FAITHFUL BY CONSTRUCTION: HIERARCHY-AWARE DECISION RULES FOR BETTER MISTAKES

Anonymous authors
Paper under double-blind review

ABSTRACT

Hierarchical classification considers settings where label classes are the leaves in a tree $\mathcal{T}$, called a *taxonomy*, and leverages that taxonomy to reduce mistake severity: a misclassified image should land on a similar class and not an unrelated one. Existing methods build the taxonomy into the training recipe, through a loss, soft targets, or feature regularization, but lack a principled account of precisely how these changes affect the decision rule. We ask what it means for a classifier's decision rule to incorporate taxonomic distances *exactly*. We define *prototype rules*, classifiers that decide by measuring distances to class prototypes in feature space, and *$\mathcal{T}$-faithfulness*, the property that squared distances between class prototypes match taxonomic distances exactly. We characterize which feature layouts and classifier heads achieve $\mathcal{T}$-faithfulness, and propose a simple frozen-head training objective that realizes $\mathcal{T}$-faithfulness by construction. An empirical evaluation shows that $\mathcal{T}$-faithful models offer competitive accuracy and the lowest mistake severity among several strong baselines. Through a neural collapse analysis of a simple mean-squared-error training objective, we identify two roadblocks that make it difficult to train a $\mathcal{T}$-faithful model without freezing the classifier head: we must anticipate and account for the effective backbone regularization strength in advance of training, and the distance-based decision rule is distorted by class-dependent offsets.

1 INTRODUCTION

In hierarchical classification, label classes are the leaves of a tree with positive edge weights, called a *taxonomy* $\mathcal{T}$. The *taxonomic distance* $d_{\mathcal{T}}(u, v)$ is the sum of the edge weights on the path joining them. Taxonomic distances quantify the degree of semantic similarity between two classes, which lets us measure the severity of a mistake: if an instance of class u is mis-classified as class v, the mistake is severe precisely when $d_{\mathcal{T}}(u, v)$ is large. While many machine learning works assume all edge weights are 1 (Bertinetto et al., 2020; Zeng et al., 2023), non-uniform edge weights have applications in lexical semantics (Jiang & Conrath, 1997) and phylogenetics (Yang & Rannala, 2012).

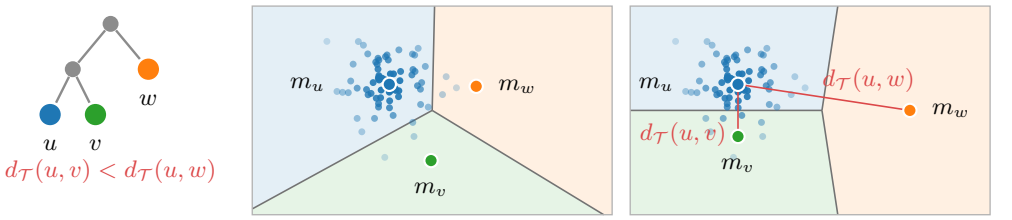
Prior works reduce mistake severity by injecting taxonomy-awareness at training time, through soft targets (Bertinetto et al., 2020), feature regularization (Zeng et al., 2023), and frozen hierarchy-aware classifier heads (Liang & Davis, 2023). Most of these shape a classifier's decision rule indirectly (Bertinetto et al., 2020; Zeng et al., 2023; Valmadre, 2022), while the frozen frame of Liang & Davis (2023) is drawn from a one-parameter family, leaving open whether a more complete treatment could reduce mistake severity further.

In this work, we show how to inject taxonomic distances directly into a model's decision rule. A standard neural network classifier's decision rule assigns instances to whichever class's logit is the largest. We will see that such decision rules admit equivalent formulations in terms of distances. They implicitly associate each class k with a vector m_k, its *prototype vector* (Snell et al., 2017), and assign each feature to whichever class's prototype vector is nearest, up to a class-dependent offset.

The distance-based decision rule formulation is appealing for our purposes. A classifier head supplies prototypes m_k while the taxonomy supplies inter-class distances $d_{\mathcal{T}}(u, v)$. To integrate taxonomic class relationships into a model's decisions, we can align prototype distances with taxonomic distances by setting

$$d_{\mathcal{T}}(u, v) \;=\; \|m_v - m_u\|_A^2, \tag{1}$$

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.



(a) taxonomy $\mathcal{T}$ (b) ignores the taxonomy (c) $\mathcal{T}$-faithful: $\|m_x - m_y\|_A^2 = d_{\mathcal{T}}(x,y)$

Figure 1: (a) A taxonomy in which v is closer to u than w. (b) A classifier whose implicit prototypes do not respect taxonomic distances. Class-u misclassifications spill over to v and w about equally. (c) A classifier whose decision rule respects taxonomic distances. Most class-u misclassifications are routed to v, the closest alternative in the taxonomy.

where $\|\cdot\|_A$ is the norm used to measure distances between features and prototypes. If the distance formulation of a classifier's decision rule lacks per-class offsets (so the score for each class is *exactly* the squared distance to that class's prototype), we call it a *prototype rule*. If a prototype rule's prototypes are arranged such that their pairwise distances match taxonomic distances, as written above, we call it $\mathcal{T}$-faithful for the taxonomy $\mathcal{T}$.

Will a $\mathcal{T}$-faithful classifier reduce mistake severity? We verify that it does empirically (§4), and offer intuition for why our approach works in Figure 1. Given a taxonomy in which u is closer to v than w (Figure 1a), a hierarchy-agnostic classifier may have no notion of that semantic relationship. An encoder learns to place features at or near their class' prototypes, but the features of hard instances can drift away from their class's prototypes. Thus, if the prototypes do not respect taxonomic class relationships, then features are equally likely to drift across the decision boundary to classes v or w (Figure 1b). Conversely, if the prototype layout respects the taxonomy, then misclassifications drift over to v before they drift to w (Figure 1c).

Our contributions are fourfold. **First,** we define $\mathcal{T}$-faithfulness, a property of classifier heads where the squared distances between their implicit prototypes match taxonomic distances. We characterize the feature layouts and frozen classifier heads that achieve $\mathcal{T}$-faithfulness (§2). **Second,** we propose a simple frozen-head training objective that realizes $\mathcal{T}$-faithfulness by construction. We fix a $\mathcal{T}$-faithful feature layout and classifier head *a priori*, freeze the classifier head, and train the encoder to place features in the intended layout (§3). Unlike many prior works in machine learning, our training objective is designed to faithfully represent arbitrary edge weights. **Third,** we offer a robust empirical evaluation of our proposed training objective. Models trained with our objective achieve competitive top-1 accuracy and better mistake severity when evaluated against several strong baselines. We also present ablations and additional experiments to characterize our design choices (§4). **Fourth,** we perform a neural collapse analysis of a simple mean-squared-error training objective, which reveals two roadblocks in realizing $\mathcal{T}$-faithfulness without freezing the classifier head: the effective backbone regularization strength must be known and accounted for in advance of training, and aligning prototype distances and taxonomic distances as in equation 1 upsets the distance-based decision rule formulation by introducing class-dependent offsets (§5).

2 Geometric Characterization of $\mathcal{T}$-Faithful Classifiers

2.1 Motivation & Design Goal

Consider the final layer of a neural network-based classifier, often called the *classifier head*. This layer is usually represented as a map $h \mapsto Wh + b$, with h a feature vector (the output of an encoder backbone), and the weight matrix W and bias vector b learned parameters. Observe that a model's decision rule admits an equivalent expression in terms of distances. Typically, a classifier head's outputs $c(h) := Wh + b$ are indexed by classes k, and the decision rule assigns features to whichever class k has the largest coordinate $c(h)_k$ in the output. Now, let A be any symmetric, positive-definite (PD) matrix. Then A defines an inner product and hence a Mahalanobis norm via $\|x\|_A^2 = x^\top Ax$ (taking $A = I$ recovers the Euclidean norm). Letting w_k denote the k-th row of W,

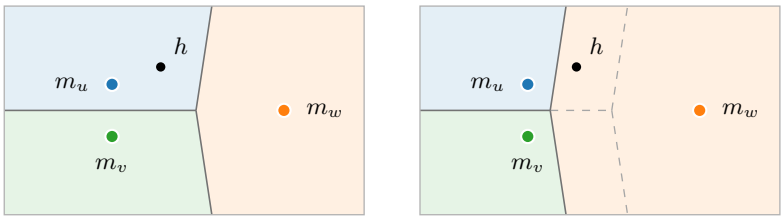


Figure 2: The same class prototypes m_u, m_v, m_w and feature h under two different choices of per-class offsets ρ_k in equation 2. Offsets can shift decision boundaries and change predictions.

and b_k the k-th scalar entry of b, we can write

$$\arg\max_k \big(\langle w_k, h\rangle + b_k\big) = \arg\min_k \big(\|h - A^{-1}w_k\|_A^2 + \rho_k\big), \qquad \rho_k := -\|A^{-1}w_k\|_A^2 - 2b_k. \quad (2)$$

The right-hand side reveals that the decision rule implicitly associates every class k with a point in feature space, $m_k := A^{-1}w_k$, and incorporates the A-distance from a feature to these representatives into each class' score. Instead of relying solely on these distances, however, the scores also incorporate a fixed per-class offset ρ_k that disrupts the distance-based formulation (Figure 2).

We now formally state what it means to integrate taxonomic distances into the decision rule. There are two requirements. The first is that squared distances between prototypes align with taxonomic distances, i.e., that

$$d_{\mathcal{T}}(u, v) = \|m_u - m_v\|_A^2 \qquad \text{for all classes } u, v. \quad (3)$$

The second is that the per-class offsets ρ_k of equation 2 are all zero. This ensures that models decide purely by distances, avoiding disruptions like those illustrated in Figure 2.

2.2 Developing the Characterization

We now describe how to incorporate fine-grained taxonomic distances into a classifier's decision rule, as in equation 3. We treat the encoder backbone as a black box mapping inputs x to features $h := \phi(x) \in \mathbb{R}^d$, and focus our study on how the classifier head $c(h) = Wh+b$ reads these features. Above, we saw that a classifier implicitly associates every class k with a representative m_k in feature space; we call each class representative its *prototype* (Snell et al., 2017), and let $M \in \mathbb{R}^{K\times d}$ be the matrix whose k-th row is the class-k prototype.

Since general classifier heads incorporate per-class offsets ρ_k in the distance formulation of their decision rules (equation 2), we focus on those where $\rho_k = 0$ for every k:

Definition 1. (M, W, A, b) is a *prototype rule* with metric A (SPD) if $w_k = Am_k$ and $b_k = -\frac{1}{2}m_k^\top Am_k$ for all classes k, where m_k, w_k are the k-th rows of M and W respectively.

A prototype rule has the form of a uniform-prior linear discriminant analysis (LDA) rule with covariance A^{-1} and means m_k (Hastie et al., 2009). While LDA fits such rules to pre-existing features, we fix M and W to respect a taxonomy and train a model to realize the prescribed geometry. The decision rules of prototype rules reduce to choosing the nearest class prototype in feature space. Applying class-independent translations to weights or biases would still allow us to express decision rules purely in terms of distances (equation 2), so there are more classifiers that could reasonably be considered "prototype rules". We can easily apply such transformations after training, so our focus on this restricted set does not sacrifice generality.

We want to determine how to align squared A-distances between prototypes with taxonomic distances, a property we formalize in the following definition.

Definition 2. A prototype rule with metric A is $\mathcal{T}$-faithful for a taxonomy $\mathcal{T}$ if $\|m_u - m_v\|_A^2 = d_{\mathcal{T}}(u, v)$ for all classes u, v, where m_u, m_v are the prototypes for u and v respectively.

It remains to describe the set of $\mathcal{T}$-faithful prototype rules. For a taxonomy $\mathcal{T}$, let $L_{\mathcal{T}}$ be the Laplacian matrix after Kron reduction onto the leaf set (see §C for background on this object). For every pair of nodes u, v, the Laplacian encodes an effective resistance $R_{\text{eff}}(u, v)$, and on a connected tree, effective resistances coincide with taxonomic distances (i.e., $R_{\text{eff}} = d_{\mathcal{T}}$). Kron reduction onto

a reduced set preserves effective resistances, so we can characterize the $\mathcal{T}$-faithful prototype rules through $L_{\mathcal{T}}$.

Theorem 3. *Let (M, W, A, b) be a prototype rule, with d the dimension of feature vectors. Let $J = I_K - \frac{1}{K}\mathbb{1}\mathbb{1}^\top$ be the orthogonal projection onto $\mathbb{1}^\perp$, where K is the number of classes and $\mathbb{1} \in \mathbb{R}^K$ the all-ones vector.*

(i) The rule is $\mathcal{T}$-faithful if and only if $JWM^\top J = L_{\mathcal{T}}^\dagger$, where $L_{\mathcal{T}}^\dagger$ is $L_{\mathcal{T}}$'s pseudo-inverse.

(ii) Fix prototypes M. There exist W, A, b such that (M, W, A, b) is a $\mathcal{T}$-faithful prototype rule if and only if $\operatorname{rank}(JM) \geq K - 1$.

