompts, so the density signal likely depends on prompt distribution.

Implications. Even a noisy per-input quality signal has practical consequences: atypical (low-density) inputs can be flagged, routed to stronger models, or escalated to humans. For evaluation, it suggests benchmark scores can be decomposed by per-input difficulty using the model itself, with no auxiliary judge. For interpretability and safety, it adds to a growing body of evidence that the geometry of a model's representations carries behaviorally meaningful information (Azaria & Mitchell, 2023; Marks & Tegmark, 2024; Kossen et al., 2024). If angular density cheaply tracks reliability, as a notion of local typicality, across modalities, it becomes a scalable monitoring tool. The cross-paradigm consistency suggests pretraining induces shared geometric regularity in which local typicality predicts reliability; whether this regularity can be deliberately strengthened is future work.

AI USE STATEMENT

In this work, we used generative AI tools to help implement standard methods used in this work. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, assist in writing proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, assist with translation, clean or reformat datasets, support qualitative or thematic data analysis, or interpret results. Additionally, we used generative AI tools to assist in drafting sections of the paper, aid in understanding technical concepts, and help visualize results for submission. We have reviewed all AI-assisted work. AI-drafted text was rewritten and edited by the authors to ensure accuracy and reflect our own voice and understanding; AI-implemented code for standard methods was verified and tested for correctness by the authors; and AI-assisted visualizations were checked against the underlying data for accuracy. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have made efforts to ensure the reproducibility of our results. The source code for our models and experiments is provided in an anonymous repository (`https://anonymous.4open.science/r/SSL_Embeddings-2591/README.md`), including implementation details and instructions for running our experiments. Due to its size, the dataset used in our experiments cannot be included as supplementary material; however, we will organize data transfer for interested researchers upon request following publication.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *ICLR Workshop*, 2017.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. A general language assistant as a laboratory for alignment, 2021. URL `https://arxiv.org/abs/2112.00861`.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, Singapore, 2023. Association for Computational Linguistics.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL `https://arxiv.org/abs/2204.05862`.

Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. *Advances in neural information processing systems*, 34:225–236, 2021.

Adrien Bardes, Jean Ponce, and Yann Lecun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR 2022-International Conference on Learning Representations*, 2022.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*, 2021.

Nicholas Carlini, Ulfar Erlingsson, and Nicolas Papernot. Distribution density, tails, and outliers in machine learning: Metrics and applications. *arXiv preprint arXiv:1910.13427*, 2019.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45, 2024.

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

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2818–2829, 2023.

François Chollet. On the measure of intelligence, 2019. URL https://arxiv.org/abs/1911.01547.

François Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. ARC-AGI-2: A new challenge for frontier ai reasoning systems, 2025. URL https://arxiv.org/abs/2505.11831.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Rumen Dangovski, Li Jing, Charlotte Loh, Seungwook Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljačić. Equivariant contrastive learning. *arXiv preprint arXiv:2111.00899*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Andrew Draganov, Sharvaree Vadgama, Sebastian Damrich, Jan Niklas Böhm, Lucas Maes, Dmitry Kobak, and Erik Bekkers. On the importance of embedding norms in self-supervised learning. *arXiv preprint arXiv:2502.09252*, 2025.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9588–9597, 2021.

Sathya Edamadaka, Soojung Yang, Ju Li, and Rafael Gómez-Bombarelli. Universally converging representations of matter across scientific foundation models. *arXiv preprint arXiv:2512.03750*, 2025.

Linus Ericsson, Henry Gouk, and Timothy M Hospedales. Why do self-supervised models transfer? investigating the impact of invariance on downstream tasks. *arXiv preprint arXiv:2111.11398*, 2021a.

Linus Ericsson, Henry Gouk, and Timothy M Hospedales. How well do self-supervised models transfer? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5414–5423, 2021b.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6824–6835, 2021.

Alex Fang, Gabriel Ilharco, Mitchell Wortsman, Yu Wan, Vaishaal Shankar, Achal Dave, and Ludwig Schmidt. Data determines distributional robustness in contrastive language image pre-training (clip). In *International Conference on Machine Learning*, 2022. URL https://api.semanticscholar.org/CorpusID:248505704.

Xianghong Fang, Jian Li, Qiang Sun, and Benyou Wang. Rethinking the uniformity metric in self-supervised learning. *arXiv preprint arXiv:2403.00642*, 2024.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations (ICLR)*, 2019.

Gemma Team and Google DeepMind. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Gemma Team and Google DeepMind. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15180–15190, 2023.

GLM Team. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024. URL https://arxiv.org/abs/2406.12793.

Micah Goldblum, Hossein Souri, Renkun Ni, Manli Shu, Viraj Prabhu, Gowthami Somepalli, Prithvijit Chattopadhyay, Mark Ibrahim, Adrien Bardes, Judy Hoffman, et al. Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks. *Advances in Neural Information Processing Systems*, 36:29343–29371, 2023.

Tom George Grigg, Dan Busbridge, Jason Ramapuram, and Russ Webb. Do self-supervised and supervised methods learn similar visual representations? *arXiv preprint arXiv:2110.00528*, 2021.

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

Sharut Gupta, Chenyu Wang, Yifei Wang, Tommi Jaakkola, and Stefanie Jegelka. In-context symmetries: Self-supervised learning through contextual world models. *Advances in Neural Information Processing Systems*, 37:104250–104280, 2024.

Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011, 2021.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *NAACL-HLT*, 2019.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. *arXiv preprint arXiv:2405.07987*, 2024.

Kentaro Imajo, Masanori Hirano, Shuji Suzuki, and Hiroaki Mikami. A judge-free llm open-ended generation benchmark based on the distributional hypothesis. *arXiv preprint arXiv:2502.09316*, 2025.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Rishi Jha, Collin Zhang, Vitaly Shmatikov, and John X Morris. Harnessing the universal geometry of embeddings. *arXiv preprint arXiv:2505.12540*, 2025.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Heinrich Jiang, Been Kim, Melody Guan, and Maya Gupta. To trust or not to trust a classifier. *Advances in neural information processing systems*, 31, 2018.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*, 2021.

Michael Kirchhof, Karsten Roth, Zeynep Akata, and Enkelejda Kasneci. A non-isotropic probabilistic take on proxy-based deep metric learning. In *European Conference on Computer Vision*, pp. 435–454. Springer, 2022.

Michael Kirchhof, Bálint Mucsányi, Seong Joon Oh, and Dr Enkelejda Kasneci. Url: A representation learning benchmark for transferable uncertainty estimates. *Advances in Neural Information Processing Systems*, 36:13956–13980, 2023.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Henry Kvinge, Tegan Emerson, Grayson Jorgenson, Scott Vasquez, Tim Doster, and Jesse Lew. In what ways are deep neural networks invariant and how should we measure this? *Advances in Neural Information Processing Systems*, 35:32816–32829, 2022.

Jae-Hun Lee, Doyoung Yoon, ByeongMoon Ji, Kyungyul Kim, and Sangheum Hwang. Rethinking evaluation protocols of visual representations learned via self-supervised learning. *arXiv preprint arXiv:2304.03456*, 2023.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018a.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018b.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. Learning towards minimum hyperspherical energy. *Advances in neural information processing systems*, 31, 2018.

Amit Mandelbaum and Daphna Weinshall. Distance-based confidence score for neural network classifiers. *arXiv preprint arXiv:1709.09844*, 2017.

Markus Marks, Manuel Knott, Neehar Kondapaneni, Elijah Cole, Thijs Defraeye, Fernando Perez-Cruz, and Pietro Perona. A closer look at benchmarking self-supervised pre-training with image classification. *International Journal of Computer Vision*, pp. 1–13, 2025.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2024.

Luca Moschella, Valentino Maiorca, Marco Fumero, Antonio Norelli, Francesco Locatello, and Emanuele Rodolà. Relative representations enable zero-shot latent space communication. *arXiv preprint arXiv:2209.15430*, 2022.

Théo Moutakanni, Maxime Oquab, Marc Szafraniec, Maria Vakalopoulou, and Piotr Bojanowski. You don't need domain-specific data augmentations when scaling self-supervised learning. *Advances in Neural Information Processing Systems*, 37:116106–116125, 2024.

Olmo Team. Olmo 3, 2025. URL `https://arxiv.org/abs/2512.13961`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL `https://arxiv.org/abs/2203.02155`.

Nicolas Papernot and Patrick McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *Advances in Neural Information Processing Systems*, 33:3407–3418, 2020.

Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. *Advances in neural information processing systems*, 36:79320–79362, 2023.

Qwen Team. Qwen2.5 technical report, 2025a. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3 technical report, 2025b. URL https://arxiv.org/abs/2505.09388.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Kevin Robbins, Xiaotong Liu, Yu Wu, Le Sun, Grady McPeak, Abby Stylianou, and Robert Pless. Will it zero-shot?: Predicting zero-shot classification performance for arbitrary queries. In *2025 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 1513–1523. IEEE, 2025.

Jona Ruthardt, Gertjan J Burghouts, Serge Belongie, and Yuki M Asano. Better language models exhibit higher visual alignment. *arXiv preprint arXiv:2410.07173*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: an open large-scale dataset for training next generation image-text models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Tyler R Scott, Andrew C Gallagher, and Michael C Mozer. von mises-fisher loss: An exploration of embedding geometries for supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10612–10622, 2021.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Nikul Patel, Jalal Naghiyev, Yann Lecun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. In *International Conference on Machine Learning*, pp. 55854–55875. PMLR, 2025.

Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pp. 2443–2449, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380379. doi: 10.1145/3404835.3463257. URL https://doi.org/10.1145/3404835.3463257.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International conference on machine learning*, pp. 20827–20840. PMLR, 2022.

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

Megan Tjandrasuwita, Chanakya Ekbote, Liu Ziyin, and Paul Pu Liang. Understanding the emergence of multimodal representation alignment. *arXiv preprint arXiv:2502.16282*, 2025.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018. URL https://api.semanticscholar.org/CorpusID:49670925.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Yifei Wang, Kaiwen Hu, Sharut Gupta, Ziyu Ye, Yisen Wang, and Stefanie Jegelka. Understanding the role of equivariance in self-supervised learning. *Advances in Neural Information Processing Systems*, 37:127483–127510, 2024.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023.

Yongyi Yang, Jacob Steinhardt, and Wei Hu. Are neurons actually collapsed? on the fine-grained structure in neural representations. In *International Conference on Machine Learning*, pp. 39453–39487. PMLR, 2023.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL https://aclanthology.org/Q14-1006.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pp. 12310–12320. PMLR, 2021.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid Loss for Language Image Pre-Training . In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952, Los Alamitos, CA, USA, October 2023. IEEE Computer Society. doi: 10.1109/ICCV51070.2023.01100. URL https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.01100.

Liu Ziyin and Isaac Chuang. Proof of a perfect platonic representation hypothesis. *arXiv preprint arXiv:2507.01098*, 2025.

Orr Zohar, Kuan-Chieh Wang, Shih-Cheng Huang, and Serena Yeung. LOVM: Language-only vision model selection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Extended Related Work

This section expands on the background summarized in Section 2, covering paradigm-specific literature.

The SSL Landscape. The SSL landscape comprises several distinct training paradigms. Contrastive methods such as SimCLR and MoCo maximize agreement between augmented views while pushing apart negative examples (Chen et al., 2020a; He et al., 2020; Chen et al., 2020b; Fan et al., 2021; Dwibedi et al., 2021). Non-contrastive approaches such as SimSiam perform representation learning without explicit negatives (Chen & He, 2021; Grill et al., 2020). Reconstruction methods like MAE and LeJEPA (He et al., 2022; Balestriero & LeCun, 2025) learn by predicting masked content in pixel or latent space. Finally, self-distillation methods such as the DINO models (Caron et al., 2021; Oquab et al., 2023; Siméoni et al., 2025) match student predictions to teacher outputs. Although these methods differ widely in loss function, architecture, and training regime, they can largely be used interchangeably for fine-tuning purposes (Ericsson et al., 2021b; Goldblum et al., 2023).

Language Model Representations. A separate body of work studies the internal representations of autoregressive language models. In this paper, we distinguish between *pretrained*—or base model—LLMs (those which have only done next-token prediction) and *finetuned*—or instruct—LLMs which have been adapted to be helpful assistants (Askell et al., 2021; Ouyang et al., 2022; Bai et al., 2022). Linear-probing analyses consisting of training small classifiers on top of frozen intermediate activations have shown that representational structure varies with depth: early layers encode surface-level information while more abstract features emerge in mid-to-late layers (Alain & Bengio, 2017; Hewitt & Manning, 2019; Belinkov, 2022; Zou et al., 2023). A common method to evaluate LLM quality is to use multiple-choice question accuracy, where the model's prediction is commonly read using its first-token log-probability over the answer-letter tokens (Brown et al., 2020).

A.1 Angular Density in Context

Angular density touches on three related constructs: directional embedding structures, typicality, and neural collapse, which we discuss in the following.

Directional structure in embedding spaces. Qiu et al. (2023) use orthogonal, angle-preserving transformations on a diffusion model's internal representations to steer image generation, showing that angular structure there is semantically meaningful enough to control outputs. Our question differs in scope and use: rather than asking what angular structure *encodes* or how it can be *manipulated* within a single model, we ask whether an embedding's local angular density, with no notion of direction relative to a learned axis, can *diagnose* whether a specific input was represented reliably.