A proof is given in §H. The theorem turns our desideratum (equation 3) into a closed-form condition. $L_{\mathcal{T}}^\dagger$ is fixed by $\mathcal{T}$, so the $\mathcal{T}$-faithfulness condition $JWM^\top J = L_{\mathcal{T}}^\dagger$ can be enforced during training. We remark that conjugation by J serves only to center prototypes: $\mathcal{T}$-faithfulness is determined by the prototypes' locations relative to each other, and is insensitive to global translations. Theorem 3(ii)'s condition is mild, asking only for affinely-independent features.

3 Proposed Training Objective

We now introduce a simple training procedure that can realize arbitrary $\mathcal{T}$-faithful prototype rules. Following Liang & Davis (2023), who fix a classifier frame in advance and train an encoder to match, we choose a $\mathcal{T}$-faithful prototype rule *a priori*, freeze it as the classifier head, and learn an encoder backbone $\phi : \mathcal{X} \to \mathbb{R}^e$ and projection layer $\psi : \mathbb{R}^e \to \mathbb{R}^{K-1}$ to place class-k features at the prototype m_k. Theorem 3 tells us that $K - 1$ is the smallest feature dimension that allows $\mathcal{T}$-faithfulness, although any $d \geq K - 1$ would suffice. ψ is the same projection head used by Liang & Davis (2023); see §D for details.

Soft targets. Suppose we have a $\mathcal{T}$-faithful prototype rule. Passing the class-k prototype m_k through the classifier head, we obtain the logits $Wm_k + b$. The class-ℓ logit is

$$(Wm_k + b)_\ell \;=\; -\tfrac{1}{2}\|m_k - m_\ell\|_A^2 + \tfrac{1}{2}m_k^\top A m_k \;=\; -\tfrac{1}{2}d_{\mathcal{T}}(k, \ell) + \tfrac{1}{2}m_k^\top A m_k.$$

We can read off a posterior by applying a temperature-scaled softmax to the logits. The second term above does not depend on ℓ, so the model's posterior at a prototype m_k is defined element-wise by

$$(q_k)_\ell \;:=\; \operatorname{softmax}\big((Wm_k + b)/\tau\big)_\ell \;\propto\; \exp\big(-d_{\mathcal{T}}(k, \ell)/2\tau\big),$$

for $\tau > 0$ a temperature hyperparameter. These are soft targets in the sense of Bertinetto et al. (2020), but our targets are derived specifically to place features at their class's prototype. The same value of τ can yield sharper targets on one taxonomy than another. Since there is a bijection between τ and the mean true-class surprisal $\mathrm{NLL}_\star = -\frac{1}{K}\sum_k \log(q_k)_k$ (§F), we sweep $\mathrm{NLL}_\star$ instead of τ.

Training. For a training sample x of class k, let $h = \psi(\phi(x))$ be its feature and $\hat{q}(h) = \operatorname{softmax}\big((Wh + b)/\tau\big)$ the classifier head's posterior at h. The per-sample loss is $\mathrm{KL}(q_k \,\|\, \hat{q}(h))$. It is zero if and only if $h = m_k$, so the objective pulls features onto their class's prototype.

4 Experiments

Setup. To test our method's (§3) empirical performance, we evaluated on three datasets: CIFAR-100 (Krizhevsky & Hinton, 2009), FGVC-aircraft (Maji et al., 2013), and Stanford-Cars (Krause et al., 2013). CIFAR-100 and FGVC-Aircraft both come with canonical taxonomies (2 and 3 levels, respectively); for Stanford-Cars, we construct a 2-level taxonomy by grouping classes of the same make. We use SGD with a one-cycle schedule for 100 epochs. Our method benefits from clipping gradient norms to 1. For a fair comparison, we also train every baseline with and without norm clipping. The effects of this knob are discussed in §G.4. We report top-1 accuracy and TreeDist@k for $k \in \{1, 5, 20\}$, a hierarchical error metric. The TreeDist@k statistic averages the taxonomic distance from a model's top k predictions to the true class; we report the average of this statistic

over all test samples.[1] Model selection uses TreeDist@1 on the validation split. We train two further seeds at the winning hyperparameter cell and report metrics measured on the test split, averaged over three runs. Each dataset's tree is weighted two different ways, giving a uniform taxonomy (where every edge weight is 1) and a CLIP taxonomy (where taxonomic distances match squared distances between per-class CLIP embeddings; see §E). Some methods allow us to integrate non-uniform edge weights into training. For these, the suffix "(unif.)" indicates that we trained with uniform edge weights, while "(CLIP)" indicates that we trained with the CLIP-derived edge weights. Further details necessary for reproducibility are given in §B.

Baselines. We now introduce our baselines through the lens of Theorem 3, which describes which combinations of prototype layouts M and classifier layers W achieve $\mathcal{T}$-faithfulness. Our simplest baseline is standard cross-entropy training (**CE**). Hierarchical Cross-Entropy **(HXE)** (Bertinetto et al., 2020) trains with taxonomy-derived soft targets. This and other soft-target losses (Valmadre, 2022) can be understood as supervising the product $MW^\top$ and trusting the optimizer to settle on the factors M, W. Cophenetic Correlation Coefficient feature regularization **(CPCC)** (Zeng et al., 2023) uses the CPCC statistic to align Euclidean distances between each class's samples' features with taxonomic distances. This approach can be understood as supervising M but not W, which may allow an anisotropic W to distort the feature layouts (Zeng et al., 2025; Barz & Denzler, 2019). Finally, **HAFrame** (Liang & Davis, 2023) is the most similar to our method. They construct and freeze a taxonomy-aware set of final-layer class weight vectors, which they call a *classifier frame*, whose pairwise angles encode taxonomic distances. They then train the encoder so that features from each class align with the corresponding classifier vector. This approach can be understood as fixing M and W *a priori*, freezing W, and teaching an encoder backbone to place features at the prototypes M. Their loss has an alignment term applied to features and a cross-entropy term applied to the classifier frame's outputs, while we apply only a KL penalty to our head's outputs. Moreover, while our frozen frame can express arbitrary edge weights, including uniform edge weights, their frame is restricted to a family parameterized by a single spacing parameter γ. Each method is swept over its native hyperparameter grid; ours sweeps the learning rate and $\mathrm{NLL}_\star$.

Frozen geometry. Our method (§3) requires the practitioner to fix a $\mathcal{T}$-faithful geometry before training. Theorem 3 tells us that there are infinitely many faithful prototype rules for any given taxonomy. In selecting our frozen geometry, we looked to our neural collapse discussion (§5) and used the prototypes M and classifier W realized by decision-faithful targets (equation 5) with $S = L_{\mathcal{T}}$. Other choices are possible, but we leave these to future work, finding that the selected configuration already yields good results. We set the feature dimension to $K - 1$, where K is the number of classes, and use centered prototypes.

Classification performance. Table 1 compares our method's classification performance against several baselines. On uniform taxonomies, our method is at or near the best accuracy and improves TreeDist@k on every dataset. On CLIP taxonomies, our method's relative improvement to TreeDist@5 is even larger, at the cost of some top-1 accuracy; however, its strong TreeDist@1 metrics suggest that any extra errors result from confusing very similar classes. Training baselines with CLIP weights does not consistently improve their mistake severity versus uniform weights.

Is the geometry realized? Our frozen-head loss (§3) is intended to train encoders to place features on their classes' prototypes. We empirically confirm that it succeeds in Figure 3. These runs were trained for 350 epochs to ensure convergence. At the end of every epoch, we computed the A-distance from each validation sample's feature to its class's prototype. The median and IQR of this distance is plotted for every epoch in Figure 3. We present results for CIFAR-100 with uniform edge weights here and defer other taxonomies to §G.3. These plots confirm that the encoder is placing features at prototypes. We find that correctly-classified samples are typically nearer their prototype than incorrectly-classified samples, as expected.

Are class-dependent offsets harmful? In §2, we showed that the distance formulation of classifier heads' decision rules is upset by class-dependent offsets ρ_k. Our frozen head training objective avoids these by construction (§3). While the correction is satisfying theoretically, we have not yet

[1] TreeDist@k is a weighted-tree analogue of hierarchical distance@k (Bertinetto et al., 2020), whose LCA height measurements ignore edge weights.

Table 1: Classification accuracy and mistake severity; we report means over three seeds. Column groups differ by which edge weights were used to *score* TreeDist@k; the groups are on different scales and are not comparable. Methods with (unif.) were trained with uniform edge weights, and methods with (CLIP) were trained with the CLIP-derived edge weights (§E). Standard deviations and TreeDist@20 are recorded in Tables 3 and 4.

	Uniform taxonomy			CLIP taxonomy (TreeDist × 100)		
Method	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓
CIFAR-100						
CE	75.33	0.799	2.66	75.31	3.02	10.86
CPCC (unif.)	74.84	0.795	2.03	75.35	2.99	8.48
CPCC (CLIP)	N/A	N/A	N/A	**75.82**	**2.94**	9.69
HXE (unif.)	75.09	0.795	2.42	75.26	2.98	9.89
HXE (CLIP)	N/A	N/A	N/A	74.50	3.05	10.17
HAFrame	**75.78**	0.784	2.13	75.74	2.99	9.10
Ours (unif.)	75.58	**0.779**	**1.97**	75.76	2.95	8.28
Ours (CLIP)	N/A	N/A	N/A	74.95	2.95	**8.18**
FGVC-Aircraft						
CE	79.74	0.902	4.17	79.61	4.36	23.50
CPCC (unif.)	81.97	0.769	3.64	81.38	3.74	20.49
CPCC (CLIP)	N/A	N/A	N/A	81.18	3.82	19.59
HXE (unif.)	81.14	0.781	4.01	80.74	3.70	22.28
HXE (CLIP)	N/A	N/A	N/A	33.71	15.94	23.86
HAFrame	**82.55**	0.734	3.50	82.48	3.54	19.40
Ours (unif.)	**82.55**	**0.673**	**3.45**	**82.66**	3.30	19.56
Ours (CLIP)	N/A	N/A	N/A	81.12	**2.70**	**16.80**
Stanford Cars						
CE	86.64	0.414	2.76	87.07	4.81	33.54
CPCC (unif.)	86.42	0.419	2.63	88.44	4.09	27.22
CPCC (CLIP)	N/A	N/A	N/A	87.66	4.43	29.15
HXE (unif.)	87.61	0.372	2.62	87.12	4.68	31.78
HXE (CLIP)	N/A	N/A	N/A	77.09	9.39	32.38
HAFrame	86.87	0.395	2.13	86.89	4.72	26.89
Ours (unif.)	**89.57**	**0.297**	**2.05**	**88.89**	3.91	26.48
Ours (CLIP)	N/A	N/A	N/A	88.67	**3.71**	**24.65**

established whether it is practically necessary. Set the bias $b_k = 0$ in a $\mathcal{T}$-faithful prototype rule; then a per-class offset of $\rho_k = -\|m_k\|_A^2 = -(L_{\mathcal{T}}^\dagger)_{kk}$ is added in the decision rule. This is exactly the offset predicted in §5. The margin between prototypes m_u, m_v changes from $d_{\mathcal{T}}(u, v)$ in a prototype rule to $d_{\mathcal{T}}(u, v)(1 - \Delta_{uv})$, where $\Delta_{uv} = (\rho_u - \rho_v)/d_{\mathcal{T}}(u, v)$. The prototype m_u is itself mis-classified once $\Delta_{uv} \geq 1$. Let $\Delta_u = \max_v \Delta_{uv}$. Δ_u vanishes when each class has the same average taxonomic distance to the rest (Lemma 20), and grows as those averages disperse. On the uniform CIFAR-100 taxonomy, every class is exchangeable with every other, so $\Delta_u \equiv 0$. The uniform Aircraft and Cars taxonomies are less symmetric, but Δ_u remains small. The CLIP taxonomies spread these averages widely, and Δ_u is larger on each as a result. Table 2 also records the performance impact of the offset in the two rightmost columns. On uniform taxonomies with $\Delta_u \approx 0$, the difference is under a point and under a percent in either column. On CLIP taxonomies, removing the correction consistently lowers accuracy and raises mistake severity by margins well outside of seed variation. FGVC-Aircraft loses 6 points of accuracy and 15% on TreeDist@5.

Additional results. We offer further results in §G, including extended versions of Table 1 and Figure 4 (§G.1, §G.3). §G.2 examines the impact of $\mathrm{NLL}_\star$ on our method's performance. §G.5 describes an experiment where we replace a learned classifier with a frozen $\mathcal{T}$-faithful head. The frozen head consistently improves TreeDist@5 at the cost of two to five points of accuracy.

5 A Neural Collapse Perspective

Table 2: Δ_u **over classes.** Δ_u is the largest fraction of class u's score margin removed by setting $b_k = 0$. This is computed from the taxonomy alone. **Uncorrected training.** Summarizes the performance impact of zeroing the bias b_k. Measurements are computed as uncorrected minus corrected over three-seed means: accuracy in points, and TreeDist@5 relative to the corrected run.

	Δ_u over classes				Uncorrected training	
Dataset	median	$\geq \frac{1}{4}$	$\geq \frac{1}{2}$	max	ΔAcc. (pts)	ΔTreeDist@5 (rel. %)
Uniform taxonomy						
CIFAR-100	0.00	0%	0%	0.00	−0.33	− 0.6
FGVC-Aircraft	0.02	0%	0%	0.09	+0.48	+ 0.4
Stanford Cars	0.01	0%	0%	0.05	−0.56	+ 0.6
CLIP taxonomy						
CIFAR-100	0.35	79%	14%	0.89	−1.19	+ 2.6
FGVC-Aircraft	0.54	88%	58%	0.98	−6.23	+15.1
Stanford Cars	0.25	51%	6%	0.72	−1.41	+ 9.1

Our frozen-head objective (§3) yields $\mathcal{T}$-faithfulness by construction. Could a loss achieve this with a learned head? We address this by studying a mean-squared-error objective in the unconstrained features model of neural collapse, and derive in closed form regression targets that match squared prototype distances and taxonomic distances exactly (equation 5). However, two roadblocks emerge: per-class offsets are not constant across classes (so they can change decisions, see Figure 2), and we must compensate for an effective regularization strength not known in advance of training.

5.1 Neural Collapse Background

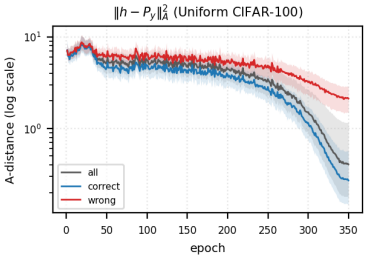


Figure 3: A-distance from validation features to their class prototype. Lines are medians; bands are IQR.

The terminal phase of training is the regime where training error has disappeared but loss minimization continues. Classical neural collapse results describe four phenomena, NC1–NC4, that emerge in this regime when training with standard cross-entropy (CE) or mean-squared-error (MSE) objectives (Papyan et al., 2020). These phenomena were first observed empirically and subsequently explained theoretically (Mixon et al., 2020). The results are as follows. **NC1** states that intra-class feature variance disappears. Every training sample is mapped to its class's mean feature vector. **NC2** states that centered class feature means form a simplex equiangular tight frame (ETF), a symmetric arrangement of K points (where K is the number of classes). A simplex ETF is defined by K unit vectors that sum to zero and whose pairwise angles are equal. **NC3** says that the classifier weight vector for each class is proportional to that class's centered mean feature vector, with a single scale shared across classes. Finally, **NC4** states that, given a feature vector h, the classifier head predicts whichever class's mean feature vector is nearest to h. These results show that standard classifiers treat every pair of classes as equally similar.

Neural collapse is analyzed theoretically in the unconstrained features model (UFM) (Mixon et al., 2020; Zhu et al., 2021; Zhou et al., 2022), which treats encoder outputs (i.e., features) and the classifier head's weights and biases as free variables in an optimization problem. This can be understood as abstracting out the encoder backbone by assuming it is sufficiently expressive to realize any features, an assumption made realistic by the overparameterized encoders of modern deep learning.

5.2 Realizing $\mathcal{T}$-faithfulness in the UFM

Existing neural collapse literature assumes one-hot regression targets. Our analysis is stated in terms of generic regression targets, which allows us to discuss which choices can achieve $\mathcal{T}$-faithfulness.

The objective we study is parameterized by $M, W \in \mathbb{R}^{K\times(K-1)}$, prototype and classifier matrices:

$$\mathcal{J}_Y(M,W) \;=\; \|MW^\top - Y\|_F^2 \;+\; \lambda_W\|W\|_F^2 \;+\; \lambda_M \operatorname{tr}\left(M^\top S M\right); \qquad \text{such that } M^\top \mathbb{1} = 0. \tag{4}$$

$\lambda_W, \lambda_M > 0$ control regularization strength. $Y \in \mathbb{R}^{K\times K}$ stores regression targets; its k-th row is the regression target for class k. $S \in \mathbb{R}^{K\times K}$ is either I or $L_\mathcal{T}$, the tree Laplacian matrix of $\mathcal{T}$, Kron-reduced onto the leaf set (§C). If $S = I$, the third term is weight decay on features, recovering the MSE objective already studied in neural collapse (Zhou et al., 2022). If $S = L_\mathcal{T}$, it is instead a standard graph-smoothness penalty (Zhu et al., 2003; Belkin et al., 2006; Smola & Kondor, 2003) rewarding the model for placing features of taxonomically-similar classes near one another. The objective is the exact reduction of a loss with balanced classes and a learnable bias, which is what we train in practice (equation 31). Substituting the closed-form optimal bias centers the class means (yielding the $M^\top\mathbb{1} = 0$ constraint) and absorbs any component of Y along $\mathbb{1}$ (the all-ones vector), so we assume $Y\mathbb{1} = 0$. Below, we will restrict the admissible values of Y to ensure that every minimizer of the per-sample loss places each training feature on its class's mean feature, allowing us to state the above in terms of prototypes rather than individual training samples.

Since $L_\mathcal{T}$ is the Laplacian of a connected graph, it has a simple zero eigenvalue and $K-1$ positive eigenvalues (§C). Write these $0 = \nu_1 < \nu_2 \le \cdots \le \nu_K$ (equation 6), and fix a corresponding orthonormal eigenbasis $\phi_1, \ldots, \phi_K$ with ϕ_l an eigenvector for ν_l. Then both choices of S admit an eigendecomposition in the $L_\mathcal{T}$ eigenbasis, $S = \sum_l s_l\,\phi_l\phi_l^\top$, with $s_l = 1$ or $s_l = \nu_l$ for $S = I$ and $S = L_\mathcal{T}$ respectively. Call Y *admissible* if we can write $Y = \sum_{l\ge 2}\gamma_l\,\phi_l\phi_l^\top$ with $\gamma_l \ge 0$ constant on each eigenspace. Intuitively, this condition means that the targets cannot distinguish directions that the taxonomy (as represented by $L_\mathcal{T}$) does not.

Theorem 4. *Let Y be admissible, $d = K-1$ (where d is the feature dimension and K the number of leaf classes), and $\lambda_M, \lambda_W > 0$. Let $0 = \nu_1 < \nu_2 \le \cdots \le \nu_K$ be the Laplacian $L_\mathcal{T}$'s eigenvalues, with an orthonormal eigenbasis $\phi_1, \ldots, \phi_K$ fixed such that ϕ_l is an eigenvector for ν_l. Then $\mathcal{J}_Y$ attains its minimum, its minimizers are unique up to a common orthogonal transformation, and every minimizer (M^*, W^*) admits a SPD A such that, if we write $Y = \sum_{l\ge 2}\gamma_l\phi_l\phi_l^\top$,*

(i) $W^* = M^*A$;

(ii) $\arg\max_k \langle w_k^*, h\rangle \;=\; \arg\min_k \left(\|h - m_k^*\|_A^2 \;-\; \|m_k^*\|_A^2\right)$ *for all* $h \in \mathbb{R}^d$;

(iii) $M^*AM^{*\top} \;=\; \sum_{l\ge 2} c_l\,\phi_l\phi_l^\top$, *where* $c_l = \left[\,\gamma_l - \kappa_l\,\right]_+$ *and* $\kappa_l = \sqrt{\lambda_M\lambda_W s_l}$.

See §I for a proof, and §K for validation that these predictions are realized in real models. Hierarchical analogues of classical neural collapse results NC2–NC4 emerge as corollaries (§J). Part 4(i) shows that trained class mean features satisfy the weight condition $w_k^* = Am_k^*$ of Definition 1, while part 4(ii) tells us that the decision rule is nearest-class-feature-mean under A, up to per-class offsets. Nothing in equation 4 mentions prototypes or distances, yet its minimizers carry the structure of §2. This is evidence that prototype rules (Definition 1) are an appropriate abstraction. Note, however, that (M^*, W^*) can only be decision-equivalent to a prototype rule if the per-class offsets $-\|m_k^*\|_A^2$ of Part 4(ii) are constant across classes.

Now, we focus on Part 4(iii). The object $M^*AM^{*\top}$ stores the A-distances between each pair of class prototypes (i.e., mean features). For two classes u and v, we have

$$\|m_u^* - m_v^*\|_A^2 \;=\; \sum_{l\ge 2} c_l\,(\phi_l(u) - \phi_l(v))^2.$$

Now, note that for any two nodes u, v in $\mathcal{T}$, the *effective resistance* between them is (§C)

$$R_{\text{eff}}(u,v) \;=\; \sum_{l\ge 2} \nu_l^{-1}\,(\phi_l(u) - \phi_l(v))^2,$$

where (recall) ν_l are the eigenvalues of $L_\mathcal{T}$. On a connected tree, effective resistances and taxonomic distances are always equal, i.e., $R_{\text{eff}}(u,v) = d_\mathcal{T}(u,v)$. Thus, the striking similarity between the above two equations offers a simple way to guarantee that squared prototype distances equal taxonomic distances: choose Y such that $c_l = \nu_l^{-1}$ for each mode $l \ge 2$. Solving the per-mode

equations $c_l = \gamma_l - \kappa_l = \nu_l^{-1}$ for γ_l, we obtain the following *decision-faithful regression targets,* with J the projection onto $\ker(L_{\mathcal{T}})^{\perp}$:

$$Y = \begin{cases} L_{\mathcal{T}}^{\dagger} + \sqrt{\lambda_M \lambda_W}\, J, & \text{if} \quad S = I \\ L_{\mathcal{T}}^{\dagger} + \sqrt{\lambda_M \lambda_W}\, L_{\mathcal{T}}^{1/2}, & \text{if} \quad S = L_{\mathcal{T}} \end{cases} \tag{5}$$

Class-dependent offsets are inevitable. Can we design regression targets such that A-distances between prototypes match taxonomic distances *and* the per-class offsets $-\|m_k^*\|_A^2$ are constant across classes? The centering constraint and Theorem 4(i) give $JW^*M^{*\top}J = M^*AM^{*\top}$; then Theorem 3(i) tells us that A-distances between prototypes match taxonomic distances if and only if $M^*AM^{*\top} = L_{\mathcal{T}}^{\dagger}$. Since $-M^*AM^{*\top}$ carries the per-class offsets on its diagonal, if A-distances are faithful, the class-k offset must be $-(L_{\mathcal{T}}^{\dagger})_{kk}$. Up to a constant shared by all classes, $(L_{\mathcal{T}}^{\dagger})_{kk}$ is the average taxonomic distance from class k to the others (Lemma 20). The offsets are equal precisely when every class is, on average, equally far from the rest, which means that many taxonomies can never realize $\mathcal{T}$-faithful prototype rules through this objective.

Regularization distorts the geometry. Observe that using the decision-faithful targets (equation 5) requires that we know, in advance of training, the feature regularization strength λ_M. When we realize neural collapse empirically (§K), we disable encoder weight decay and explicitly regularize features alone with an explicit λ_M parameter. Real networks benefit enormously from weight decay on an entire encoder backbone, and disabling it to justify a choice of regression targets is not practical. Thus, the *effective* strength of feature regularization is not a scalar λ_M known up-front, but rather a property of an encoder estimable only after training—i.e., long after we would have chosen our regression targets. This is a structural challenge of simultaneously shaping feature and classifier head geometry through a loss: regularization acts on each component separately. In classical neural collapse, the realized geometry is symmetric about the origin (§5.1), so regularization shrinks it uniformly without distorting it. $\mathcal{T}$-faithful geometry requires $c_l = \nu_l^{-1}$ in each mode, so failing to adjust for the κ_l subtraction exactly warps the geometry.

Freezing the classifier head. Using a frozen classifier head (§3) defuses both of the above issues at once. We get a prototype rule and no longer need to precisely correct for encoder regularization.

6 Discussion and Conclusions

In this paper, we introduced the concepts of *prototype rules* and *$\mathcal{T}$-faithfulness,* and characterized which feature and classifier geometries realize faithful decisions (§2). We showed that a standard squared-error training objective can achieve faithfulness up to a per-class offset (§5), and proposed a simple frozen-head alternative to realize faithfulness exactly (§3). Finally, we showed that our frozen-head objective matches or improves on every baseline in terms of mistake severity.

Limitations. A prototype rule assigns each class a single cell under a fixed metric A. This abstraction does not allow for multi-modal classes or classes with arbitrary feature spreads, forcing the encoder to absorb this structure. $\mathcal{T}$-faithfulness requires $d \geq K - 1$ (Theorem 3), which is costly when class counts become large. Our evaluation shows that our method can decrease accuracy versus baselines on CLIP trees (§4), although it also decreases mistake severity in that setting. Our evaluation uses three datasets with under 200 classes each, measures performance across three seeds, and selects models using a single seed's validation score. Finally, our CLIP trees had weights derived from CLIP visual similarity (§E) rather than semantics.

Future work. The frozen classifier head is a differentiable function of the tree's edge weights, which could allow us to learn edge weights; learnable weights could move prototypes to whatever configuration the encoder finds easiest, so regularization or a held-out signal is necessary. Our construction only needs a Laplacian, so it could be used for arbitrary graphs; however, we could no longer read effective resistances as taxonomic distance. Finally, our method could augment post-hoc hierarchical inference procedures (Deng et al., 2012; Goren et al., 2024; Karthik et al., 2021) by ranking classes in a taxonomy-informed manner.

REFERENCES

Björn Barz and Joachim Denzler. Hierarchy-based image embeddings for semantic image retrieval. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 638–647, 2019. doi: 10.1109/WACV.2019.00073.

Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7:2399–2434, December 2006. ISSN 1532-4435.

Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A. Lord. Making better mistakes: Leveraging class hierarchies with deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12503–12512, 2020. doi: 10.1109/CVPR42600.2020.01252.

Jia Deng, Jonathan Krause, Alexander C. Berg, and Li Fei-Fei. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3450–3457, 2012. doi: 10.1109/CVPR.2012.6248086.

Karel Devriendt. Effective resistance is more than distance: Laplacians, simplices and the schur complement. *Linear Algebra and its Applications*, 639:24–49, April 2022. ISSN 0024-3795. doi: 10.1016/j.laa.2022.01.002. URL http://dx.doi.org/10.1016/j.laa.2022.01.002.

Florian Dorfler and Francesco Bullo. Kron reduction of graphs with applications to electrical networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(1):150–163, 2013. doi: 10.1109/TCSI.2012.2215780.

Shani Goren, Ido Galil, and Ran El-Yaniv. Hierarchical selective classification. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer New York, NY, 2 edition, 2009. doi: https://doi.org/10.1007/978-0-387-84858-7.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pp. 1026–1034, USA, 2015. IEEE Computer Society. ISBN 9781467383912. doi: 10.1109/ICCV.2015.123. URL https://doi.org/10.1109/ICCV.2015.123.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2012.

Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In Keh-Jiann Chen, Chu-Ren Huang, and Richard Sproat (eds.), *Proceedings of the 10th Research on Computational Linguistics International Conference*, pp. 19–33, Taipei, Taiwan, August 1997. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP). URL https://aclanthology.org/O97-1002/.

Shyamgopal Karthik, Ameya Prabhu, Puneet Kumar Dokania, and Vineet Gandhi. No cost likelihood manipulation at test time for making better mistakes in deep networks. *ArXiv*, abs/2104.00795, 2021. URL https://api.semanticscholar.org/CorpusID:233004606.

D. J. Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, 1993. ISSN 1572-8897. doi: 10.1007/BF01164627. URL https://doi.org/10.1007/BF01164627.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pp. 554–561, 2013. doi: 10.1109/ICCVW.2013.77.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images, 2009.

Gabriel Kron. Tensor analysis of networks. 1967. URL https://api.semanticscholar.org/CorpusID:118887725.

Tong Liang and Jim Davis. Inducing neural collapse to a fixed hierarchy-aware frame for reducing mistake severity, 2023. URL https://arxiv.org/abs/2303.05689.

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

Dustin G. Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features, 2020. URL https://arxiv.org/abs/2011.11619.

Vardan Papyan, X. Y. Han, and David L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020. doi: 10.1073/pnas.2015509117. URL https://www.pnas.org/doi/abs/10.1073/pnas.2015509117.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *ArXiv*, abs/2103.00020, 2021. URL https://api.semanticscholar.org/CorpusID:231591445.

Alexander J. Smola and Risi Kondor. Kernels and regularization on graphs. In Bernhard Schölkopf and Manfred K. Warmuth (eds.), *Learning Theory and Kernel Machines*, pp. 144–158, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-45167-9.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4080–4090, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Jack Valmadre. Hierarchical classification at multiple operating points. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Ziheng Yang and Bruce Rannala. Molecular phylogenetics: principles and practice. *Nature Reviews Genetics*, 13(5):303–314, May 2012. ISSN 1471-0064. doi: 10.1038/nrg3186. URL https://doi.org/10.1038/nrg3186.

Siqi Zeng, Remi Tachet des Combes, and Han Zhao. Learning structured representations by embedding class hierarchy. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7J-30ilaUZM.

Siqi Zeng, Sixian Du, Makoto Yamada, and Han Zhao. Learning structured representations by embedding class hierarchy with fast optimal transport. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=AnL6BuWzxa.

Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features, 2022. URL https://arxiv.org/abs/2203.01238.

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pp. 912–919. AAAI Press, 2003. ISBN 1577351894.

Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features, 2021. URL https://arxiv.org/abs/2105.02375.

A AI DISCLOSURE

In this work, we used generative AI tools to provide critical ingredients for proving mathematical claims, assist in the writing of proofs, formulate mathematical claims, provide feedback on research methodology and experiments, and implement methods. We have not used generative AI tools to propose or refine hypotheses, interpret results or developing theoretical models and conceptual frameworks, support qualitative or and thematic data analysis. Generating synthetic data sets, assistance with translation, cleaning and reformatting datasets are not applicable to this work. The genesis of this work was the observation that the distance-based formulation of a classifier's decision rule (NC4, §5.1) could be used with the tree Laplacian (which can be interpreted as an embedding of a tree into a metric space) to obtain a classifier that decides in terms of taxonomic distances. Since we were inspired by neural collapse, our initial attempt to realize this was through neural collapse theory. We prompted an LLM to adapt existing objectives studied in neural collapse to integrate a taxonomy's Laplacian, and it produced a long proof of a statement similar to Theorem 4 in a single attempt. The resulting statement and proof were long, vague, and incorrect in several parts; however, with human and LLM effort, we were able to modify it into an elegant and correct argument, which can be seen in its entirety in §I. Without the use of generative AI, we identified the two flaws with this approach detailed in §5.2, and leveraged these insights to develop §2's theory and §3's frozen-head training objective. We proposed §2's definitions and results ourselves, and used generative AI to suggest refinements and improvements. We also used generative AI for feedback on our experiments and methodology (§4, §G), as well as for assistance in writing code. All code was manually read and verified for correctness and fidelity to the intended methods, and we confirmed that baselines yielded similar results in our evaluations and prior evaluations whenever possible. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

B REPRODUCIBILITY

In this section, we state necessary details to reproduce our experiments (§4, §G). This section applies to all experiments except for the empirical validation of neural collapse, whose appendix contains its reproducibility information §K.1. We will release our code upon acceptance.

B.1 DATASET SPLITS

None of the three datasets we use (CIFAR-100, Aircraft, Cars) ships a native validation split. We hold out 10% of the training pool to serve this role, stratified by class. We use the exact same validation split across every experiment.

B.2 COMPARATIVE SWEEPS

Now, we describe the setup of experiments, except for the empirical realization of neural collapse (§K). For CIFAR-100, we use a ResNet-18 encoder at the resolution 32×32, using the small-image stem (3×3 stride-1 first convolution, no max-pool). For Aircraft and Cars, we use ImageNet-pretrained ResNet-50 backbones running with resolution 224×224. Aircraft images have their copyright banners cropped before any additional processing. Training augmentation is a random resized crop (area scale $[0.7, 1]$, aspect ratio $[3/4, 4/3]$), a horizontal flip with probability 0.5, and colour jitter of 0.4; evaluation resizes the shorter side and center-crops.

All models are optimized with SGD (momentum 0.9, batch size 128) with weight decay of 10^{-4} over 100 epochs, using a one-cycle schedule. We train every cell both with and without gradient norms clipped to one (the effects of this knob are documented in §G.4). We measure TreeDist@1 over the validation split after every epoch, and keep the model with the lowest (i.e., best) TreeDist@1 score. The sweeps were performed with a single seed, and the hyperparameter cell was selected from this sweep. The winning cell was then re-run with two additional seeds. Reported figures are averages over all three seeds.

Hyperparameter Grids. Cross-entropy sweeps its learning rate in $\{10^{-3}, 10^{-2}, 10^{-1}, 0.5, 1\}$. Hierarchical cross-entropy sweeps its learning rate in the same grid, and additionally sweeps its

α parameter in $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$. CPCC also uses the same learning rate grid as cross-entropy; its λ parameter is swept over $\{0.01, 0.05, 0.1, 0.5, 1, 2\}$. We found HAFrame to be highly sensitive to its learning rate. For Aircraft and Cars, we swept its learning rate over $\{0.05, 0.1, 0.5, 1\}$, while for CIFAR-100 we narrowed it to $\{0.0625, 0.125, 0.25, 0.5\}$. We swept its α parameter over the grid $\{0.3, 0.4, 0.5, 0.6\}$ and its γ parameter over the grid $\{1.0, 2.0, 3.0, 4.0, 5.0\}$, just as in the original paper. For our method, we swept the learning rate over $\{0.0625, 0.125, 0.25, 0.4\}$ and $\mathrm{NLL}_\star$ over $\{0.5, 0.7, 1.0\}$.

Our method's fixed settings. As mentioned in the main text, our method can be used with any $\mathcal{T}$-faithful prototype rule. We use M and W that minimize our neural collapse objective (§5) when using $S = L_{\mathcal{T}}$, $\lambda_W = \lambda_M = 10^{-4}$, and decision-faithful regression targets. When training with CLIP taxonomies, we multiply the edge weights so the mean leaf-to-leaf distance matches that of the corresponding uniform taxonomy.

Projection Heads. Our method adopts HAFrame's projection head (§D). We follow HAFrame in raising the learning rate for the projection head by a factor of 10, relative to the levels mentioned in the paragraph above.

Compute. We trained each model on a single NVIDIA L40S GPU.

C Graph Laplacian & Related Notions

A taxonomy is a weighted tree $\mathcal{T} = (V, E, w)$ with V the vertices, E the edges, and w positive edge weights. Label classes are $\mathcal{T}$'s leaves; internal nodes and weights are then chosen to encode relationships between classes. The *taxonomic distance* $d_{\mathcal{T}}(u, v)$ is the sum of the edge weights on the path joining u and v. f For a graph with n vertices, the Laplacian matrix $L \in \mathbb{R}^{n\times n}$ is defined as $L = D - P$, where $P \in \mathbb{R}^{n\times n}$ is the graph's adjacency matrix, where $P[u, v] = 1/w_{uv}$ if $(u, v) \in E$ and 0 otherwise; and $D \in \mathbb{R}^{n\times n}$ is diagonal with $D[u, u] = \sum_v A[u, v]$. We can eigendecompose the Laplacian as

$$L \;=\; \sum_{l=1}^{|V|} \nu_l\, \phi_l \phi_l^\top, \qquad 0 = \nu_1 < \nu_2 \le \cdots \le \nu_{|V|}, \tag{6}$$

with orthonormal eigenvectors ϕ_l and eigenvalues repeated by multiplicity. Interpreting edge weights as resistances in an electrical circuit, the *effective resistance* between u and v is

$$R_{\text{eff}}(u, v) \;=\; (e_u - e_v)^\top L^\dagger (e_u - e_v) \;=\; \sum_{l=2}^{|V|} \nu_l^{-1} \big(\phi_l(u) - \phi_l(v)\big)^2, \tag{7}$$

with $L^\dagger$ the Moore—Penrose pseudo-inverse of L and e_u, e_v standard basis vectors (Klein & Randić, 1993). In trees, the path between two nodes is unique, so resistances add in series and $R_{\text{eff}} = d_{\mathcal{T}}$.

Instances are labelled only by leaves, so the objects our analysis considers are indexed by leaves alone. We therefore eliminate the internal nodes through Kron reduction (Kron, 1967; Dorfler & Bullo, 2013; Devriendt, 2022). Partitioning L as $(\mathcal{L}, \mathcal{I})$, leaves and internal nodes, the matrix

$$L_{\mathcal{T}} \;\coloneqq\; L_{\mathcal{L}\mathcal{L}} - L_{\mathcal{L}\mathcal{I}}\, L_{\mathcal{I}\mathcal{I}}^{-1}\, L_{\mathcal{I}\mathcal{L}} \tag{8}$$

is again a graph Laplacian. Further, it preserves effective resistances (and hence taxonomic distances) between every pair of leaves.

D Projection Head

In §3, we mentioned that our objective uses an encoder backbone $\phi : \mathcal{X} \to \mathbb{R}^e$ followed by a projection head $\psi : \mathbb{R}^e \to \mathbb{R}^{K-1}$, where K is the number of leaf classes. The projection head we use is the same one used by Liang & Davis (2023). The only difference is that, since their frame lives in $\mathbb{R}^K$ instead of $\mathbb{R}^{K-1}$, the projection heads' output dimensions differ.

The projection head is a batch normalization layer followed by two residual blocks,

$$h \;=\; R_2\big(R_1(\mathrm{BN}(u))\big), \qquad R_i(x) \;=\; x + \mathrm{PReLU}\Big(\mathrm{BN}\big(L_{i,2}\,\mathrm{PReLU}(\mathrm{BN}(L_{i,1}x))\big)\Big),$$

where $L_{i,1}, L_{i,2} \in \mathbb{R}^{d\times d}$ are bias-free linear layers, and PReLU is the parametric rectified linear unit introduced by He et al. (2015). PReLU is used in place of ReLU so the coordinates of h can take negative values, necessary to produce both HAFrame's and our designated prototypes.

E Reweighting the Trees

Our experiments weight three trees two different ways, yielding a total of six taxonomies. We consider standard uniform edge weights, where every edge is weighted with "1". We also wish to consider non-uniform edge weights, allowing us to evaluate our method's performance on a larger class of taxonomies. We use the CLIP model (Radford et al., 2021) as a measure of visual similarity between classes, and allow these measurements to weight the trees in a procedure we describe here.