Typicality. Because local density is known to correlate with prototypicality in supervised settings (Carlini et al., 2019), a natural concern is that angular density simply recovers *typicality*, how representative a sample is of the data distribution, rather than a distinct notion of representation quality. We resolve this empirically in Section 4 via a global-typicality baseline (Mahalanobis distance to a single dataset-wide centroid, using no class labels and no local neighborhood structure): global typicality does not consistently track representation quality and its direction of correlation is unstable across models, whereas angular density is monotonically predictive in every model tested.

Neural collapse. Angular density's within-model behavior is also conceptually adjacent to *neural collapse*(Yang et al., 2023), where within-class variance collapses relative to between-class separation over training. Neural collapse is defined via class labels; angular density has no such conditioning. We test this relationship directly in Section 4 by measuring neighbor purity (the fraction of an embedding's nearest neighbors sharing its ground-truth label) as a function of angular density bin: purity rises monotonically with density in every model, but only some models (DINOv2, DINOv3, MoCov3, SimCLR) approach neural-collapse-like purity ceilings in their densest bins, while others (MAE, SimSiam, LeJEPA) remain substantially class-mixed even at the extreme. Angular density is therefore not a fixed proxy for within-class concentration; whether it behaves like one depends on how class-discriminative the underlying representation already is.

Together, directional structure, typicality, and within-class concentration are each, on their own, known to relate to representation geometry. What remains open, and what we establish in this paper, is whether a single, per-input, label-free measure can serve as a practical proxy for representation reliability across paradigms sharing no architecture, objective, or modality.

B Embedding Property Definitions

Further results using invariance and Mahalanobis distance properties can be found in Appendix E.

B.1 Norm and Angular Density

Here, we display the full definitions for embedding norm, angular density, and augmentation invariance. Figure 7 illustrated the measurement protocol. We first pre-process the embeddings and then compute all properties and create bins based on the sorting criteria. Finally, we take the individual bins and compute their model quality, via k-NN accuracy, and cross-model alignment, via k-NN overlap.

Definition B.1 (Embedding Norm)**.** The *norm* of an embedding z_i is its Euclidean magnitude:

$$\boxed{p_{\text{norm}}(z_i) = \|z_i\|}$$

Definition B.2 (Angular Density)**.** Given a fixed set of embeddings $\{z_j\}_{j=1}^{N}$ from a model and dataset, the *angular density* of an embedding z_i is its mean cosine similarity to its k nearest neighbors within that set

$$\boxed{p_{\text{ang}}(z_i) = \frac{1}{k} \sum_{j \in \text{kNN}(i)} \frac{z_i \cdot z_j}{\|z_i\| \|z_j\|}}$$

Here $\text{kNN}(i)$ denotes the indices of the k embeddings most similar to z_i by cosine similarity.[3]

B.2 Augmentation Invariance

Related Work The SSL community has widely assumed that learning invariance to augmentations is critical for developing good representations (Bardes et al., 2022; Tian et al., 2020), with crops shown to be the most effective for representation quality (Chen et al., 2020a; Moutakanni et al., 2024). However, recent work challenges the necessity and universality of invariance (Ericsson et al., 2021a; Dangovski et al., 2021; Gupta et al., 2024; Wang et al., 2024). To our knowledge, only two works directly measure the augmentation invariance: Purushwalkam & Gupta (2020) showed that the performance gain from crop invariance is dataset-dependent, while Kvinge et al. (2022) measured rotation invariance, an augmentation that is not commonly used in SSL pipelines. Thus, while augmentation invariance has been central to SSL design, whether increased invariance actually leads to improved embeddings remains unclear.

For the purpose of this paper, we define augmentation invariance in the following way:

Definition B.3 (Augmentation Invariance)**.** Given an input image x, a model checkpoint t, and an augmentation operator $a_s : \mathcal{X} \to \mathcal{X}$, sampled from an augmentation family $\mathcal{A}$ and parameterized by strength $s \in [0, 1]$, where $s = 1$ denotes the identity transformation, we define an augmented embedding as $z_s = f_\theta^{(t)}(a_s(x))$, and the corresponding base embedding as $z = f_\theta^{(t)}(x)$. The specific-augmentation invariance at strength s is measured by the cosine similarity $\text{SI}(z, z_s) = (z \cdot z_s)/(\|z\|\|z_s\|)$. We then define the embedding's overall *augmentation invariance* as the area under the specific-invariance curve over augmentation strengths[4]:

$$\boxed{p_{\text{inv}}(z) = \int_0^1 \text{SI}(z_s)\, ds}. \tag{3}$$

The score lies in $[0, 1]$, where 1 indicates perfect invariance.

[3] We use $k = 10$ (excluding z_i itself). Lee et al. (2023) report comparable behavior for $k \in \{10, 20\}$, and we ablate this choice in Section D.

[4] In practice, we approximate the integral as a discrete average over $s \in \{0, 0.1, \ldots, 1.0\}$.

A direct augmentation-invariance score $p_{\text{inv}}(z)$ can be affected by the angular density of the local embedding neighborhood: embeddings in dense regions naturally have angularly close neighbors, including nearby augmented variants, and may therefore receive high invariance scores even when the effect is partly driven by density rather than true augmentation stability. To reduce this confounding effect, we use a density-normalized invariance score

$$\tilde{p}_{\text{inv}}(z) = p_{\text{inv}}(z)/\bar{p}^{(t)}_{\text{ang}},$$

where $\bar{p}^{(t)}_{\text{ang}} = N^{-1}\sum_{i=1}^{N} p_{\text{ang}}(z_i)$ is the mean angular density across the embedding set at checkpoint t. We use $\tilde{p}_{\text{inv}}(z)$ throughout.

B.3 Mahalanobis Distance to Class Centroids

A common alternative for measuring whether an embedding is well-represented by the model is its Mahalanobis distance to the class centroids (Lee et al., 2018b). Unlike norm and angular density, this property requires class labels to compute, as it relies on per-class statistics estimated from the training embeddings. We include it as a supervised baseline against which the unsupervised properties above can be compared.

We further give this baseline an additional advantage over our unsupervised properties: rather than taking the distance to the nearest centroid, we use the distance to the validation embedding's true-class centroid. This makes our reported Mahalanobis scores an optimistic upper bound on what a label-free deployment of the method could achieve.

Definition B.4 (Mahalanobis Distance to Class Centroids)**.** Given a labeled training set of embeddings $\{z_j^{\text{train}}\}_{j=1}^{N}$ partitioned into classes $\{C_c\}_{c=1}^{K}$, we compute per-class means $\mu_c = |C_c|^{-1}\sum_{j\in C_c} z_j^{\text{train}}$ and per-class covariances $\Sigma_c = |C_c|^{-1}\sum_{j\in C_c}(z_j^{\text{train}} - \mu_c)(z_j^{\text{train}} - \mu_c)^\top$. For a validation embedding z_i with true class y_i, the *Mahalanobis score* is its negative squared Mahalanobis distance to its true-class centroid:

$$\boxed{p_{\text{mahal}}(z_i) = -(z_i - \mu_{y_i})^\top \Sigma_{y_i}^{-1}(z_i - \mu_{y_i})}$$

Higher scores indicate that z_i lies closer to its true-class centroid in the whitened embedding space, and are interpreted as indicating that the embedding is more in-distribution for its class.

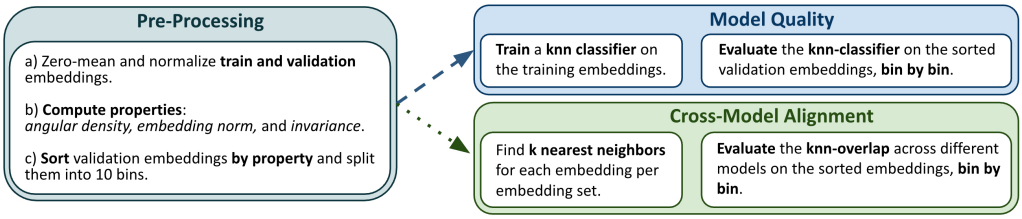


Figure 7: **Measurement protocol for the two model quality measures. Left:** Downstream accuracy via k-NN classification. **Right:** Cross-model alignment via k-NN overlap.

C Experimental Details

This section presents experimental details, covering implementation, data collection, and hardware.

C.1 Implementation & Hardware

Our implementation uses Python 3.12. The anonymized code repository is available at `https://anonymous.4open.science/r/SSL_Embeddings-2591/README.md`. The experiments operate on pre-computed embeddings extracted. Because the resulting embedding files are too large to redistribute, the code is not runnable end-to-end without access to those embeddings. We document the expected data format in the anonymous repository so the pipeline can be reproduced on user-supplied embeddings.

For k-NN probing, we use scikit-learn's KNeighborsClassifier (Pedregosa et al., 2011) with Euclidean distance, k=10, and brute-force search, following Caron et al. (2021); Marks et al. (2025). Ablations over k are reported in Appendix D.

To run the pipeline, we used a GPU server with 7x Nvidia H100 (6x 80 GB / 1x 94 GB: each having 26 / 34 TFLOPS fp64 peak performance, 29392 Gflops / 29031 Gflops according to gpu-burn) and 2x AMD EPYC 9354, 32 cores, 3.25 GHz (each having around 2.5 TFLOPS fp64 peak performance), with 1 TB DDR5 4800MHz RAM. Each slurm-experiment run received one GPU.

The pretraining and embedding generation was performed on a cluster, where one node includes 512 GiB and 4x Nvidia A100. The number of nodes used varies depending on the SSL method and the scale of the model size.

C.2 DATA (SSL)

We construct a dataset of SSL embeddings by training seven methods—SimCLR, MoCoV3, MAE, DINOv2, DINOv3, LeJEPA, and SimSiam—on ImageNet-1k, saving checkpoints throughout training. At each checkpoint t, we compute embeddings for the full ImageNet-1k training and validation sets, storing backbone features $f_\theta^{(t)}(x)$ and, when available, projector outputs $g_\phi^{(t)}(f_\theta^{(t)}(x))$. To analyze augmentation invariance, we additionally train models under weak, medium, and strong crop regimes, focusing on crop augmentation because it is among the most consequential augmentations for SSL (Chen et al., 2020a; Moutakanni et al., 2024). At each checkpoint, we apply crops with strengths $s \in [0, 1]$, where s denotes relative crop area: $s = 1$ corresponds to no crop and smaller values to stronger augmentation. Due to storage constraints, these augmentation-strength embeddings are computed on a validation subset rather than the full dataset. Throughout training, we monitor and report downstream classification accuracies to ensure models reach expected performance levels and to enable correlation analysis between representation properties and task performance.

Performance across training. We evaluate the k-NN probing performance on ImageNet-1K throughout the training process. Since the total number of training epochs varies across methods, we report the results at relative training progress milestones (10%, 20%, ..., 100%) for fair comparison. As shown in Table 2, the k-NN performance consistently improves as training progresses, and the final reproduced results generally align with those reported in the corresponding published works.

We further investigate the linear probing performance as shown in Table 3. Except for LeJEPA, whose ImageNet pretraining recipe is not fully disclosed, our reproduced results are generally consistent with the published results. For LeJEPA, we obtain reasonable performance, although it remains noticeably lower than the reported result in the original work.

Table 2: Top-1 k-NN probing Accuracy (%) vs Training Progress (%).

Method	10	20	30	40	50	60	70	80	90	100
MAE	3.49	22.90	29.82	36.52	38.87	41.83	43.75	45.55	46.38	46.98
LeJEPA	3.74	24.51	32.21	35.64	38.95	41.05	43.47	45.66	47.29	48.75
SimSiam	26.80	38.42	42.79	44.83	47.34	48.96	51.33	52.90	55.01	55.65
SimCLR	44.42	45.90	46.59	48.84	51.20	53.47	55.61	57.79	59.10	59.33
MoCo v3	43.41	55.89	62.11	64.15	65.98	67.44	68.51	69.08	69.23	69.22
DINOv2	52.83	68.78	72.04	73.08	74.18	75.23	76.32	77.38	78.16	78.25
DINOv3	58.10	72.95	76.67	78.11	78.72	79.90	80.88	81.73	82.24	82.21

Table 3: Linear Probing Accuracy (%) on ImageNet-1K. **Published** results are reported from the original papers and official repositories. **Reproduced** results are obtained using the same pretraining recipe and evaluation configuration as described in the corresponding published works.

Method	MAE	LeJEPA	SimSiam	SimCLR	MoCov3	DINOv2	DINOv3
Reproduced (%)	74.3	66.4	68.0	69.0	76.6	82.4	83.2
Published (%)	75.8	72.3	68.1	69.3	76.7	82.9	83.5

C.3 SSL Model Details

All models are pretrained on the ImageNet-1K training split. Model evaluation follows the standard SSL protocol: evaluation on the ImageNet-1K validation split using a k-NN classifier and linear probing. Ideally, each pretrained model should match the performance reported in the published results. Table 3 shows the reproduced and the published results.

SimCLR For SimCLR, we pretrain ResNet-50 on ImageNet-1K for 800 epochs using the exact configuration from SimCLR with NT-Xent, the normalized temperature-scaled cross-entropy loss. The k-NN accuracy of the final checkpoint is 59.3%, and the linear probing accuracy on the ImageNet validation dataset is 69.0%.