CLIP-Derived prototypes. We encode every image of the training splits for CIFAR-100 and Stanford-Cars, and every image of the trainval split for FGVC-aircraft, with a frozen CLIP ViT-L/14–336 image tower (Radford et al., 2021), and retain their joint-space embeddings. We first apply the same pre-processing transformations to each image: bicubic resizing to the model's input size, center cropping, and standard CLIP normalization. CIFAR-100 images are up-sampled from 32px to 336px. Every embedding is ℓ_2–normalized, then averaged within a class; the averages are once again ℓ_2–normalized, giving a unit-norm prototype for each class. This lets us derive *target distances* between every pair of classes:

$$d_{uv} \;\coloneqq\; \|m_u - m_v\|_2^2 \;=\; 2 - 2\,\langle m_u, m_v\rangle, \qquad u, v \in \mathcal{L}. \tag{9}$$

Since the characterization of §2 matches *squared* prototype distances to taxonomic distances, we square the CLIP distances so that our models can reproduce CLIP's prototype geometry approximately.

Deriving the edge weights. On a tree, the distance between two classes is the sum of the edge weights along the path separating them. Thus, the distance function is linear in the edge weights. Writing $P_{uv} \subseteq E$ for the edges on the u–v path,

$$R_{\text{eff}}(u,v) \;=\; d_{\mathcal{T}}(u,v) \;=\; \sum_{e\in E} A_{(uv),e}\, w_e, \qquad A_{(uv),e} = \mathbb{1}\{e \in \mathcal{P}(u,v)\}. \tag{10}$$

We can use this to state a nonnegative least-squares problem to be solved:

$$w^\star \;=\; \arg\min_{w\,\geq\,0} \|Aw - d\|_2^2. \tag{11}$$

Although we cannot in general reconstruct squared distances between CLIP prototypes in our trees, we settle for the best possible approximation thereof. Given that our goal is to construct irregular but plausible edge weights, the inexact squared-distance reconstruction suffices for our purposes.

F τ–$\mathrm{NLL}_\star$ Reparameterization

We use the τ and $\mathrm{NLL}_\star$ parameters of §3 throughout. We can write $\mathrm{NLL}_\star$ as a function of τ as follows:

$$\mathrm{NLL}_\star(\tau) = \frac{1}{K}\sum_k \log \sum_\ell \exp\big(-d_{\mathcal{T}}(k,\ell)/2\tau\big),$$

and hence

$$\frac{d}{d\tau}\,\mathrm{NLL}_\star(\tau) = \frac{1}{2\tau^2 K}\sum_k \mathbb{E}_{\ell\sim q_k}\big[d_{\mathcal{T}}(k,\ell)\big] > 0.$$

Thus, $\mathrm{NLL}_\star$ strictly increases from 0 (as $\tau \to 0$) to $\log K$ (as $\tau \to \infty$), and hence the map $\tau \mapsto \mathrm{NLL}_\star(\tau)$ is a bijection onto $(0, \log K)$, justifying our reparameterization.

Table 3: Classification performance on uniform taxonomies. This table complements Table 1 by including TreeDist@20 and standard deviations.

Method	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100				
CE	$75.33_{\pm 0.24}$	$0.7993_{\pm 0.0070}$	$2.6572_{\pm 0.0020}$	$3.5053_{\pm 0.0043}$
CPCC (unif.)	$74.84_{\pm 0.17}$	$0.7945_{\pm 0.0039}$	$2.0278_{\pm 0.0024}$	$3.4333_{\pm 0.0003}$
HXE (unif.)	$75.09_{\pm 0.32}$	$0.7953_{\pm 0.0105}$	$2.4207_{\pm 0.0038}$	$3.4618_{\pm 0.0034}$
HAFrame	$\mathbf{75.78}_{\pm 0.24}$	$0.7839_{\pm 0.0086}$	$2.1315_{\pm 0.0090}$	$3.4404_{\pm 0.0013}$
Ours (unif.)	$75.58_{\pm 0.20}$	$\mathbf{0.7785}_{\pm 0.0073}$	$\mathbf{1.9676}_{\pm 0.0068}$	$\mathbf{3.4270}_{\pm 0.0008}$
FGVC-Aircraft				
CE	$79.74_{\pm 0.54}$	$0.9019_{\pm 0.0238}$	$4.1738_{\pm 0.0110}$	$5.3206_{\pm 0.0050}$
CPCC (unif.)	$81.97_{\pm 0.36}$	$0.7687_{\pm 0.0219}$	$3.6360_{\pm 0.0128}$	$4.9722_{\pm 0.0033}$
HXE (unif.)	$81.14_{\pm 0.51}$	$0.7813_{\pm 0.0192}$	$4.0141_{\pm 0.0074}$	$5.2605_{\pm 0.0012}$
HAFrame	$\mathbf{82.55}_{\pm 0.35}$	$0.7339_{\pm 0.0234}$	$3.5036_{\pm 0.0138}$	$4.8969_{\pm 0.0034}$
Ours (unif.)	$\mathbf{82.55}_{\pm 0.95}$	$\mathbf{0.6731}_{\pm 0.0348}$	$\mathbf{3.4478}_{\pm 0.0127}$	$\mathbf{4.8860}_{\pm 0.0034}$
Stanford Cars				
CE	$86.64_{\pm 0.45}$	$0.4144_{\pm 0.0143}$	$2.7628_{\pm 0.0020}$	$3.5216_{\pm 0.0017}$
CPCC (unif.)	$86.42_{\pm 0.48}$	$0.4189_{\pm 0.0149}$	$2.6345_{\pm 0.0146}$	$3.4277_{\pm 0.0091}$
HXE (unif.)	$87.61_{\pm 0.38}$	$0.3717_{\pm 0.0107}$	$2.6187_{\pm 0.0056}$	$3.4655_{\pm 0.0004}$
HAFrame	$86.87_{\pm 0.16}$	$0.3952_{\pm 0.0030}$	$2.1290_{\pm 0.0046}$	$3.0825_{\pm 0.0007}$
Ours (unif.)	$\mathbf{89.57}_{\pm 0.20}$	$\mathbf{0.2968}_{\pm 0.0081}$	$\mathbf{2.0491}_{\pm 0.0037}$	$\mathbf{3.0639}_{\pm 0.0015}$

G Additional Experimental Results

G.1 Full Classification Performance

Tables 3 and 4 complete Table 1 in the main text. They include standard deviations and TreeDist@20 scores omitted from the main text.

G.2 $\mathrm{NLL}_\star$ Sensitivity Study

Our loss (§3) carries a hyperparameter τ, which we reparameterize as $\mathrm{NLL}_\star$ for comparability across datasets. Tables 5 and 6 present the effects of different values of $\mathrm{NLL}_\star$ on accuracy and severity metrics, across all six taxonomies. On all six datasets, as targets flatten (i.e., as $\mathrm{NLL}_\star \to \log K$), accuracy and TreeDist@1 suffer while TreeDist@5 and TreeDist@20 improve; however, the trends begin to reverse at the extreme ends of the grid. For mistake severity (TreeDist@$\{5, 20\}$), we see monotonic improvement for all CLIP taxonomies as $\mathrm{NLL}_\star$ increases; for the uniform taxonomies, we see some improvement but eventual losses as $\mathrm{NLL}_\star$ grows. Accuracy suffers at both extremes, peaking when $\mathrm{NLL}_\star \in [0.35, 0.7]$ across every dataset; however, this metric suffers less from the sharpest settings than the flattest. All metrics are stable in the middle of the swept range.

G.3 All Geometry Alignment Plots

In Figure 4, we reproduce Figure 3 and add the analogous plots for the remaining five taxonomies.

G.4 Impact of Gradient Norm Clipping

We found that our method can suffer performance loss and training instability, issues we address by clipping gradient norms to 1. This appendix serves to summarize the effects of gradient norm clipping on our method as well as every baseline. For the Table 1, we used the best TreeDist@1 validation-split criterion to select which hyperparameter cell to report; this criterion also chose whether we reported each method with or without gradient norm clipping. In this appendix, we use an identical selection criterion to choose the best model with and without gradient norm clipping. This lets us understand how clipping affects different methods on different benchmarks. The results are reported in Tables 7 and 8.

Table 4: Classification performance on CLIP taxonomies. This table complements Table 1 by including TreeDist@20 and standard deviations.

Method	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100				
CE	$75.31_{\pm 0.28}$	$0.0302_{\pm 0.0003}$	$0.1086_{\pm 0.0003}$	$0.1443_{\pm 0.0002}$
CPCC (unif.)	$75.35_{\pm 0.26}$	$0.0299_{\pm 0.0004}$	$0.0848_{\pm 0.0002}$	$0.1407_{\pm 0.0001}$
CPCC (CLIP)	$\mathbf{75.82}_{\pm 0.34}$	$\mathbf{0.0294}_{\pm 0.0005}$	$0.0969_{\pm 0.0002}$	$0.1371_{\pm 0.0001}$
HXE (unif.)	$75.26_{\pm 0.18}$	$0.0298_{\pm 0.0002}$	$0.0989_{\pm 0.0004}$	$0.1430_{\pm 0.0002}$
HXE (CLIP)	$74.50_{\pm 0.41}$	$0.0305_{\pm 0.0005}$	$0.1017_{\pm 0.0003}$	$0.1437_{\pm 0.0003}$
HAFrame	$75.74_{\pm 0.52}$	$0.0299_{\pm 0.0008}$	$0.0910_{\pm 0.0002}$	$0.1416_{\pm 0.0004}$
Ours (unif.)	$75.76_{\pm 0.31}$	$0.0295_{\pm 0.0005}$	$0.0828_{\pm 0.0002}$	$0.1413_{\pm 0.0001}$
Ours (CLIP)	$74.95_{\pm 0.32}$	$0.0295_{\pm 0.0005}$	$\mathbf{0.0818}_{\pm 0.0003}$	$\mathbf{0.1242}_{\pm 0.0001}$
FGVC-Aircraft				
CE	$79.61_{\pm 0.56}$	$0.0436_{\pm 0.0010}$	$0.2350_{\pm 0.0013}$	$0.3030_{\pm 0.0010}$
CPCC (unif.)	$81.38_{\pm 0.60}$	$0.0374_{\pm 0.0009}$	$0.2049_{\pm 0.0005}$	$0.2850_{\pm 0.0006}$
CPCC (CLIP)	$81.18_{\pm 0.72}$	$0.0382_{\pm 0.0024}$	$0.1959_{\pm 0.0021}$	$0.2673_{\pm 0.0007}$
HXE (unif.)	$80.74_{\pm 0.93}$	$0.0370_{\pm 0.0015}$	$0.2228_{\pm 0.0012}$	$0.2963_{\pm 0.0010}$
HXE (CLIP)	$33.71_{\pm 0.15}$	$0.1594_{\pm 0.0003}$	$0.2386_{\pm 0.0003}$	$0.2828_{\pm 0.0008}$
HAFrame	$82.48_{\pm 0.40}$	$0.0354_{\pm 0.0009}$	$0.1940_{\pm 0.0005}$	$0.2724_{\pm 0.0008}$
Ours (unif.)	$\mathbf{82.66}_{\pm 0.97}$	$0.0330_{\pm 0.0013}$	$0.1956_{\pm 0.0016}$	$0.2782_{\pm 0.0022}$
Ours (CLIP)	$81.12_{\pm 0.21}$	$\mathbf{0.0270}_{\pm 0.0017}$	$\mathbf{0.1680}_{\pm 0.0002}$	$\mathbf{0.2409}_{\pm 0.0003}$
Stanford Cars				
CE	$87.07_{\pm 0.27}$	$0.0481_{\pm 0.0010}$	$0.3354_{\pm 0.0010}$	$0.4298_{\pm 0.0005}$
CPCC (unif.)	$88.44_{\pm 0.34}$	$0.0409_{\pm 0.0008}$	$0.2722_{\pm 0.0003}$	$0.3945_{\pm 0.0004}$
CPCC (CLIP)	$87.66_{\pm 0.16}$	$0.0443_{\pm 0.0013}$	$0.2915_{\pm 0.0004}$	$0.3971_{\pm 0.0005}$
HXE (unif.)	$87.12_{\pm 0.47}$	$0.0468_{\pm 0.0014}$	$0.3178_{\pm 0.0007}$	$0.4235_{\pm 0.0002}$
HXE (CLIP)	$77.09_{\pm 0.31}$	$0.0939_{\pm 0.0015}$	$0.3238_{\pm 0.0008}$	$0.4235_{\pm 0.0003}$
HAFrame	$86.89_{\pm 0.39}$	$0.0472_{\pm 0.0013}$	$0.2689_{\pm 0.0011}$	$0.3888_{\pm 0.0008}$
Ours (unif.)	$\mathbf{88.89}_{\pm 0.48}$	$0.0391_{\pm 0.0018}$	$0.2648_{\pm 0.0011}$	$0.3905_{\pm 0.0005}$
Ours (CLIP)	$88.67_{\pm 0.82}$	$\mathbf{0.0371}_{\pm 0.0031}$	$\mathbf{0.2465}_{\pm 0.0034}$	$\mathbf{0.3630}_{\pm 0.0026}$

The tables show that gradient norm clipping can improve our method's performance, most noticeably on the Aircraft dataset. Baselines sometimes benefit from it as well.