MoCo V3 We choose ViT-Base with patch size of 16 as the backbone of the MoCo V3 pretraining for 300 epochs which aligns with the original configuration as MoCo v3. MoCo v3 implements the InfoNCE van den Oord et al. (2018) loss. With this settings, the k-NN accuracy of the final checkpoint is 69.2%, and the linear probing accuracy is 76.6%.

SimSiam The backbone of SimSiam is a ResNet-50. We pretrain the model using the exact hyperparameters from the official SimSiam repository for 100 epochs. For evaluation, linear probing on ImageNet achieves 68.0% accuracy, while k-NN accuracy is 55.7%. SimSiam is trained using a negative cosine similarity loss between two augmented views. Specifically, given two views with projected features z_1, z_2 and corresponding predictions p_1, p_2, the loss is defined as:

$$\mathcal{L}_{\text{SimSiam}} = -\frac{1}{2}\left(\frac{p_1 \cdot \text{stopgrad}(z_2)}{\|p_1\|_2\|z_2\|_2} + \frac{p_2 \cdot \text{stopgrad}(z_1)}{\|p_2\|_2\|z_1\|_2}\right), \tag{4}$$

where $\text{stopgrad}(\cdot)$ denotes the stop-gradient operation.

DINOv2 For DINOv2, we pretrained a ViT-L/16 model following the fast-setup configuration described in the official DINOv2 pretraining protocol for 100 epochs. The overall loss function combines an image-level objective (L_{DINO}), a patch-level objective (L_{MIM}), and the KoLeo regularizer, which encourages diversity in the learned representations. Both L_{DINO} and L_{MIM} are formulated as cross-entropy losses.

The linear probing and k-NN evaluation results are consistent with those reported in the official repository, achieving 82.4% and 78.3%, respectively.

DINOv3 Based on DINOv2, DINOv3 introduces an additional regularization term—*Gram anchoring*—to improve the learning of dense features. This objective operates on the Gram matrix, defined as the matrix of all pairwise dot products between patch-level features within an image. Rather than constraining features directly, Gram anchoring enforces consistency in their similarity structure by aligning the student Gram matrix with that of a reference *Gram teacher*, typically an earlier snapshot of the teacher network. The loss is defined as:

$$\mathcal{L}_{\text{Gram}} = \left\|X_S X_S^\top - X_G X_G^\top\right\|_F^2, \tag{5}$$

where X_S and X_G denote the ℓ_2-normalized patch feature matrices of the student and Gram teacher, respectively.

We follow the exact training recipe of Siméoni et al. (2025) to pretrain a ViT-Large model for 100 epochs (12,500 iterations). For evaluation, k-NN probing achieves 82.14% accuracy, while linear probing reaches 83.2%.

MAE We pretrain a ViT-Large model for 800 epochs using the recommended default settings as He et al. (2022), keeping all configurations identical to the standard setup. Embeddings are extracted from the pretrained model using only the [CLS] token representations. The model is trained with a masked autoencoder (MAE) objective, minimizing the mean squared error (MSE) between the reconstructed and original pixel values of the masked patches. For evaluation, k-NN probing achieves an accuracy of 47.0%, while linear probing reaches 74.3%.

LeJEPA The default training configuration for LeJEPA is not publicly specified. We therefore reproduce results on ImageNet using a ViT-Large backbone (patch size 14, embedding dimension 1024) with a 3-layer MLP projector (hidden dimension 2048, output dimension 128). The model is trained with AdamW, using a learning rate scaled with the global batch size, a weight decay of 5×10^{-2}, and a schedule consisting of linear warmup followed by cosine decay. We adopt a multi-crop augmentation strategy with V global views (224×224) and 6 local views (98×98).

The LeJEPA objective combines an invariance loss with a spectral regularization term (SIGReg):

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{\text{inv}} + \lambda\mathcal{L}_{\text{SIGReg}}. \tag{6}$$

The invariance loss aligns representations across multiple augmented views by minimizing the squared distance between the mean representation of global views and that of all views. The SIGReg term regularizes the feature distribution by projecting features onto random directions and matching their trigonometric moments over a fixed grid $t \in [0, 3]$ to a Gaussian kernel $\phi(t) = \exp(-t^2/2)$. In practice, we use 17 discretization points and 256 random projections for SIGReg. The reproduced linear probing and k-NN results are 66.4% and 48.8% respectively.

C.4 VLM Model Details

In our VLM experiments, we evaluate performance using the ImageNet-Captions dataset Fang et al. (2022). This dataset augments 463,622 images from the 1.2 million images in the ImageNet 2012 training set with corresponding textual descriptions sourced from Flickr Young et al. (2014). For our evaluation pipeline, we randomly split the dataset into training and validation sets using an 80:20 ratio. All VLM experiments are conducted using this same ImageNet-Captions split to ensure consistency.

CLIP CLIP is pretrained using a symmetric contrastive loss (InfoNCE) to align image–text pairs, leveraging large-scale web-scraped datasets such as WebImageText (WIT) Srinivasan et al. (2021). Specifically, the model computes cosine similarities between all image–text pairs within a batch and applies a softmax-based contrastive loss over the resulting similarity matrix to maximize the similarity of matched pairs while minimizing that of mismatched pairs.

$$L_{\text{CLIP}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp\left(s(i,i)/\tau\right)}{\sum_{j=1}^{N}\exp\left(s(i,j)/\tau\right)} \tag{7}$$

where N denotes the batch size, $s(i, j)$ represents the cosine similarity between the i-th image and the j-th text embedding, and τ is a learnable temperature parameter that controls the sharpness of the similarity distribution. The numerator corresponds to the similarity score of the matched image–text pair, while the denominator aggregates similarity scores over all possible image-text pairs within the batch.

In our experiments, we adopt a pretrained CLIP model with a ViT-L/14 image encoder. The text encoder is a 12-layer Transformer with causal masked self-attention, hidden size 768, 12 attention heads, a context length of 77 tokens, and a BPE(Byte Pair Encoding)-based vocabulary.

OpenCLIP OpenCLIP follows the same contrastive training objective as CLIP, but is pretrained on larger open datasets such as LAION-2B Schuhmann et al. (2022).

SigLIP SigLIP replaces the softmax-based contrastive loss from CLIP with a sigmoid-based binary classification objective applied independently to image-text pairs.

$$L_{\text{SigLIP}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\log\frac{1}{1+\exp^{z_{ij}(-ts(i,j)+b)}} \tag{8}$$

where t is the learnable temperature parameter and b is the learnable bias term. z_{ij} denotes the label for the image–text pair, where $z_{ij} = 1$ for matched pairs and $z_{ij} = -1$ for mismatched pairs.

We adopted the pretrained SigLIP: the vision encoder is ViT-Large with the patch size of 16. The text encoder is a transformer with masked self-attention, typically using a hidden size of 768, 12 attention heads, and a context length of 64 tokens.

LLava LLaVA belongs to a different family of vision-language models compared to CLIP-based methods. Unlike CLIP, LLaVA relies heavily on the language model, while the image encoder is typically initialized from a pretrained CLIP model and kept frozen during training. The model is optimized using a standard causal language modeling objective based on autoregressive cross-entropy loss, where the next token is predicted conditioned on both the visual features and the previously generated text tokens.

We use LLaVA-v1.6-7B to generate the image and text embeddings for our experiments. The generated text embeddings have a dimensionality of 4096, while the image embeddings have a dimensionality of 1024, as the 7B model utilizes a pretrained CLIP ViT-L/14 image encoder.

InternVL InternVL adopts a hybrid training strategy, where image and text embeddings are aligned using a symmetric InfoNCE contrastive loss, while multimodal generation is optimized using an autoregressive cross-entropy loss similar to LLaVA. Unlike LLaVA, which typically keeps the image encoder frozen during training, InternVL scales the visual encoder and jointly pretrains it together with the language model.

In our experiments, we use the latest InternVL3.5-8B model in its pretrained form without additional fine-tuning. InternVL3.5-8B consists of approximately 8 billion parameters and employs the InternViT-300M visual encoder together with the InternLM2.5-7B language model. Both the image and text embeddings produced by the model have a dimensionality of 4096.

C.5 LLM Model Details

For LLM models, we choose both base and instruction-tuned variants. These models differ in parameter scale, architecture design, context length and attention mechanisms. We describe the detailed configurations of each model below.

OLMo OLMo3-7B and 32B are dense decoder-only Transformer models with a context length of 64K tokens. They employ YaRN (Yet another RoPE extensioN) for context scaling and adopt a hybrid attention mechanism combining sliding-window and full attention. The 32B variant further incorporates grouped-query attention (GQA) to improve efficiency at scale. The models are designed with both instruction-following and reasoning capabilities in mind, with dedicated variants optimized for long-horizon reasoning.

Qwen Qwen2.5-72B is a 72B dense decoder-only Transformer with a 128K context window, trained on approximately 18 trillion tokens. It is instruction-tuned for general-purpose language understanding, coding, and multilingual tasks.

Qwen3-1.7B and Qwen3-8B are smaller dense Transformer models with 32K context length. Unlike previous versions, Qwen3 introduces a hybrid thinking paradigm, allowing the model to operate in both standard instruction-following mode and explicit reasoning mode within a single architecture.

Gemma Gemma2-27B is a 27B dense decoder-only Transformer with 8K context, trained on around 13 trillion tokens. Gemma3-4B is a 4B multimodal Transformer model that supports 128K text context via RoPE scaling and integrates a SigLIP vision encoder for processing 896×896 images, represented as 256 visual tokens.

Mistral Mistral-7B is a 7B dense Transformer with 32K context, utilizing grouped-query attention (GQA) and sliding-window attention.

GLM GLM-4-9B is a 9B dense decoder-only Transformer with 32K context, employing multi-query attention (MQA) and partial rotary embeddings. This model exhibits excellent capabilities in mathematical reasoning and general tasks.

C.6 LLM Quality Measure

LLMs require us to adopt a different measure of model quality as LLM evaluations rarely have (Chang et al., 2024; Imajo et al., 2025). Typical LLM benchmarking consists of measuring whether

the model produces the correct answers to pre-specified prompts. Furthermore, there are known challenges with benchmarking pretrained LLMs which have not yet been tuned to interact as a (**?**Askell et al., 2021; Ouyang et al., 2022; Bai et al., 2022). We use the following methodology to measure an LLM's quality: we use 2K samples from the MMLU, Hellaswag, Arc-AGI, PIQA and Winogrande (Hendrycks et al., 2021; Zellers et al., 2019; Chollet, 2019; Chollet et al., 2025; Bisk et al., 2020; Sakaguchi et al., 2020) datasets and average the model's accuracy across these benchmarks. If the model is pretrained, we provide 5 random in-context examples with correct answers (Brown et al., 2020); for finetuned models, we simply provide the question directly. We evaluate the model's accuracy on each sample by giving it the prompt and evaluating which of the answer choices ('A', 'B', 'C', or 'D') has the highest log-likelihood on the immediate next output token (Brown et al., 2020). We record LLM 'embeddings' as the model activations obtained from the last token of the prompt. When evaluating LLM angular densities, we use the full set of embeddings for each corresponding evaluation dataset to evaluate nearest neighbors.

D Ablation Studies

Our setup exposes several hyperparameters. The angular density, a per-embedding quantity defined relative to its k nearest neighbors in the dataset, depends on the choice of k. Consequently, the resulting invariance also depends on k, since we normalize by the angular density to suppress correlation effects. For model accuracy, we train the k-NN classifier on normalized training embeddings and evaluate it on normalized validation embeddings. For cross-model alignment, we use validation embeddings throughout, ensuring the model has not seen them during training. The following subsections ablate each of these design decisions and quantify the contribution of the individual components.

D.1 Model Accuracy

The relationship does not rely on the neighborhood structure Both our quality measures, k-NN classification accuracy and cross-model k-NN overlap, are themselves computed via nearest-neighbor structure on the same unit-normalized embeddings that angular density is defined on. To rule out the possibility that the property-quality relationship we report is an artifact of this shared neighborhood structure rather than a genuine relationship, we additionally evaluate angular density against *linear-probing* accuracy, computed per property bin at the final training checkpoint across all seven SSL models. As shown in Figure 8, angular density predicts linear-probing accuracy with a clean, monotonic relationship in every one of the seven models, confirming that the density-quality relationship is not an artifact of shared neighborhood structure with our other quality measures. Embedding norm, evaluated under the same protocol, reproduces the same paradigm-dependent inconsistency reported before: accuracy increases with norm for MAE, LeJEPA, SimSiam, and SimCLR, but decreases with norm for MoCov3, DINOv2, and DINOv3.

The model-accuracy experiments use normalized embeddings throughout: the k-NN classifier is fitted on training embeddings and evaluated on validation embeddings. We set $k = 10$, following Lee et al. (2023), who show that $k \in \{10, 20\}$ yield comparable performance; prior work has also used $k = 20$ (Huh et al., 2024; Caron et al., 2021). Our ablations confirm this equivalence (Appendix D). The following sections ablate the effect of the embedding set (training vs. validation), normalization, and k.