G.5 Fitting Classifiers to Arbitrary Encoders

Our characterization theory (§2) shows that there are a large number of prototype rules that can be considered $\mathcal{T}$-faithful for a taxonomy $\mathcal{T}$. Here, we leverage Theorem 3 to fit frozen classifier heads to arbitrary encoders. We train models using cross-entropy (**CE**) and with the CPCC regularizer (**CPCC**) of Zeng et al. (2023); both these methods learn an encoder and a classifier head. Given a fully-trained model, we then compute every class's mean feature vector over training data and use those as the prototypes in Theorem 3. Using the theorem, we then compute a classifier head (W, b) to align with those prototypes to yield a $\mathcal{T}$-faithful prototype rule, and replace the learned classifier head with (W, b). In Table 9, we compare classification accuracy and mistake severity when using the *learned* and the *fitted frozen* classifier heads. We see that using the fitted frozen head consistently yields meaningful improvement in mistake severity (TreeDist@$\{5, 20\}$), but at the cost of accuracy (up to 4 points, on Stanford-Cars).

H Proof of Theorem 3

In this section, we prove Theorem 3. We begin by proving some simple lemmas, and then prove each part of the theorem in turn. As in the statement of the theorem, (M, W, A, b) denotes a prototype rule; d is the dimension of feature vectors; and $J = I_K - \frac{1}{K}\mathbb{1}\mathbb{1}^\top$ is the orthogonal projection onto $\mathbb{1}^\perp$.

We introduce further notation for the proof. For a symmetric matrix G, we let

$$\mathcal{D}(G)_{uv} := G_{uu} + G_{vv} - 2G_{uv}.$$

Table 5: Sensitivity to the target sharpness $\mathrm{NLL}_\star$ on the uniform taxonomies, at each dataset's selected learning rate. We report means over three seeds; the value selected by the sweep of §4 is bolded. That sweep only considered $\mathrm{NLL}_\star \in \{0.5, 0.7, 1.0\}$; the wider range here is measured post-hoc, although we use the learning rate selected on the smaller grid. Standard deviations are provided.

$\mathrm{NLL}_\star$	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100				
0.2	$75.70_{\pm 0.40}$	$0.7805_{\pm 0.0113}$	$2.0360_{\pm 0.0016}$	$3.4285_{\pm 0.0003}$
0.35	$75.77_{\pm 0.16}$	$0.7762_{\pm 0.0054}$	$1.9988_{\pm 0.0051}$	$3.4263_{\pm 0.0007}$
0.5	$75.78_{\pm 0.13}$	$0.7710_{\pm 0.0034}$	$1.9819_{\pm 0.0011}$	$3.4256_{\pm 0.0007}$
0.7	$75.69_{\pm 0.27}$	$0.7751_{\pm 0.0120}$	$1.9742_{\pm 0.0019}$	$3.4256_{\pm 0.0005}$
1.0	$75.34_{\pm 0.17}$	$0.7839_{\pm 0.0045}$	$1.9640_{\pm 0.0011}$	$3.4260_{\pm 0.0008}$
1.5	$74.57_{\pm 0.15}$	$0.8101_{\pm 0.0088}$	$1.9657_{\pm 0.0041}$	$3.4271_{\pm 0.0002}$
2.5	$74.04_{\pm 0.26}$	$0.8226_{\pm 0.0094}$	$1.9621_{\pm 0.0040}$	$3.4244_{\pm 0.0003}$
FGVC-Aircraft				
0.2	$82.98_{\pm 0.32}$	$0.6461_{\pm 0.0129}$	$3.5026_{\pm 0.0056}$	$4.9028_{\pm 0.0020}$
0.35	$82.75_{\pm 0.68}$	$0.6627_{\pm 0.0256}$	$3.4794_{\pm 0.0171}$	$4.8921_{\pm 0.0057}$
0.5	$82.88_{\pm 0.54}$	$0.6637_{\pm 0.0109}$	$3.4575_{\pm 0.0040}$	$4.8884_{\pm 0.0048}$
0.7	$82.99_{\pm 0.29}$	$0.6661_{\pm 0.0071}$	$3.4447_{\pm 0.0049}$	$4.8846_{\pm 0.0013}$
1.0	$82.32_{\pm 0.31}$	$0.6869_{\pm 0.0145}$	$3.4402_{\pm 0.0083}$	$4.8824_{\pm 0.0013}$
1.5	$80.87_{\pm 1.19}$	$0.7371_{\pm 0.0414}$	$3.4514_{\pm 0.0195}$	$4.8856_{\pm 0.0054}$
2.5	$77.82_{\pm 0.80}$	$0.8481_{\pm 0.0312}$	$3.4947_{\pm 0.0108}$	$4.8966_{\pm 0.0037}$
Stanford Cars				
0.2	$89.45_{\pm 0.23}$	$0.3008_{\pm 0.0056}$	$2.0808_{\pm 0.0020}$	$3.0742_{\pm 0.0034}$
0.35	$89.59_{\pm 0.32}$	$0.2968_{\pm 0.0091}$	$2.0621_{\pm 0.0043}$	$3.0688_{\pm 0.0010}$
0.5	$89.37_{\pm 0.23}$	$0.3035_{\pm 0.0057}$	$2.0565_{\pm 0.0030}$	$3.0679_{\pm 0.0004}$
0.7	$89.03_{\pm 0.41}$	$0.3172_{\pm 0.0128}$	$2.0528_{\pm 0.0081}$	$3.0686_{\pm 0.0033}$
1.0	$88.92_{\pm 0.28}$	$0.3188_{\pm 0.0092}$	$2.0435_{\pm 0.0042}$	$3.0648_{\pm 0.0020}$
1.5	$88.79_{\pm 0.29}$	$0.3213_{\pm 0.0094}$	$2.0384_{\pm 0.0026}$	$3.0621_{\pm 0.0008}$
2.5	$87.21_{\pm 0.17}$	$0.3641_{\pm 0.0059}$	$2.0518_{\pm 0.0029}$	$3.0663_{\pm 0.0021}$

H.1 Lemmas

Lemma 5. *For any symmetric matrix G,*

(i) $J\,\mathcal{D}(G)\,J = -2\,JGJ$, *and*

(ii) $\mathcal{D}(JGJ) = \mathcal{D}(G)$.

As a consequence, for G_1, G_2 symmetric, we have

$$\mathcal{D}(G_1) = \mathcal{D}(G_2) \quad \Longleftrightarrow \quad JG_1J = JG_2J.$$

Proof. We begin by noting that $\mathcal{D}$ is a linear operator, and it annihilates

$$\mathcal{K} := \left\{ a\mathbb{1}^\top + \mathbb{1}a^\top : a \in \mathbb{R}^K \right\}.$$

Indeed, for $X = a\mathbb{1}^\top + \mathbb{1}a^\top$, $X_{uv} = a_u + a_v$. Then

$$\mathcal{D}(X)_{uv} = 2a_u + 2a_v - 2(a_u + a_v) = 0.$$

Now, we prove Lemma 5(i). We can write

$$\mathcal{D}(G) = g\mathbb{1}^\top + \mathbb{1}g^\top - 2G$$

for $g := \mathrm{diag}(G)$. Since $\ker J = \mathrm{span}\{\mathbb{1}\}$, conjugation by J in the above annihilates the first two terms on the right-hand side, which yields the result.

Table 6: The same measurements as Table 5, but computed over CLIP taxonomies. Unlike that table, we multiply TreeDist@k by 100 for readability here.

$\text{NLL}_\star$	Acc. (%) ↑	TreeDist@1 $\times 100$ ↓	TreeDist@5 $\times 100$ ↓	TreeDist@20 $\times 100$ ↓
CIFAR-100				
0.2	$74.97_{\pm 0.14}$	$2.995_{\pm 0.034}$	$8.529_{\pm 0.005}$	$12.782_{\pm 0.002}$
0.35	$75.49_{\pm 0.11}$	$2.917_{\pm 0.019}$	$8.285_{\pm 0.011}$	$12.545_{\pm 0.005}$
0.5	$75.17_{\pm 0.29}$	$2.932_{\pm 0.029}$	$8.175_{\pm 0.027}$	$12.415_{\pm 0.010}$
0.7	$74.71_{\pm 0.47}$	$2.965_{\pm 0.074}$	$8.106_{\pm 0.018}$	$12.325_{\pm 0.012}$
1.0	$74.14_{\pm 0.26}$	$2.987_{\pm 0.036}$	$8.062_{\pm 0.014}$	$12.259_{\pm 0.002}$
1.5	$73.18_{\pm 0.15}$	$3.078_{\pm 0.004}$	$8.035_{\pm 0.012}$	$12.219_{\pm 0.004}$
2.5	$70.93_{\pm 0.36}$	$3.178_{\pm 0.026}$	$8.016_{\pm 0.003}$	$12.185_{\pm 0.001}$
FGVC-Aircraft				
0.2	$77.04_{\pm 0.44}$	$3.686_{\pm 0.110}$	$18.169_{\pm 0.045}$	$25.202_{\pm 0.031}$
0.35	$80.34_{\pm 0.57}$	$2.909_{\pm 0.123}$	$17.374_{\pm 0.051}$	$24.563_{\pm 0.036}$
0.5	$81.30_{\pm 0.57}$	$2.657_{\pm 0.143}$	$16.826_{\pm 0.122}$	$24.113_{\pm 0.076}$
0.7	$81.61_{\pm 0.39}$	$2.554_{\pm 0.059}$	$16.647_{\pm 0.056}$	$23.857_{\pm 0.028}$
1.0	$80.67_{\pm 0.59}$	$2.645_{\pm 0.053}$	$16.179_{\pm 0.065}$	$23.510_{\pm 0.022}$
1.5	$77.41_{\pm 0.81}$	$3.049_{\pm 0.118}$	$15.898_{\pm 0.083}$	$23.272_{\pm 0.068}$
2.5	$65.00_{\pm 0.66}$	$3.699_{\pm 0.020}$	$15.877_{\pm 0.021}$	$23.222_{\pm 0.013}$
Stanford Cars				
0.2	$88.63_{\pm 0.30}$	$3.751_{\pm 0.155}$	$25.304_{\pm 0.025}$	$37.059_{\pm 0.026}$
0.35	$89.03_{\pm 0.15}$	$3.590_{\pm 0.062}$	$24.769_{\pm 0.010}$	$36.528_{\pm 0.010}$
0.5	$88.93_{\pm 0.28}$	$3.654_{\pm 0.076}$	$24.474_{\pm 0.051}$	$36.165_{\pm 0.029}$
0.7	$88.88_{\pm 0.21}$	$3.637_{\pm 0.057}$	$24.257_{\pm 0.034}$	$35.899_{\pm 0.030}$
1.0	$87.86_{\pm 0.38}$	$4.043_{\pm 0.155}$	$24.088_{\pm 0.116}$	$35.673_{\pm 0.109}$
1.5	$87.39_{\pm 0.41}$	$4.039_{\pm 0.143}$	$23.744_{\pm 0.049}$	$35.346_{\pm 0.025}$
2.5	$86.01_{\pm 0.26}$	$4.199_{\pm 0.034}$	$23.519_{\pm 0.041}$	$35.131_{\pm 0.029}$

Next, we prove Lemma 5(ii). It will suffice to show that $JGJ - G \in \mathcal{K}$. Define $c := \frac{1}{K} G\mathbb{1}$ and $\overline{g} := \frac{1}{K^2}\mathbb{1}^\top G\mathbb{1}$. Then

$$\begin{aligned} JGJ - G &= -\mathbb{1}c^\top - c\mathbb{1}^\top + \overline{g}\mathbb{1}\mathbb{1}^\top \\ &= \left(-c + \frac{\overline{g}}{2}\mathbb{1}\right)\mathbb{1}^\top + \mathbb{1}\left(-c + \frac{\overline{g}}{2}\mathbb{1}\right)^\top \\ &\in \mathcal{K}. \end{aligned}$$

For the conclusion, there are two directions to argue. For the forward direction, conjugate both sides of the equality $\mathcal{D}(G_1) = \mathcal{D}(G_2)$ by J and apply Lemma 5(i). For the reverse, use Lemma 5(ii) for

$$\mathcal{D}(G_1) = \mathcal{D}(JG_1J) = \mathcal{D}(JG_2J) = \mathcal{D}(G_2).$$

□

Lemma 6. *Let $R_{uv} = d_{\mathcal{T}}(u, v)$. Then $R = \mathcal{D}\big(L_{\mathcal{T}}^\dagger\big)$ and $JL_{\mathcal{T}}^\dagger J = L_{\mathcal{T}}^\dagger$; hence $-\frac{1}{2}JRJ = L_{\mathcal{T}}^\dagger$.*

Proof. Using the concepts of §6, we have

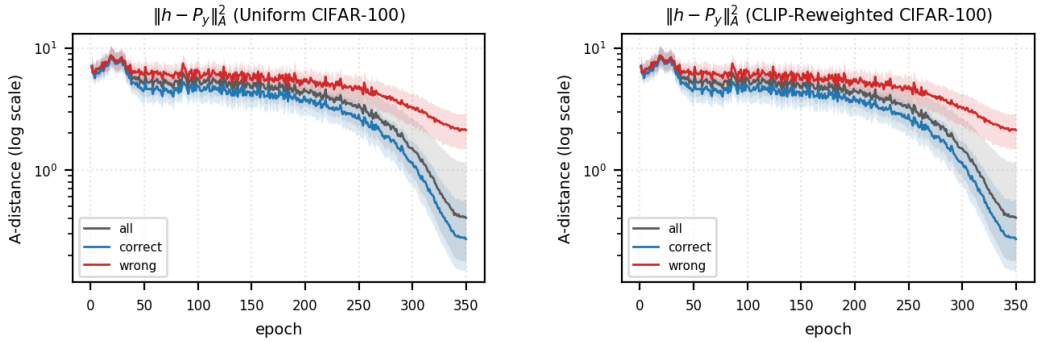
$$R_{uv} = R_{\text{eff}}(u, v) = (e_u - e_v)^\top L_{\mathcal{T}}^\dagger (e_u - e_v) = \mathcal{D}\big(L_{\mathcal{T}}^\dagger\big)_{uv}.$$

We now argue that $JL_{\mathcal{T}}^\dagger J = L_{\mathcal{T}}^\dagger$. Since $L_{\mathcal{T}}$ is the Laplacian of a connected graph, we have

$$\ker L_{\mathcal{T}} = \ker L_{\mathcal{T}}^\dagger = \text{span}\{\mathbb{1}\}.$$

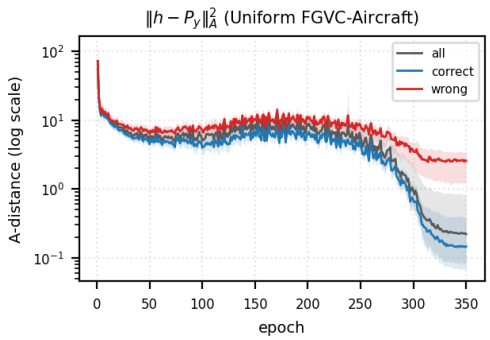
Fixing an eigendecomposition of $L_{\mathcal{T}}$ as in §6, we have $0 = \nu_1 < \nu_2$, and ϕ_1 spans the kernel; thus, we have

$$L_{\mathcal{T}}^\dagger = \sum_{l=2}^{K} \nu_l^{-1} \phi_l \phi_l^\top, \qquad J = \sum_{l=2}^{K} \phi_l \phi_l^\top,$$

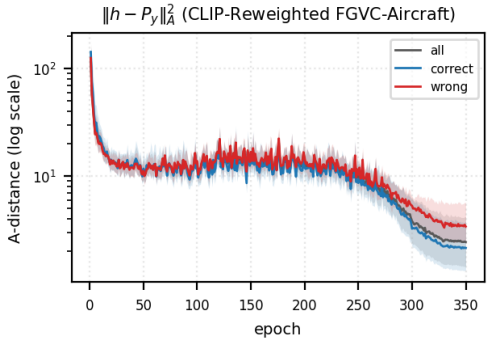


(a) CIFAR-100, uniform

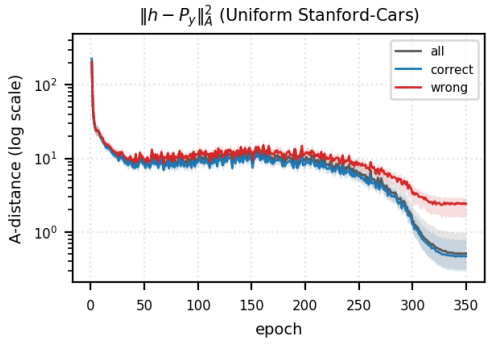
(b) CIFAR-100, reweighted



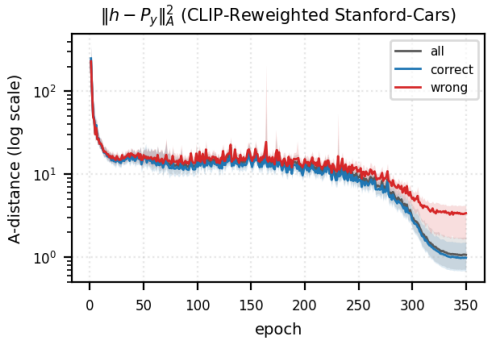
(c) FGVC-Aircraft, uniform



(d) FGVC-Aircraft, reweighted



(e) Stanford Cars, uniform



(f) Stanford Cars, reweighted

Figure 4: A-distance from validation samples to their class's prototypes, computed once per epoch. Solid lines are medians; shaded regions are IQRs.

the latter holding since J projects orthogonally onto $\mathbb{1}^\perp = \mathrm{span}\{\phi_2, \ldots, \phi_K\}$. Orthonormality of the eigenbasis gives $J\phi_l = \phi_l$ for $l \geq 2$; hence $JL_{\mathcal{T}}^\dagger J = L_{\mathcal{T}}^\dagger$.

To conclude the proof, apply Lemma 5(i) for $L_{\mathcal{T}}^\dagger = -\frac{1}{2}JRJ$. □

Lemma 7. *Let (M, W, A, b) be a prototype rule and let $G = WM^\top$. Then G is symmetric, $G_{uv} = \langle m_u, m_v \rangle_A$, and $\mathcal{D}(G)_{uv} = \|m_u - m_v\|_A^2$.*

Proof. The definition of a prototype rule and symmetry of A give

$$w_k = Am_k \iff w_k^\top = m_k^\top A \iff W = MA.$$

Table 7: Effect of gradient-norm clipping on the uniform taxonomies. Each method appears with clipping off and on; hyperparameters were re-selected for each setting on validation TreeDist@1. The better setting per method and metric is bolded. Means and standard deviations over three seeds on the test split.

Method	Clip	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100					
CE	off	$\mathbf{75.40}_{\pm 0.10}$	$0.8048_{\pm 0.0020}$	$2.7418_{\pm 0.0038}$	$3.5290_{\pm 0.0016}$
	on	$75.33_{\pm 0.24}$	$\mathbf{0.7993}_{\pm 0.0070}$	$\mathbf{2.6572}_{\pm 0.0020}$	$\mathbf{3.5053}_{\pm 0.0043}$
CPCC (unif.)	off	$74.84_{\pm 0.17}$	$0.7945_{\pm 0.0039}$	$\mathbf{2.0278}_{\pm 0.0024}$	$3.4333_{\pm 0.0003}$
	on	$\mathbf{75.86}_{\pm 0.06}$	$\mathbf{0.7705}_{\pm 0.0024}$	$2.0465_{\pm 0.0017}$	$\mathbf{3.4298}_{\pm 0.0009}$
HXE (unif.)	off	$\mathbf{75.57}_{\pm 0.35}$	$\mathbf{0.7859}_{\pm 0.0112}$	$2.6113_{\pm 0.0071}$	$3.4933_{\pm 0.0005}$
	on	$75.09_{\pm 0.32}$	$0.7953_{\pm 0.0105}$	$\mathbf{2.4207}_{\pm 0.0038}$	$\mathbf{3.4618}_{\pm 0.0034}$
HAFrame	off	$75.23_{\pm 0.33}$	$0.8032_{\pm 0.0101}$	$\mathbf{2.0846}_{\pm 0.0033}$	$\mathbf{3.4380}_{\pm 0.0004}$
	on	$\mathbf{75.78}_{\pm 0.24}$	$\mathbf{0.7839}_{\pm 0.0086}$	$2.1315_{\pm 0.0090}$	$3.4404_{\pm 0.0013}$
Ours (unif.)	off	$75.58_{\pm 0.20}$	$0.7785_{\pm 0.0073}$	$\mathbf{1.9676}_{\pm 0.0068}$	$3.4270_{\pm 0.0008}$
	on	$\mathbf{75.78}_{\pm 0.13}$	$\mathbf{0.7735}_{\pm 0.0066}$	$1.9816_{\pm 0.0029}$	$\mathbf{3.4253}_{\pm 0.0011}$
FGVC-Aircraft					
CE	off	$79.74_{\pm 0.54}$	$0.9019_{\pm 0.0238}$	$4.1738_{\pm 0.0110}$	$5.3206_{\pm 0.0050}$
	on	$\mathbf{80.59}_{\pm 0.31}$	$\mathbf{0.8481}_{\pm 0.0156}$	$\mathbf{4.1604}_{\pm 0.0099}$	$\mathbf{5.3152}_{\pm 0.0026}$
CPCC (unif.)	off	$81.49_{\pm 0.31}$	$0.7777_{\pm 0.0160}$	$3.7318_{\pm 0.0070}$	$5.0190_{\pm 0.0056}$
	on	$\mathbf{81.97}_{\pm 0.36}$	$\mathbf{0.7687}_{\pm 0.0219}$	$\mathbf{3.6360}_{\pm 0.0128}$	$\mathbf{4.9722}_{\pm 0.0033}$
HXE (unif.)	off	$\mathbf{81.14}_{\pm 0.51}$	$\mathbf{0.7813}_{\pm 0.0192}$	$4.0141_{\pm 0.0074}$	$5.2605_{\pm 0.0012}$
	on	$80.34_{\pm 0.15}$	$0.7981_{\pm 0.0037}$	$\mathbf{3.9646}_{\pm 0.0068}$	$\mathbf{5.2383}_{\pm 0.0028}$
HAFrame	off	$80.98_{\pm 0.94}$	$0.7963_{\pm 0.0437}$	$3.5240_{\pm 0.0098}$	$4.9052_{\pm 0.0058}$
	on	$\mathbf{82.55}_{\pm 0.35}$	$\mathbf{0.7339}_{\pm 0.0234}$	$\mathbf{3.5036}_{\pm 0.0138}$	$\mathbf{4.8969}_{\pm 0.0034}$
Ours (unif.)	off	$81.86_{\pm 0.41}$	$0.7123_{\pm 0.0146}$	$\mathbf{3.4466}_{\pm 0.0055}$	$\mathbf{4.8833}_{\pm 0.0018}$
	on	$\mathbf{82.55}_{\pm 0.95}$	$\mathbf{0.6731}_{\pm 0.0348}$	$3.4478_{\pm 0.0127}$	$4.8860_{\pm 0.0034}$
Stanford Cars					
CE	off	$86.64_{\pm 0.45}$	$0.4144_{\pm 0.0143}$	$2.7628_{\pm 0.0020}$	$3.5216_{\pm 0.0017}$
	on	$\mathbf{86.89}_{\pm 0.28}$	$\mathbf{0.4040}_{\pm 0.0083}$	$\mathbf{2.7546}_{\pm 0.0057}$	$\mathbf{3.5187}_{\pm 0.0012}$
CPCC (unif.)	off	$\mathbf{86.42}_{\pm 0.48}$	$\mathbf{0.4189}_{\pm 0.0149}$	$2.6345_{\pm 0.0146}$	$3.4277_{\pm 0.0091}$
	on	$85.78_{\pm 0.14}$	$0.4197_{\pm 0.0031}$	$\mathbf{2.1578}_{\pm 0.0039}$	$\mathbf{3.1397}_{\pm 0.0016}$
HXE (unif.)	off	$87.58_{\pm 0.34}$	$\mathbf{0.3678}_{\pm 0.0109}$	$\mathbf{2.5989}_{\pm 0.0030}$	$\mathbf{3.4579}_{\pm 0.0004}$
	on	$\mathbf{87.61}_{\pm 0.38}$	$0.3717_{\pm 0.0107}$	$2.6187_{\pm 0.0056}$	$3.4655_{\pm 0.0004}$
HAFrame	off	$86.87_{\pm 0.16}$	$0.3952_{\pm 0.0030}$	$2.1290_{\pm 0.0046}$	$3.0825_{\pm 0.0007}$
	on	$\mathbf{87.28}_{\pm 0.12}$	$\mathbf{0.3823}_{\pm 0.0052}$	$\mathbf{2.0891}_{\pm 0.0013}$	$\mathbf{3.0732}_{\pm 0.0014}$
Ours (unif.)	off	$89.07_{\pm 0.22}$	$0.3149_{\pm 0.0069}$	$2.0583_{\pm 0.0010}$	$3.0682_{\pm 0.0017}$
	on	$\mathbf{89.57}_{\pm 0.20}$	$\mathbf{0.2968}_{\pm 0.0081}$	$\mathbf{2.0491}_{\pm 0.0037}$	$\mathbf{3.0639}_{\pm 0.0015}$

Then $G = WM^\top = MAM^\top$ is symmetric, with $G_{uv} = m_u^\top A m_v = \langle m_u, m_v \rangle_A$. Finally, note that

$$\begin{aligned} \|m_u - m_v\|_A^2 &= (m_u - m_v)^\top A (m_u - m_v) \\ &= m_u^\top A m_u + m_v^\top A m_v - 2 m_u^\top A m_v \\ &= G_{uu} + G_{vv} - 2G_{uv} \\ &= \mathcal{D}(G)_{uv}. \end{aligned}$$

□

H.2 Proof of Theorem 3(i)

(M, W, A, b) is $\mathcal{T}$-faithful precisely when $\|m_u - m_v\|_A^2 = d_{\mathcal{T}}(u, v)$ for every pair of classes u, v. Lemma 7 identifies the left-hand side as equal to $\mathcal{D}(G)_{uv}$; $R_{uv} = d_{\mathcal{T}}(u, v)$ by definition, so $\mathcal{T}$-faithfulness is equivalent to the matrix equality $\mathcal{D}(G) = R$. Lemma 6 then tells us that $R = \mathcal{D}(L_{\mathcal{T}}^\dagger)$, so the matrix equality is equivalent to $\mathcal{D}(L_{\mathcal{T}}^\dagger) = \mathcal{D}(G)$. Lemma 5 then tells us that the latter matrix equality is equivalent to

$$L_{\mathcal{T}}^\dagger = J L_{\mathcal{T}}^\dagger J = JGJ = JWM^\top J,$$

completing the proof.