Training vs validation embeddings To investigate whether k-NN probing depends on the choice of embedding set, we compare two scenarios:

- **Train** $\rightarrow$ **Val** (our default): the k-NN classifier is fitted on the training embeddings and evaluated on the validation embeddings.
- **Train** $\rightarrow$ **Train** (50/50 split): the k-NN classifier is fitted on 50% of the training embeddings and evaluated on the held-out 50%.

All other factors are held fixed: embeddings are normalized, $k = 10$, and binning by embedding properties is identical across scenarios. Figure 9 shows the mean k-NN accuracy at the final epoch, averaged across models and reported per bin. The two scenarios yield nearly identical curves,

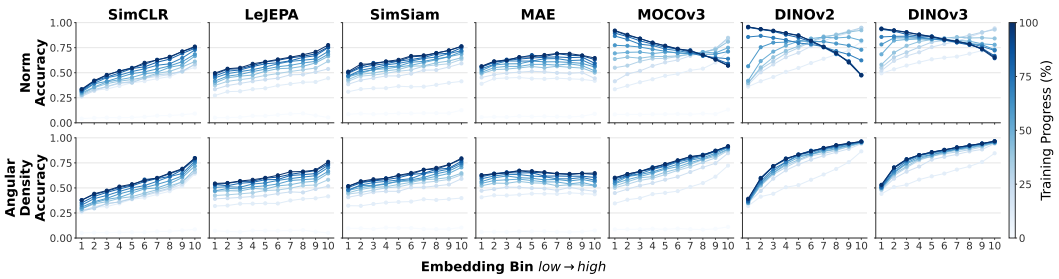


Figure 8: **Linear-probing accuracy as a function of embedding norm bin (top) and angular density bin (bottom)** across training checkpoints (light to dark with progress), for all seven SSL models.

indicating that our default protocol, i.e., evaluating on validation embeddings, does not introduce a systematic bias relative to the within-training-set alternative.

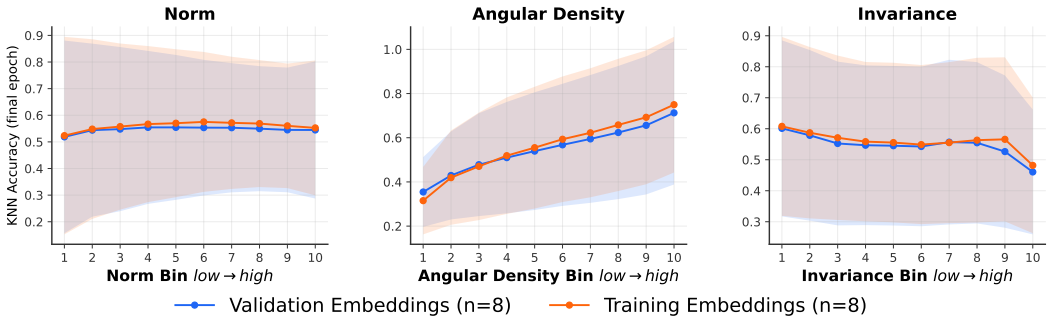


Figure 9: **Effect of embeddings used for the evaluation of the k-NN-classifier.** Mean values across all models for the final epoch.

Normalization To investigate whether k-NN probing depends on normalization of the embeddings, we compare two scenarios:

- **Normed embeddings** (our default): the k-NN classifier is fitted and evaluated on normed embeddings.
- **Unnormed embeddings**: the k-NN classifier is fitted and evaluated on unnormed embeddings.

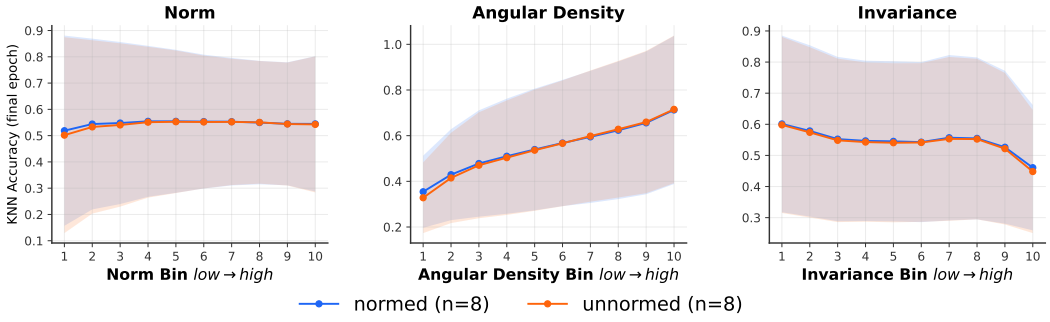


Figure 10: **Effect of embedding normalization on the evaluation of the k-NN-classifier.** Mean values across all models for the final epoch.

All other factors are held fixed: validation embeddings for evaluation, $k = 10$, and binning by embedding properties is identical across scenarios. Figure 10 shows the mean k-NN accuracy at the final epoch, averaged across models and reported per bin. The two scenarios yield nearly identical curves, indicating that our default protocol, i.e., using normed embeddings, does not introduce a systematic bias relative to the unnormed alternative.

Different values of k To assess the sensitivity of model accuracy to the choice of k, we repeat the k-NN probing protocol with $k \in \{5, 10, 20\}$, holding all other factors fixed (normalized embeddings, train $\rightarrow$ val split, identical binning). Figure 11 shows the mean k-NN accuracy at the final epoch, averaged across models and reported per bin. The three values yield nearly identical curves, consistent with the findings of Lee et al. (2023), and we adopt $k = 10$ as the default.

D.2 Cross-Model Alignment

This section presents ablations for the cross-model alignment experiments. In the main paper we report results on validation embeddings to avoid potential confounds from the model having seen the

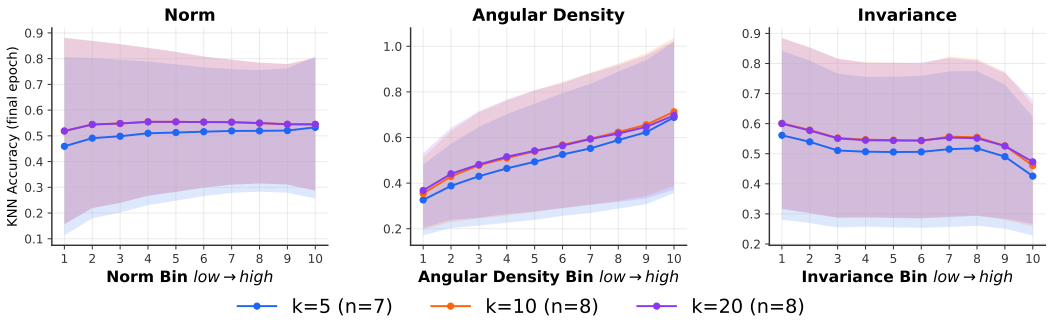


Figure 11: **Effect of different values of k, influencing the embedding properties.** Mean values across all models for the final epoch.

data during training. Here we additionally compare against training embeddings to verify that this choice does not affect our conclusions. We also ablate the number of neighbors used for k-NN-based alignment. Prior work reports that $k = 10$ and $k = 20$ yield very similar results (Lee et al., 2023; Caron et al., 2021; Marks et al., 2025), and we confirm this on our setup, additionally including $k = 5$.

Figure 12 shows the effect of $k \in \{5, 10, 20\}$ on validation embeddings. The three settings are nearly indistinguishable; the $k = 10$ curve is in fact occluded by $k = 20$ due to a near-perfect overlap, indicating that alignment is largely insensitive to k in this range.

Figure 13 repeats the experiment on training embeddings. Differences between k values are slightly larger here, but the qualitative behavior — including the relative ordering of methods and the trajectory over training — is preserved. We conclude that our findings are robust to the choice of k and to whether alignment is measured on training or validation embeddings.

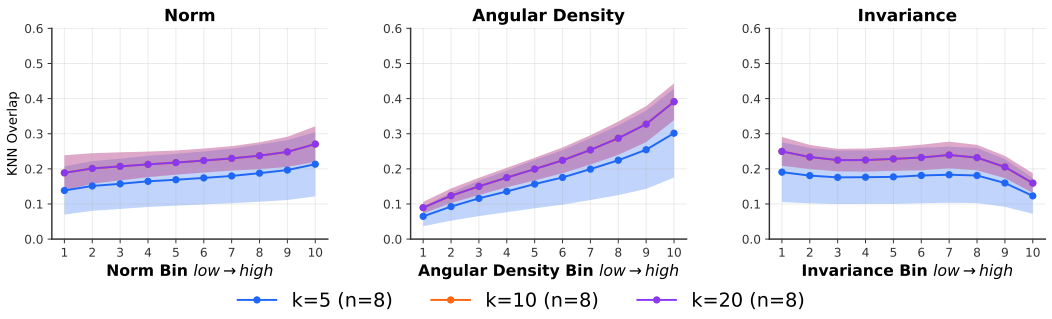


Figure 12: **Effect of k on k-NN Overlap for each bin sorted by embedding property averaged over all models.** Colors indicate the different values of k. The k-NN overlap scores are averaged across all models for validation embeddings from the final training step. Standard deviation is illustrated with lighter colored bands. Red ($k = 10$) seems to be missing on the first glance this is due to the perfect overlap with $k = 20$.

D.3 SimCLR with and without projector

To assess the effect of a projector, we include SimCLR with projector in our study. Figures 14 and 15 show how the PQC scores differ across the two SimCLR variant and quality criteria, i.e., model quality and cross-model alignment.

For model quality, the general relationship is very similar for embedding norm and angualr density. However, SimCLR without projector achieves higher metric scores across all three properties. This means that across all scenarios the effect the embedding properties capture information about the quality and cross-model alignment is stronger. In contrast to model quality, the results for cross-

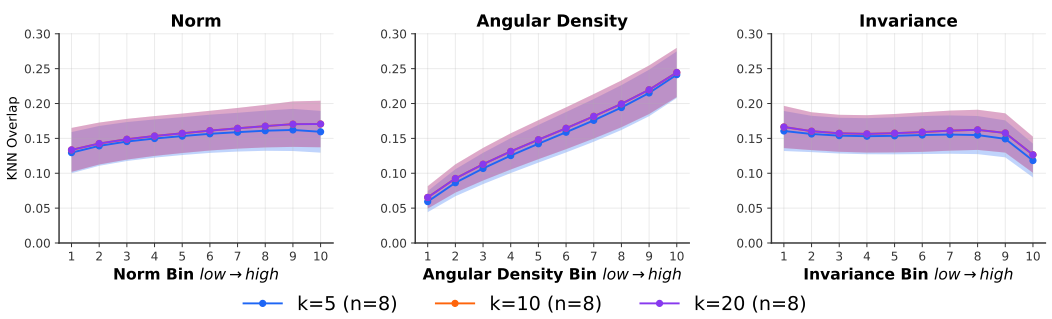


Figure 13: **Effect of k on k-NN Overlap for each bin sorted by embedding property averaged over all models.** This figure mirrors Figure 12, the difference between the two is that here we look at training embeddings instead of training embeddings.

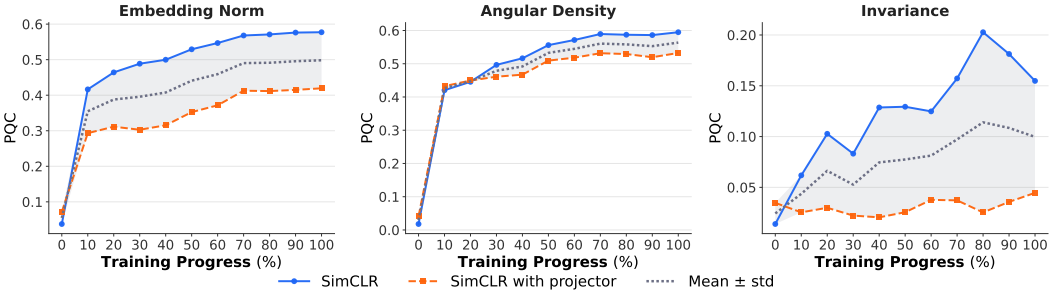


Figure 14: **Model quality PQC for SimCLR with vs without projector.** Colors indicate with and without projector.

model alignment are much clearer. Both SimCLR variants behave exactly the same across all three properties.

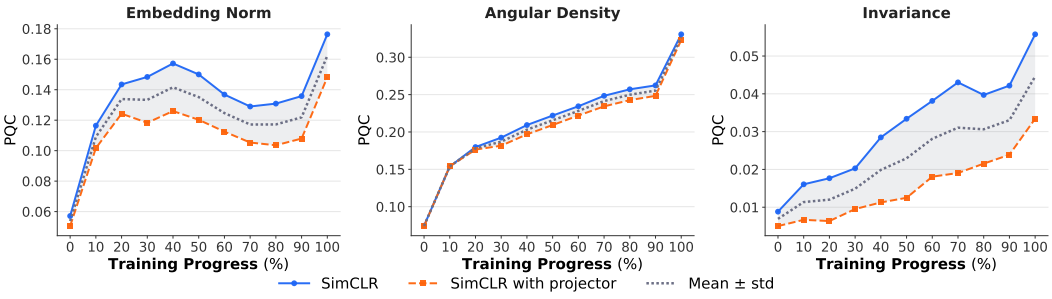


Figure 15: **Cross-model alignment PQC for SimCLR with vs without projector.** This figure mirrors Figure 14.

D.4 Correlations between embedding properties

Our main results show that embedding norm and angular density both predict representation quality, while invariance does not. This raises the question of whether norm and angular density capture distinct properties of the embedding or merely two views of the same underlying signal.

To answer this, we examine the overlap between property-based bins. For each pair of properties (p_A, p_B) and each model, we sort the validation embeddings independently by each property and partition into bins $D_1^A, \ldots, D_{10}^A$ and $D_1^B, \ldots, D_{10}^B$. We then compute a 10×10 confusion matrix C, where entry C_{ij} represents the proportion of samples in bin D_i^A that also fall in bin D_j^B. If two

properties are perfectly correlated, this matrix will be the identity; if they are independent, it will be uniform.

Figures 16, 17, and 18, show the resulting matrices at the final epoch for each model, for the three pairs (norm, angular density), (norm, invariance), and (angular density, invariance).

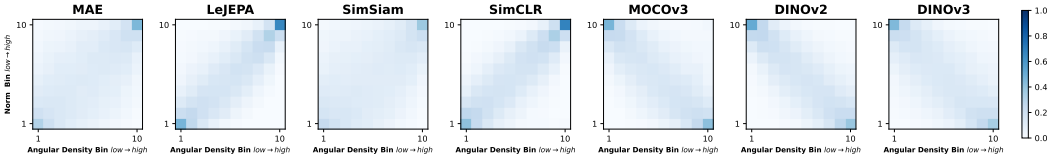


Figure 16: **Correlation between norm and angular density.** Each heatmap shows bin overlap: entry (i, j) is the proportion of samples in norm bin i that fall in angular density bin j. Diagonal structure indicates correlation.

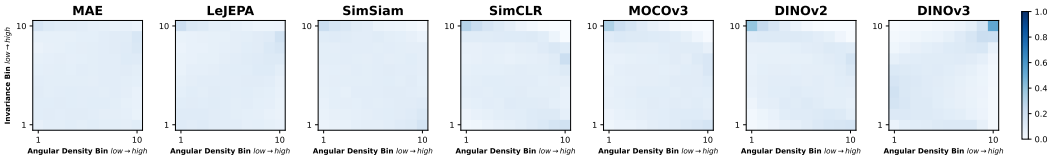


Figure 17: **Correlation between angular density and invariance.** Each heatmap shows bin overlap: entry (i, j) is the proportion of samples in norm bin i that fall in angular density bin j. Diagonal structure indicates correlation.

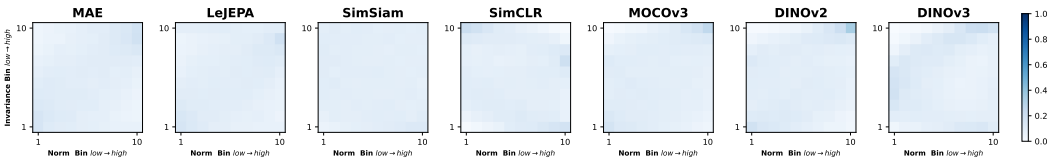


Figure 18: **Correlation between embedding norm and invariance.** Each heatmap shows bin overlap: entry (i, j) is the proportion of samples in norm bin i that fall in angular density bin j. Diagonal structure indicates correlation.

We further examine how these property correlations evolve over training. For each model, we compute the pairwise Spearman correlations between angular density, embedding norm, and invariance at every checkpoint, using ImageNet-1k validation embeddings. Figure 19 reports the resulting trajectories, with one panel per model.

D.5 Alternative Scorings

Alternative approaches including rank-based correlation might be, Spearman's ρ or AUROC, we include both as additional perspectives.

Figure 20 shows how the Spearman correlation between k-NN accuracy and the embedding properties evolve over training. Note that for these experiments we compare the sample wise scores, while PQC requires pre-sorted bins. We observe that across training the correlation coefficient for all properties, including Mahalanobis distance to the class mean, never exceed higher scores than 0.4 and lower than -0.5, suggesting weak to moderate correlation. However, the Spearman correlation is rank-based and thus only describes how well a relationship can be described by a monotonic function and the correctness can either be 0 or 1 for each sample.

In addition to Spearman, we also include Area Under the Receiver-Operating characteristic curve (AUROC) in this section. AUROC assesses how well an ordering regarding a property (here: embedding norm, angular density, invariance to augmentations, and mahalanobis distance to the class

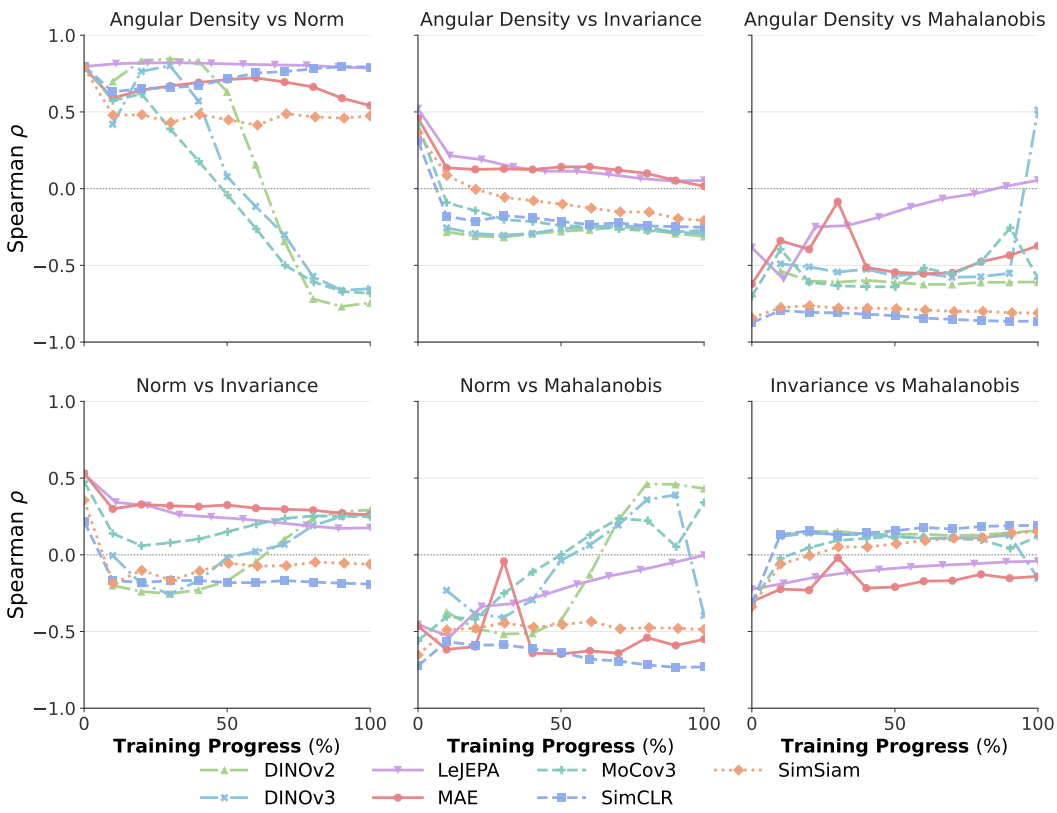


Figure 19: **Pairwise Spearman-Correlation between embedding properties across training.** Colors indicate models. Spearman correlation in relationship to the progress of training.

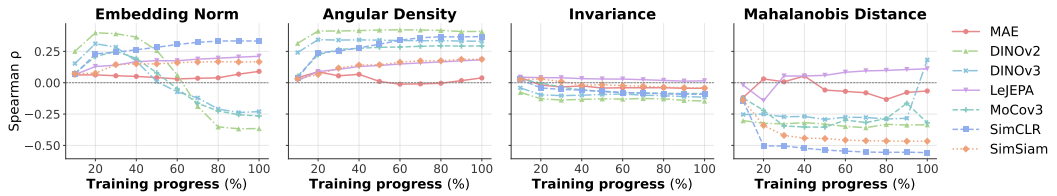


Figure 20: **Spearman correlation between properties and k-NN accuracy over training.**

mean) separate correctly-classified from the incorrectly-classified embeddings. Figure 21 shows the AUROC scores over the course of training. We observe, that angular density achieves the highest scores. While invariance is, with values around 0.5, essentially uninformative. AUROC proves helpful regarding classification accuracy, however it cannot be applied to assess cross-model alignment, since it requires a binary classification scenario, which is in our case just modeled as correctly / falsely classified.

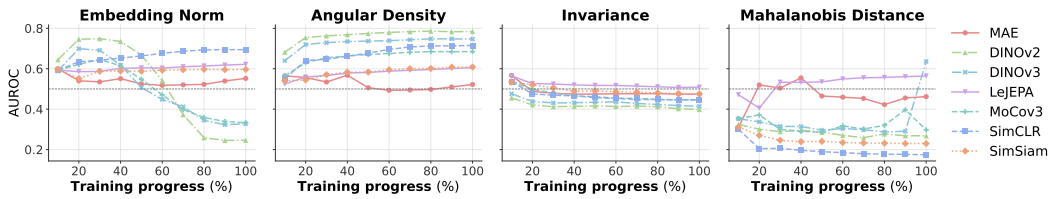


Figure 21: **AUROC regarding properties over training.**

Table 4: **PQC scores for the embedding norm and angular density with respect to accuracy (instruct variants).** Models sorted by mean benchmark score.

Model	Norm	Ang. Density
Qwen2.5 72B-It (0.86)	0.05	0.07
Gemma2 27B-It (0.79)	0.04	0.25
Qwen3 8B-It (0.76)	0.21	0.29
GLM4 9B-Chat (0.76)	0.13	0.23
OLMo3.1 32B-It (0.72)	0.07	0.09
Mistralv0.3 7B-It (0.67)	0.13	0.17
Gemma3 4B-It (0.63)	0.07	0.28
OLMo3 7B-It (0.60)	0.08	0.09
Qwen3 1.7B-It (0.60)	0.16	0.11
Mean ($\pm$ std)	**0.10** ($\pm$ 0.05)	**0.18** ($\pm$ 0.09)

D.6 LLM

Figure 22 extends the LLM results in the main paper. Here we additionally plot the k-NN accuracy across angular density bins for nine LLMs after pretraining. We observe that the behavior is fairly consistent and k-NN accuracy grows with growing angular density. We show the instruct model variants' PQC scores in Table 4.

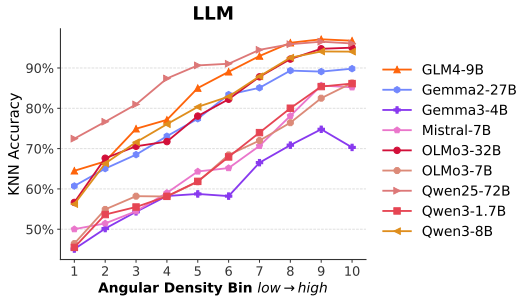


Figure 22: **k-NN Accuracy vs. angular density.** Colors indicate different LLMs.

Table 5: PQC scores for Classification Accuracy and Cross-Model Alignment across image and text modalities. Rows sorted by overall k-NN accuracy (shown in parentheses). Quality measures computed regarding *image* models.

	Classification Accuracy				**Cross-Model Alignment**			
	Image		**Text**		**Image**		**Text**	
Model	Norm	Ang. Density	Norm	Ang. Density	Norm	Ang. Density	Norm	Ang. Density
InternVL 8B (0.44)	0.09	0.06	0.01	0.15	0.03	0.05	0.00	0.05
LLaVA 7B (0.53)	0.15	0.05	0.00	0.23	0.03	0.05	0.00	0.05
CLIP ViT-L/14 (0.82)	0.03	0.33	0.05	0.26	0.04	0.20	0.03	0.09
OpenCLIP ViT-L/14 (0.82)	0.16	0.32	0.11	0.29	0.03	0.18	0.02	0.08
SIGLIP ViT-L/16 (0.86)	0.11	0.27	0.08	0.22	0.04	0.18	0.03	0.09
Mean	0.11 $\pm$ 0.05	0.21 $\pm$ 0.13	0.05 $\pm$ 0.04	0.23 $\pm$ 0.05	0.03 $\pm$ 0.01	0.13 $\pm$ 0.07	0.02 $\pm$ 0.01	0.07 $\pm$ 0.02

D.7 VLM

Table 5 provides additional results for VLMs. The quality measures, classification accuracy and cross-model alignment, always refer to quality evaluated based on image embeddings. The embedding properties norm and angular density are computed for both *text* and *image* embedding properties and used for the measurement protocol.

D.8 Reference-Set Composition

Angular density is computed relative to a reference set of embeddings. This section examines how the composition of that reference set, its class balance and presence of out-of-distribution samples, affects the reliability of the resulting property.

Class imbalance. We construct reference sets at four levels of class imbalance (imbalance factor ranging from approximately 2 to 1280, via exponential subsampling) across all seven SSL models, and evaluate PQC separately for overall, majority-class, and minority-class queries.

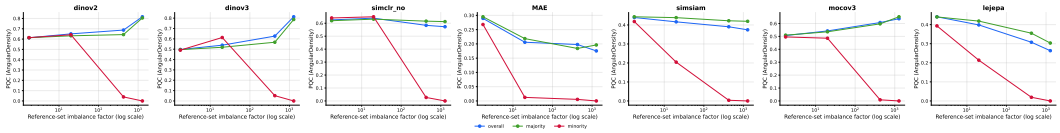


Figure 23: **PQC (angular density, classification accuracy) as a function of reference-set class imbalance,** shown separately for overall, majority-class, and minority-class queries.

Figure 23 and Figure 24 show that overall and majority-class reliability are essentially unaffected across every imbalance level tested. Minority-class reliability degrades in two distinct ways. At moderate imbalance, we observe architecture-dependent degradation in angular density's ordering ability: Monotonicity remains at 0.9-1.0 for DINOv2, DINOv3, SimCLR, and MoCov3 even at approximately 20×imbalance, while MAE and SimSiam already drop to 0.5-0.9 at the same level. At severe imbalance, minority-class accuracy collapses to exactly 0% in every density bin for every model, reflecting a degenerate classification task, too few minority reference exemplars to classify correctly at all, rather than a failure of angular density specifically.

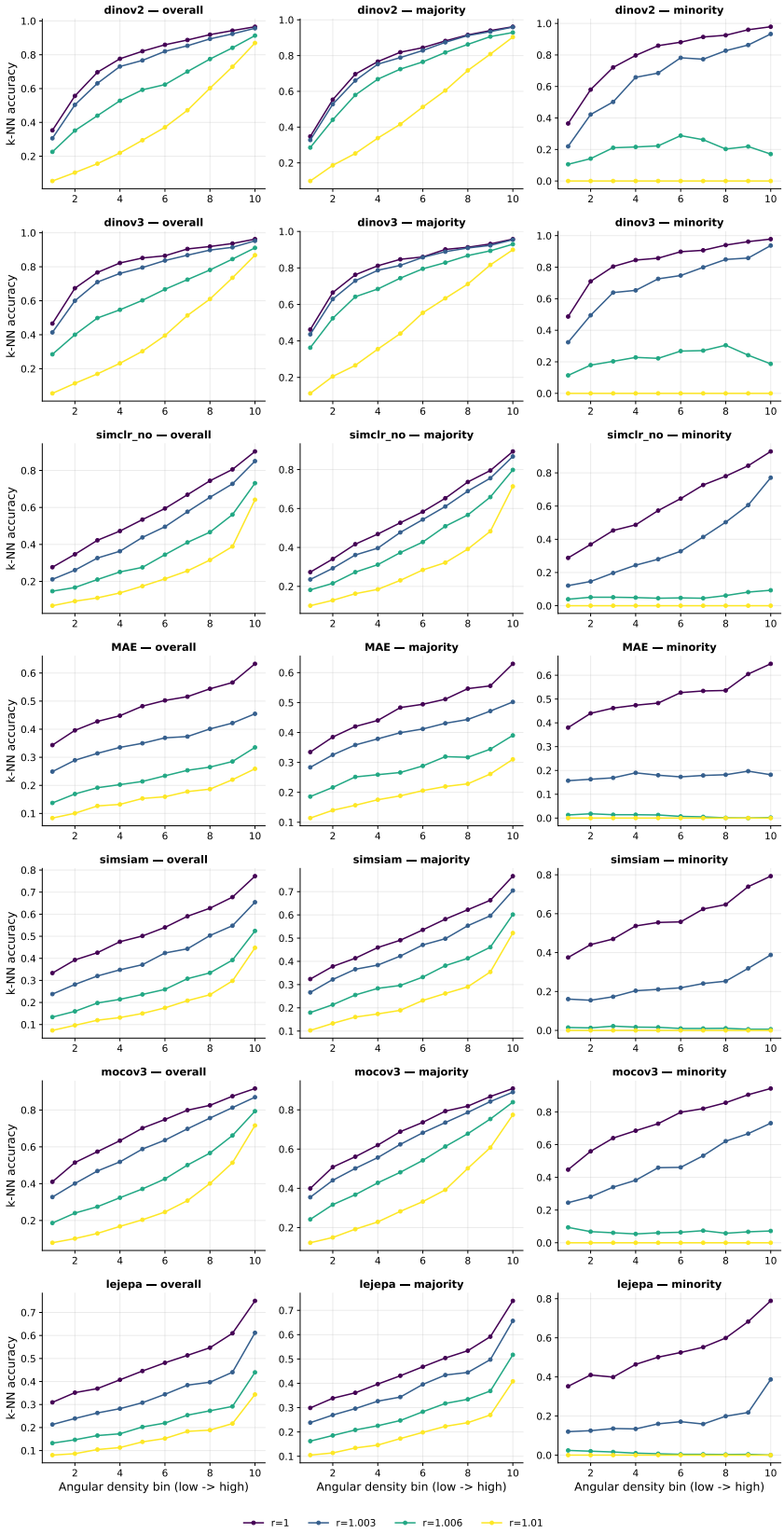


Figure 24: k**-NN accuracy vs. angular density bin under class imbalance,** for overall, majority-class, and minority-class queries. Colors denote imbalance factor r.

Out-of-distribution contamination. We contaminate the reference set with a controlled fraction of out-of-distribution embeddings (held-out ImageNet classes excluded entirely from the query set) at contamination levels up to 50%, across all seven SSL models, and evaluate PQC separately for the overall query set, queries near the injected OOD cluster, and queries far from it.

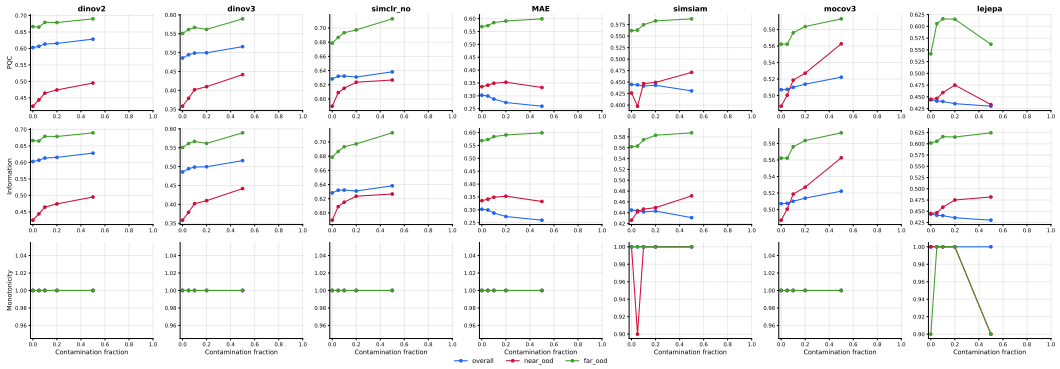


Figure 25: **PQC, Information, and Monotonicity as a function of OOD contamination fraction,** for overall, near-OOD, and far-OOD queries, at near-full reference-set size.

At near-full reference size (approximately 1281 samples per in-scope class), contamination up to 50% has little effect on PQC for most models, remaining flat or mildly increasing for both near- and far-from-OOD queries, with only a small decline for our weakest model (MAE), see Figures 25 and 27. This robustness, however, is conditional on reference-set size rather than a general property of angular density. Repeating the contamination ablation at five reference-set sizes (10, 30, 100, 300, and approximately 1281 samples per in-scope class, see Figure 28) shows that robustness to OOD contamination breaks down as the reference set shrinks, with an architecture-dependent onset that matches the class-imbalance ablation above: at 30 samples per in-scope class, DINOv2 and DINOv3 remain largely robust while every other model shows either meaningful degradation or is already near floow, and at 10 samples per class, PQC collapses toward zero under 50% contamination for nearly every model. Taken together with the class-imbalance results, reliability under either failure mode depends primarily on how much clean, in-distribution reference data is available, rather than on contamination or imbalance fraction alone.

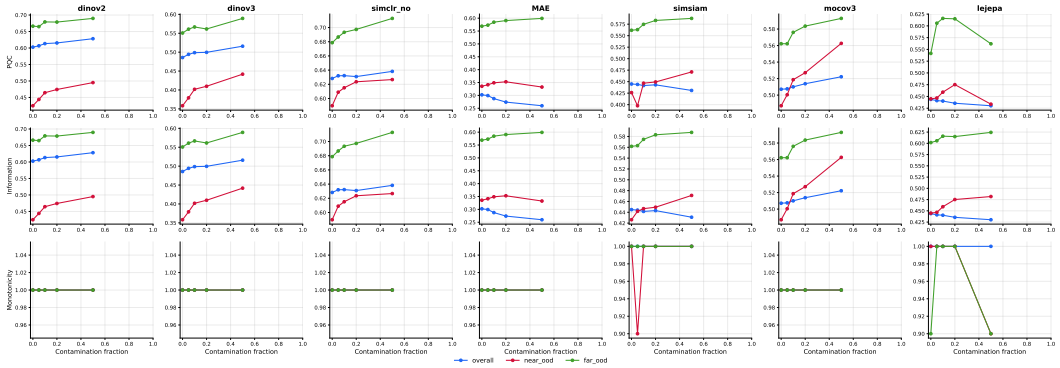


Figure 26: **PQC under OOD contamination, crossed with reference-set size.** Rows correspond to the maximum number of reference samples per in-scope class (1281 down to 10); columns correspond to overall, near-OOD, and far-OOD queries.

E Additional Results

This appendix presents results that complement the main paper: PQC trajectories over training (Section E.3) and qualitative samples (Section E.7).

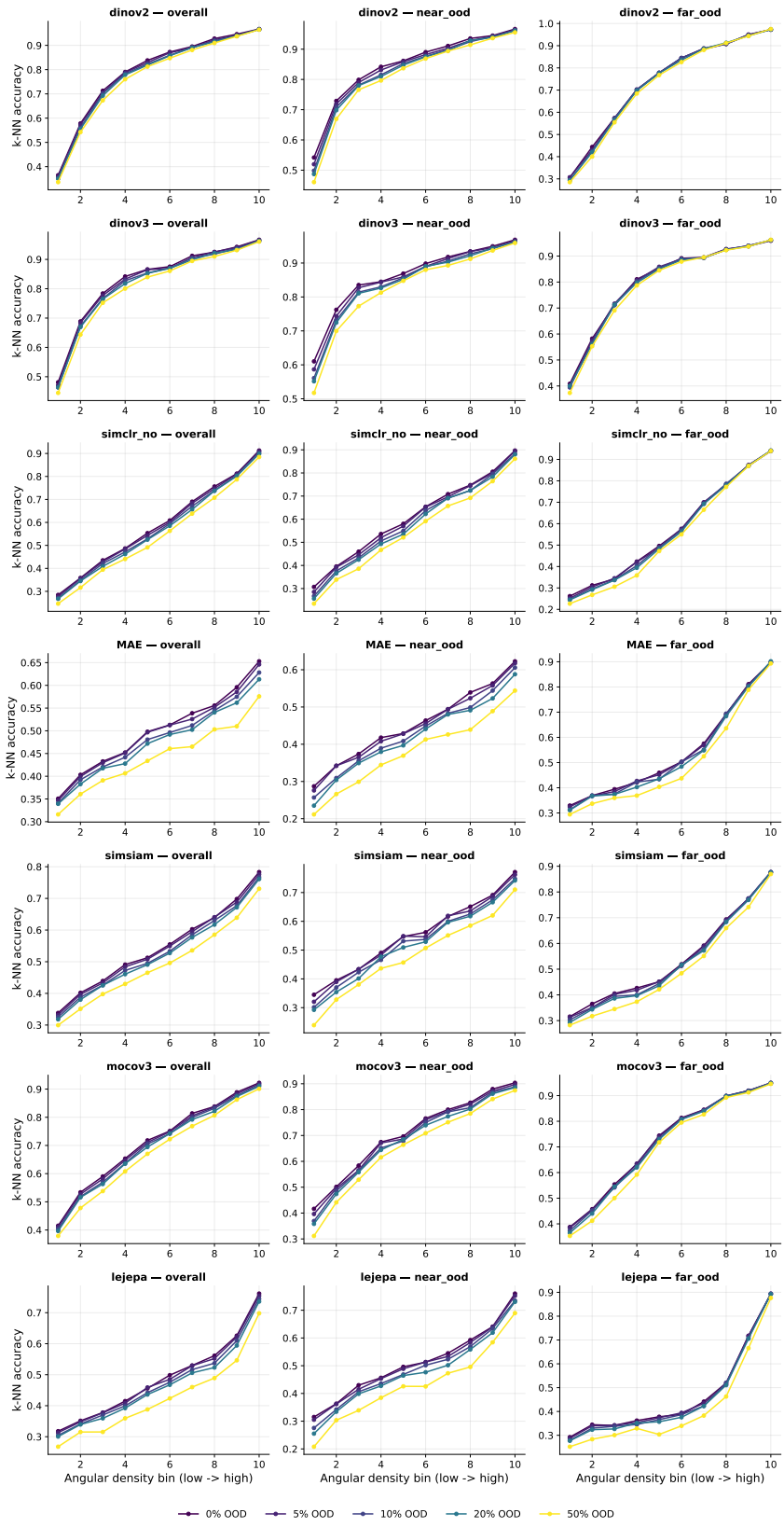


Figure 27: k**-NN accuracy vs. angular density bin under OOD contamination,** for overall, near-OOD, and far-OOD queries, at near-full reference-set size. Colors denote contamination fraction.

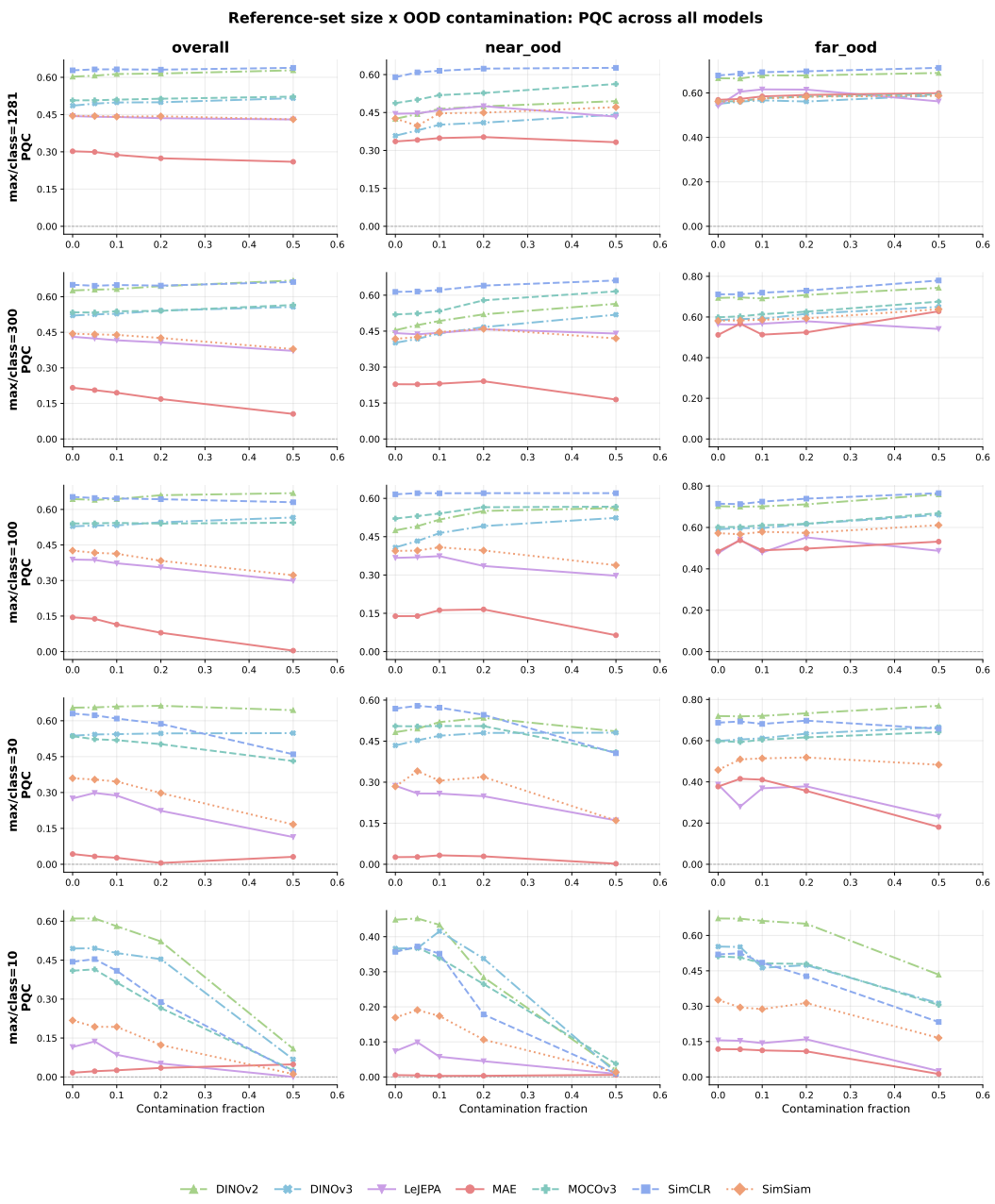


Figure 28: **PQC under OOD contamination, crossed with reference-set size.** Rows correspond to the maximum number of reference samples per in-scope class (1281 down to 10); columns correspond to overall, near-OOD, and far-OOD queries.

E.1 More Properties

We add invariance to augmentations and Mahalanobis distance to the class mean as additional candidates.

Invariance does not encode certainty. Finally, we examine augmentation invariance, shown in Figure 29. Since raw invariance scores can be confounded by local angular density, we use the density-normalized score $\tilde{p}_{\text{inv}}(z)$ throughout. We again sort embeddings into ten bins and plot the mean score against k-NN classification accuracy and average cross-model k-NN overlap.

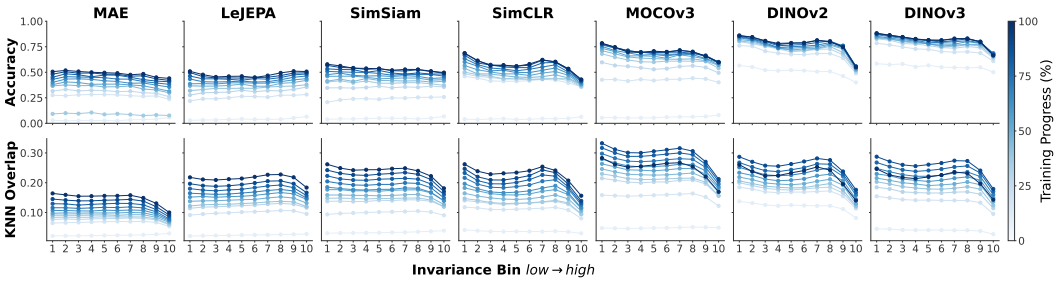


Figure 29: **Invariance to augmentation vs. model certainty.** Each column corresponds to one SSL method. Within each subplot, curves show the invariance–quality measure at different training checkpoints (colored from light to dark as training progresses). Each point represents one bin D_i, plotted at its augmentation invariance (x-axis) against its median classification accuracy (top) and weighted average k-NN overlap (bottom).

Unlike angular density and norm, augmentation invariance does not consistently predict representation quality. Although one might expect more invariant embeddings to yield better representations, Figure 29 shows flatter and more model-dependent trends. This holds for both classification accuracy and cross-model alignment. The result suggests that stability under augmentation is not equivalent to downstream separability or cross-model geometric agreement, especially after accounting for angular-density effects. Instead, invariance appears to reflect objective- and augmentation-specific inductive biases rather than a universal marker of SSL representation quality.

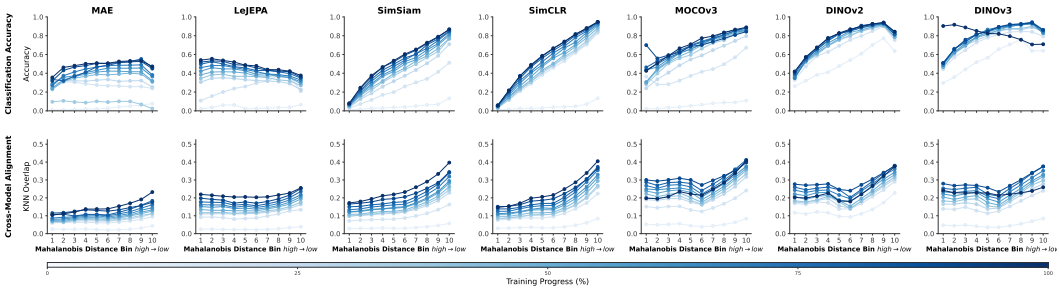


Figure 30: **Mahalanobis distance to class mean vs. model certainty.** This Figure mirrors Figure 29 but for Mahalanobis distance.

Mahalanobis Distance to Class Means encodes representation quality. A natural alternative to the geometric properties studied in the main paper is the Mahalanobis distance from an embedding to its class mean, computed using the within-class covariance. Mahalanobis distance has a long history as a confidence signal in supervised settings Lee et al. (2018a): embeddings close to their class centroid are more typical of that class and tend to be classified correctly. Indeed, when computed with ground-truth labels, Mahalanobis distance seems to be a strong per-input quality signal across most models, see Figure 30. However, it shows a weak relationship to accuracy on MAE, LeJEPA and DINOv3, and does not encode cross-model alignment consistently across models. Additionally, computing the Mahalanobis distance requires class labels, both to define the class means and to assign each embedding to its class. This renders it unsuitable as a deployment-time diagnostic in the unsupervised setting. We thus include it for reference but focus on label-free properties.

E.2 PQC AFTER TRAINING

Table 6 shows the PQC scores for all SSL models after training for both classification accuracy as well as cross-model alignment. Note that the final row shows the mean overall models. We observe that for both quality measures, angular density consistently achieves a higher PQC score.

Table 6: **SSL PQC scores for embedding properties regarding classification accuracy and cross-model alignment.** The scores are for the final training step. Models sorted by k-NN probing accuracy.

	Classification Accuracy				Cross-Model Alignment			
Model	Norm	Ang. Density	Invariance	Mahalanobis	Norm	Ang. Density	Invariance	Mahalanobis
MAE (0.47)	0.14	0.06	0.05	0.04	0.08	0.22	0.04	0.13
LeJEPA (0.49)	0.38	0.33	0.00	0.15	0.14	0.25	0.01	0.02
SimSiam (0.56)	0.30	0.29	0.07	0.79	0.15	0.30	0.05	0.23
SimCLR (0.59)	0.58	0.59	0.15	0.89	0.18	0.33	0.06	0.26
MoCov3 (0.69)	0.41	0.42	0.11	0.46	0.06	0.34	0.06	0.15
DINOv2 (0.78)	0.50	0.61	0.19	0.25	0.03	0.34	0.06	0.09
DINOv3 (0.82)	0.31	0.49	0.18	0.16	0.01	0.31	0.07	0.01
Mean	0.37 ± 0.13	0.40 ± 0.18	0.11 ± 0.07	0.39 ± 0.31	0.09 ± 0.06	0.30 ± 0.04	0.05 ± 0.02	0.12 ± 0.09

E.3 PQC over Training

The main paper reports the PQC metric only at the final epoch due to space constraints. Figures 31 and 32 extend this analysis to all checkpoints, for classification accuracy and cross-model alignment respectively. For both quality measures, the angular-density curve is more consistent across models than the corresponding curves for embedding norm and invariance.

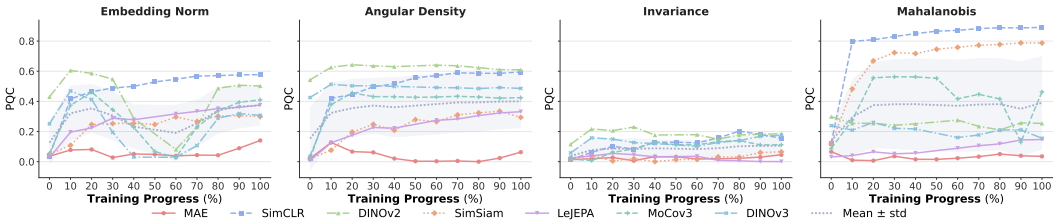


Figure 31: **PQC for classification accuracy over training.** Columns correspond to embedding properties employed as sorting criteria. Colors indicate the different SSL models. Within each subplot, curves show the PQC score at different training checkpoints. Each point represents the PQC score for a training step and model based on the embedding property for sorting the bins. Dashed gray line denotes mean values per progress step and standard deviation is shown as light gray band.

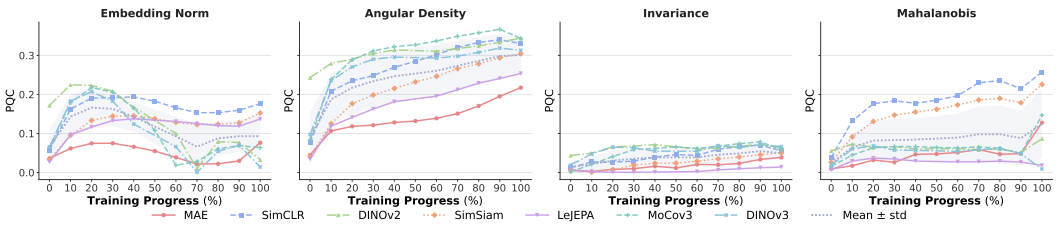


Figure 32: **PQC for cross-model alignment over training.** This figure mirrors Figure 31 for cross-model alignment.

E.4 LLM Depth Ablation

We include a copy of Figure 6a with standard deviations included at Figure 33.

E.5 Neighbor Purity and Neural Collapse

To test whether angular behaves as a label-free proxy for within-class concentration, i.e., neural-collapse-like effect, we compute *neighbor purity*, the fraction of an embedding's k nearest neighbors sharing its ground truth label, as a function of angular density bin for each of our seven SSL methods.

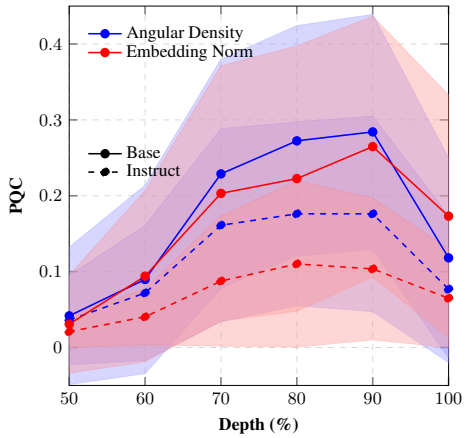


Figure 33: Identical to Figure 6a but with standard deviations included.

As shown in Figure 34, purity increases monotonically with density bin for every model. Models split into two groups by their purity ceiling at the highest-density bin: DINOv2, DINOv3, MoCov3, and SimCLR reach approximately 0.82-0.95, consistent with an emergent neural-collapse-like effect, while MAE, SimSiam, and LeJEPA reach only approximately 0.30-0.55, remaining substantially class-mixed even in their densest regions. This split matches the downstream accuracy ranking of these models exactly, suggesting that whether angular density behaves like a neural-collapse proxy is conditional on how class-discriminative the underlying representation already is, rather than a property of angular density itself.

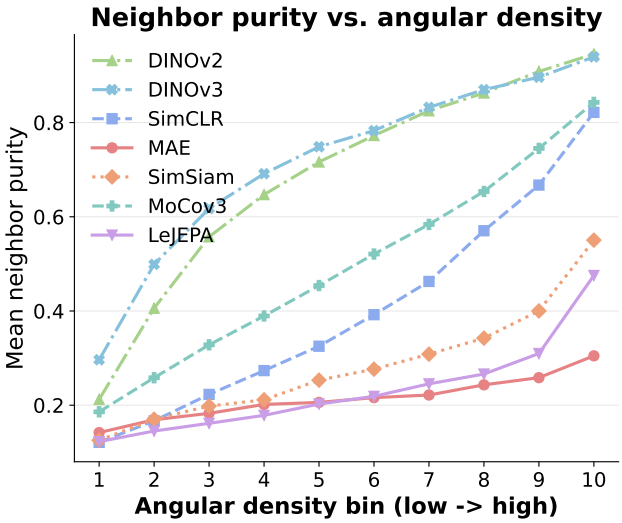


Figure 34: **Neighbor purity as a function of angular density bin.** Purity is the fraction of an embedding's k nearest neighbors sharing its ground-truth label.

E.6 Global Typicality Baseline

A natural question is whether angular density is distinct from a broader, class-independent notion of data typicality, or is simply a geometric proxy for it. To test this directly, we construct a *global typicality* baseline: Mahalanobis distance to a single centroid and covariance computed over the entire reference set, using no class labels and no local neighborhood structure — that is, literally the distance from an embedding to the center of the overall data distribution. We evaluate this baseline

using the same binning and accuracy protocol used for angular density, embedding norm, and our per-class Mahalanobis baseline (Section E.1), across all seven SSL models at their final checkpoints.

As shown in Figure 35, global typicality does not consistently track representation quality, and its direction of correlation is not stable across models. For five of the seven models (MAE, DINOv3, LeJEPA, MoCov3, SimSiam), accuracy *decreases*, nearly monotonically, as embeddings move closer to the global centroid — the opposite of what a global-typicality account would predict. Only DINOv2 and SimCLR show the expected positive relationship. Angular density, evaluated identically, is monotonically positive in all seven models. This indicates that angular density is not simply a proxy for global distributional typicality — a quantity that, measured directly, frequently points in the wrong direction — but instead reflects local neighborhood structure specifically, which predicts representation quality reliably across models where the global alternative does not.

Figure 36 extends this comparison across training checkpoints rather than only the final one, confirming that the divergence between global typicality and angular density is not specific to fully-trained models.

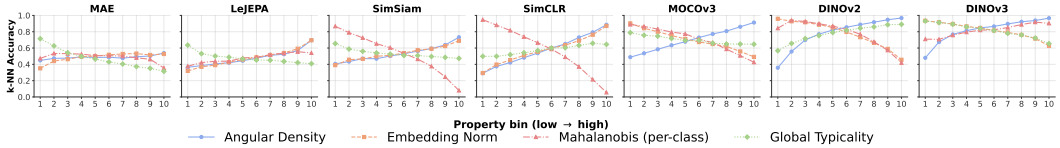


Figure 35: **k-NN classification accuracy as a function of property bin, comparing angular density, embedding norm, per-class Mahalanobis distance, and the global typicality baseline,** across all seven SSL models at their final checkpoints.

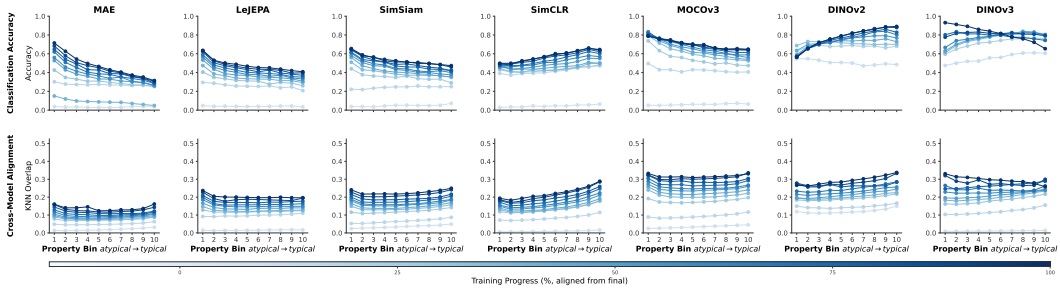


Figure 36: **Global typicality baseline across training.** Top: classification accuracy. Bottom: cross-model k-NN overlap, as a function of property bin (atypical $\rightarrow$ typical by the global-centroid measure), colored by training progress.

E.7 Qualitative Samples

We show qualitative examples of high- and low-property samples for SimCLR, MoCov3, and LeJEPA, using validation embeddings at the final epoch. We focus on embedding norm and angular density, the two properties identified in the main paper as predictive of model quality. For each model and each property, we identify the two classes with the highest mean property value and the two with the lowest, and within each, display the ten images with the most extreme values. This yields 40 images per (model, property) plot(Figures 37–42). Visually, high-property images tend to be visually clean exemplars of their class, while low-property images are atypical, occluded, or contain ambiguous content.



Figure 37: **Qualitative examples for SimCLR, angular density.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean angular density; within each, we show the ten images with the largest values. **Bottom two rows:** the two classes with the lowest mean angular density; within each, we show the ten images with the smallest densities. Class names are indicated on the left.



Figure 38: **Qualitative examples for SimCLR, embedding norm.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean embedding norm; within each, we show the ten images with the largest norms. **Bottom two rows:** the two classes with the lowest mean embedding norm; within each, we show the ten images with the smallest norms. Class names are indicated on the left.



Figure 39: **Qualitative examples for MoCov3, angular density.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean angular density; within each, we show the ten images with the largest values. **Bottom two rows:** the two classes with the lowest mean angular density; within each, we show the ten images with the smallest densities. Class names are indicated on the left.

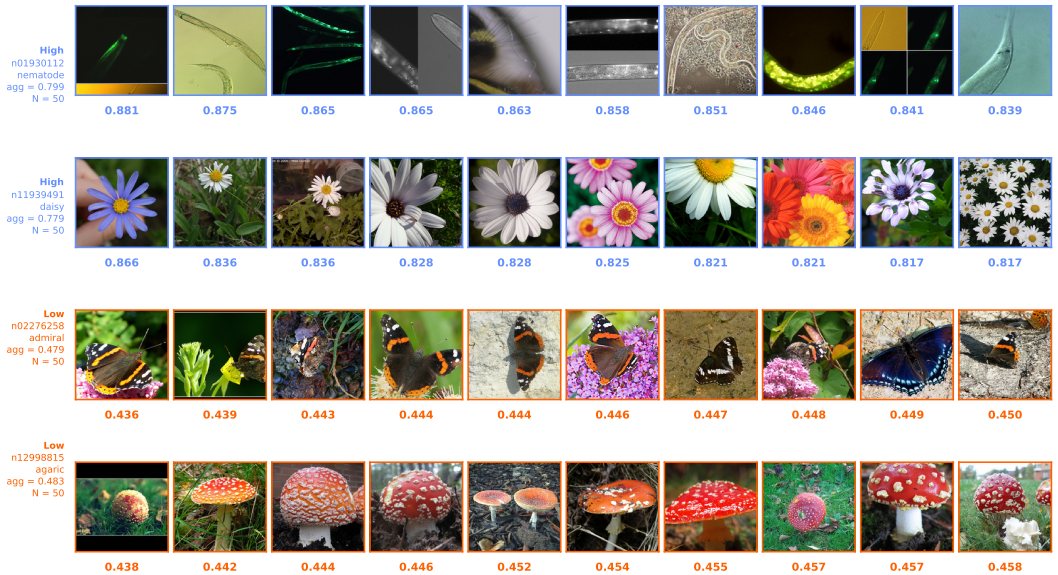


Figure 40: **Qualitative examples for MoCov3, embedding norm.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean embedding norm; within each, we show the ten images with the largest norms. **Bottom two rows:** the two classes with the lowest mean embedding norm; within each, we show the ten images with the smallest norms. Class names are indicated on the left.

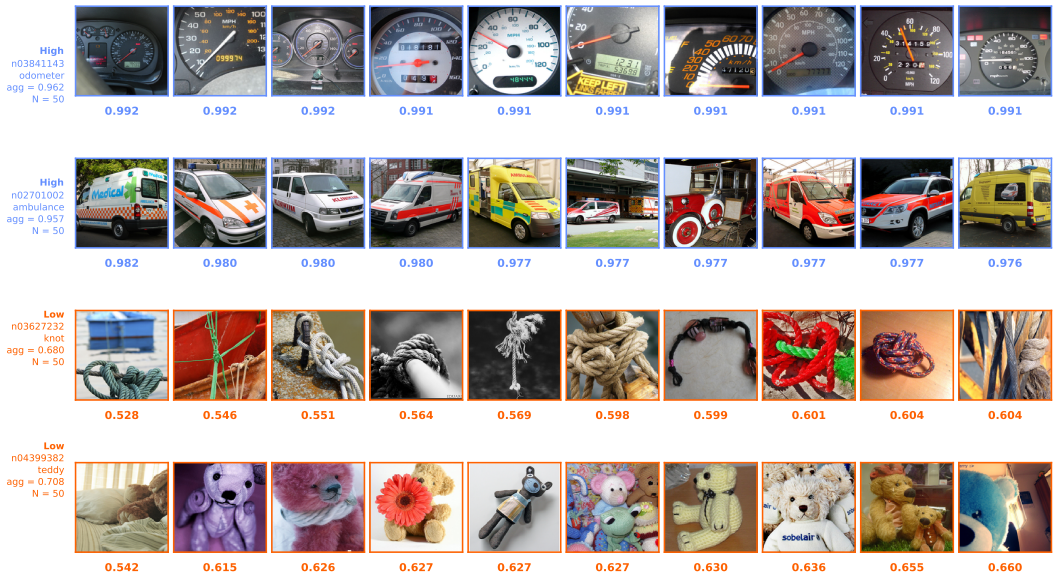


Figure 41: **Qualitative examples for LeJEPA, angular density.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean angular density; within each, we show the ten images with the largest values. **Bottom two rows:** the two classes with the lowest mean angular density; within each, we show the ten images with the smallest densities. Class names are indicated on the left.

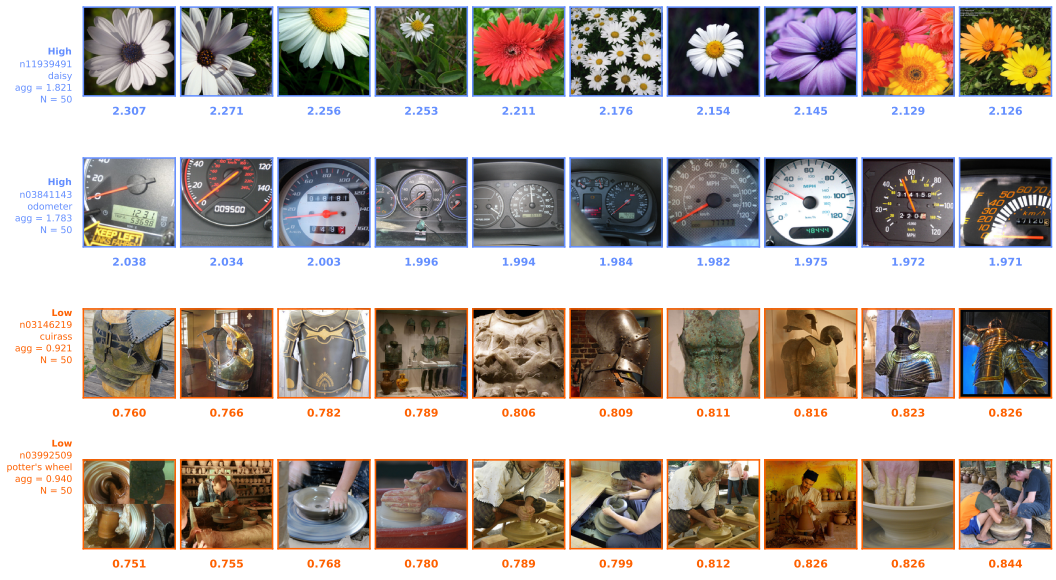


Figure 42: **Qualitative examples for LeJEPA, embedding norm.** Validation images from ImageNet-1k at the final training epoch. **Top two rows:** the two classes with the highest mean embedding norm; within each, we show the ten images with the largest norms. **Bottom two rows:** the two classes with the lowest mean embedding norm; within each, we show the ten images with the smallest norms. Class names are indicated on the left.