Table 8: The same measurements as Table 7, on the CLIP taxonomies.

Method	Clip	Acc. (%) ↑	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100					
CE	off	$\mathbf{75.33}_{\pm 0.19}$	$0.0307_{\pm 0.0002}$	$0.1117_{\pm 0.0004}$	$0.1453_{\pm 0.0002}$
	on	$75.31_{\pm 0.28}$	$\mathbf{0.0302}_{\pm 0.0003}$	$\mathbf{0.1086}_{\pm 0.0003}$	$\mathbf{0.1443}_{\pm 0.0002}$
CPCC (unif.)	off	$\mathbf{75.61}_{\pm 0.13}$	$\mathbf{0.0298}_{\pm 0.0001}$	$0.0989_{\pm 0.0006}$	$0.1425_{\pm 0.0002}$
	on	$75.35_{\pm 0.26}$	$0.0299_{\pm 0.0004}$	$\mathbf{0.0848}_{\pm 0.0002}$	$\mathbf{0.1407}_{\pm 0.0001}$
CPCC (CLIP)	off	$\mathbf{75.82}_{\pm 0.34}$	$\mathbf{0.0294}_{\pm 0.0005}$	$0.0969_{\pm 0.0002}$	$0.1371_{\pm 0.0001}$
	on	$75.33_{\pm 0.16}$	$0.0297_{\pm 0.0001}$	$\mathbf{0.0900}_{\pm 0.0003}$	$\mathbf{0.1348}_{\pm 0.0001}$
HXE (unif.)	off	$\mathbf{75.27}_{\pm 0.32}$	$0.0302_{\pm 0.0004}$	$0.1048_{\pm 0.0003}$	$0.1432_{\pm 0.0002}$
	on	$75.26_{\pm 0.18}$	$\mathbf{0.0298}_{\pm 0.0002}$	$\mathbf{0.0989}_{\pm 0.0004}$	$\mathbf{0.1430}_{\pm 0.0002}$
HXE (CLIP)	off	$\mathbf{74.77}_{\pm 0.21}$	$0.0309_{\pm 0.0006}$	$0.1072_{\pm 0.0002}$	$\mathbf{0.1437}_{\pm 0.0003}$
	on	$74.50_{\pm 0.41}$	$\mathbf{0.0305}_{\pm 0.0005}$	$\mathbf{0.1017}_{\pm 0.0003}$	$\mathbf{0.1437}_{\pm 0.0003}$
HAFrame	off	$75.54_{\pm 0.47}$	$\mathbf{0.0297}_{\pm 0.0004}$	$\mathbf{0.0837}_{\pm 0.0002}$	$\mathbf{0.1412}_{\pm 0.0008}$
	on	$\mathbf{75.74}_{\pm 0.52}$	$0.0299_{\pm 0.0008}$	$0.0910_{\pm 0.0002}$	$0.1416_{\pm 0.0004}$
Ours (unif.)	off	$\mathbf{76.26}_{\pm 0.30}$	$\mathbf{0.0288}_{\pm 0.0004}$	$\mathbf{0.0828}_{\pm 0.0001}$	$0.1414_{\pm 0.0003}$
	on	$75.76_{\pm 0.31}$	$0.0295_{\pm 0.0005}$	$\mathbf{0.0828}_{\pm 0.0002}$	$\mathbf{0.1413}_{\pm 0.0001}$
Ours (CLIP)	off	$74.63_{\pm 0.41}$	$0.0300_{\pm 0.0005}$	$0.0824_{\pm 0.0002}$	$0.1246_{\pm 0.0001}$
	on	$\mathbf{74.95}_{\pm 0.32}$	$\mathbf{0.0295}_{\pm 0.0005}$	$\mathbf{0.0818}_{\pm 0.0003}$	$\mathbf{0.1242}_{\pm 0.0001}$
FGVC-Aircraft					
CE	off	$78.16_{\pm 0.39}$	$0.0441_{\pm 0.0005}$	$0.2380_{\pm 0.0003}$	$\mathbf{0.3024}_{\pm 0.0002}$
	on	$\mathbf{79.61}_{\pm 0.56}$	$\mathbf{0.0436}_{\pm 0.0010}$	$\mathbf{0.2350}_{\pm 0.0013}$	$0.3030_{\pm 0.0010}$
CPCC (unif.)	off	$81.38_{\pm 0.60}$	$0.0374_{\pm 0.0009}$	$0.2049_{\pm 0.0005}$	$\mathbf{0.2850}_{\pm 0.0006}$
	on	$\mathbf{82.47}_{\pm 0.34}$	$\mathbf{0.0357}_{\pm 0.0013}$	$\mathbf{0.2015}_{\pm 0.0006}$	$\mathbf{0.2850}_{\pm 0.0002}$
CPCC (CLIP)	off	$79.67_{\pm 0.39}$	$0.0420_{\pm 0.0006}$	$0.2296_{\pm 0.0008}$	$0.2932_{\pm 0.0008}$
	on	$\mathbf{81.18}_{\pm 0.72}$	$\mathbf{0.0382}_{\pm 0.0024}$	$\mathbf{0.1959}_{\pm 0.0021}$	$\mathbf{0.2673}_{\pm 0.0007}$
HXE (unif.)	off	$80.74_{\pm 0.93}$	$\mathbf{0.0370}_{\pm 0.0015}$	$\mathbf{0.2228}_{\pm 0.0012}$	$\mathbf{0.2963}_{\pm 0.0010}$
	on	$\mathbf{81.01}_{\pm 0.61}$	$0.0382_{\pm 0.0006}$	$0.2272_{\pm 0.0002}$	$0.2985_{\pm 0.0004}$
HXE (CLIP)	off	$33.71_{\pm 0.15}$	$\mathbf{0.1594}_{\pm 0.0003}$	$\mathbf{0.2386}_{\pm 0.0003}$	$0.2828_{\pm 0.0008}$
	on	$\mathbf{35.03}_{\pm 0.37}$	$0.1604_{\pm 0.0008}$	$0.2414_{\pm 0.0020}$	$\mathbf{0.2813}_{\pm 0.0009}$
HAFrame	off	$80.98_{\pm 1.34}$	$0.0385_{\pm 0.0023}$	$0.1967_{\pm 0.0031}$	$0.2743_{\pm 0.0027}$
	on	$\mathbf{82.48}_{\pm 0.40}$	$\mathbf{0.0354}_{\pm 0.0009}$	$\mathbf{0.1940}_{\pm 0.0005}$	$\mathbf{0.2724}_{\pm 0.0008}$
Ours (unif.)	off	$81.94_{\pm 0.18}$	$\mathbf{0.0330}_{\pm 0.0011}$	$\mathbf{0.1954}_{\pm 0.0011}$	$\mathbf{0.2776}_{\pm 0.0010}$
	on	$\mathbf{82.66}_{\pm 0.97}$	$\mathbf{0.0330}_{\pm 0.0013}$	$0.1956_{\pm 0.0016}$	$0.2782_{\pm 0.0022}$
Ours (CLIP)	off	$79.06_{\pm 1.04}$	$0.0318_{\pm 0.0017}$	$0.1688_{\pm 0.0024}$	$\mathbf{0.2396}_{\pm 0.0012}$
	on	$\mathbf{81.12}_{\pm 0.21}$	$\mathbf{0.0270}_{\pm 0.0017}$	$\mathbf{0.1680}_{\pm 0.0002}$	$0.2409_{\pm 0.0003}$
Stanford Cars					
CE	off	$86.29_{\pm 0.34}$	$0.0515_{\pm 0.0009}$	$0.3368_{\pm 0.0008}$	$0.4301_{\pm 0.0006}$
	on	$\mathbf{87.07}_{\pm 0.27}$	$\mathbf{0.0481}_{\pm 0.0010}$	$\mathbf{0.3354}_{\pm 0.0010}$	$\mathbf{0.4298}_{\pm 0.0005}$
CPCC (unif.)	off	$87.38_{\pm 0.48}$	$0.0459_{\pm 0.0014}$	$0.2873_{\pm 0.0002}$	$0.3990_{\pm 0.0005}$
	on	$\mathbf{88.44}_{\pm 0.34}$	$\mathbf{0.0409}_{\pm 0.0008}$	$\mathbf{0.2722}_{\pm 0.0003}$	$\mathbf{0.3945}_{\pm 0.0004}$
CPCC (CLIP)	off	$87.24_{\pm 0.34}$	$0.0459_{\pm 0.0012}$	$\mathbf{0.2886}_{\pm 0.0008}$	$\mathbf{0.3934}_{\pm 0.0006}$
	on	$\mathbf{87.66}_{\pm 0.16}$	$\mathbf{0.0443}_{\pm 0.0013}$	$0.2915_{\pm 0.0004}$	$0.3971_{\pm 0.0005}$
HXE (unif.)	off	$\mathbf{87.65}_{\pm 0.51}$	$\mathbf{0.0444}_{\pm 0.0024}$	$0.3179_{\pm 0.0008}$	$0.4237_{\pm 0.0002}$
	on	$87.12_{\pm 0.47}$	$0.0468_{\pm 0.0014}$	$\mathbf{0.3178}_{\pm 0.0007}$	$\mathbf{0.4235}_{\pm 0.0002}$
HXE (CLIP)	off	$77.09_{\pm 0.31}$	$0.0939_{\pm 0.0015}$	$\mathbf{0.3238}_{\pm 0.0008}$	$\mathbf{0.4235}_{\pm 0.0003}$
	on	$\mathbf{77.64}_{\pm 0.23}$	$\mathbf{0.0935}_{\pm 0.0005}$	$0.3311_{\pm 0.0003}$	$0.4270_{\pm 0.0002}$
HAFrame	off	$\mathbf{86.89}_{\pm 0.39}$	$\mathbf{0.0472}_{\pm 0.0013}$	$\mathbf{0.2689}_{\pm 0.0011}$	$\mathbf{0.3888}_{\pm 0.0008}$
	on	$86.41_{\pm 0.10}$	$0.0488_{\pm 0.0005}$	$0.2840_{\pm 0.0002}$	$0.3938_{\pm 0.0002}$
Ours (unif.)	off	$\mathbf{88.89}_{\pm 0.48}$	$\mathbf{0.0391}_{\pm 0.0018}$	$0.2648_{\pm 0.0011}$	$0.3905_{\pm 0.0005}$
	on	$88.67_{\pm 0.09}$	$0.0397_{\pm 0.0005}$	$\mathbf{0.2631}_{\pm 0.0005}$	$\mathbf{0.3902}_{\pm 0.0004}$
Ours (CLIP)	off	$88.19_{\pm 0.29}$	$0.0392_{\pm 0.0009}$	$\mathbf{0.2402}_{\pm 0.0013}$	$\mathbf{0.3563}_{\pm 0.0008}$
	on	$\mathbf{88.67}_{\pm 0.82}$	$\mathbf{0.0371}_{\pm 0.0031}$	$0.2465_{\pm 0.0034}$	$0.3630_{\pm 0.0026}$

H.3 Proof of Theorem 3(ii)

Write $N = JM$, whose k-th row is the centered prototype $m_k - \overline{m}$ for $\overline{m} = \frac{1}{K}\sum_k m_k$.

Start with the forward direction. If (M, W, A, b) is $\mathcal{T}$-faithful, then $L_{\mathcal{T}}^{\dagger} = NAN^{\top}$. Thus

$$K - 1 = \operatorname{rank}\big(L_{\mathcal{T}}^{\dagger}\big) \le \operatorname{rank}(N) \le \operatorname{rank}(J) = K - 1;$$

Table 9: Comparing classification accuracy and mistake severity on CLIP taxonomies, using *learned* versus *fitted frozen* heads.

Loss	Head	Acc. (%) ↑	Bal. acc. (%) ↑	CE loss	TreeDist@1 ↓	TreeDist@5 ↓	TreeDist@20 ↓
CIFAR-100							
CE	Learned	$75.33_{\pm 0.19}$	$75.33_{\pm 0.19}$	$0.988_{\pm 0.007}$	$0.0307_{\pm 0.0002}$	$0.1117_{\pm 0.0004}$	$0.1453_{\pm 0.0002}$
CE	Fitted frozen	$72.63_{\pm 0.40}$	$72.63_{\pm 0.40}$	$4.543_{\pm 0.000}$	$0.0320_{\pm 0.0004}$	$0.0847_{\pm 0.0001}$	$0.1259_{\pm 0.0000}$
CPCC (Unif.)	Learned	$75.61_{\pm 0.13}$	$75.61_{\pm 0.13}$	$0.960_{\pm 0.008}$	$0.0298_{\pm 0.0001}$	$0.0989_{\pm 0.0006}$	$0.1425_{\pm 0.0002}$
CPCC (Unif.)	Fitted frozen	$73.18_{\pm 0.11}$	$73.18_{\pm 0.11}$	$4.543_{\pm 0.000}$	$0.0311_{\pm 0.0003}$	$0.0838_{\pm 0.0001}$	$0.1248_{\pm 0.0000}$
CPCC (CLIP)	Learned	$75.82_{\pm 0.34}$	$75.82_{\pm 0.34}$	$0.956_{\pm 0.005}$	$0.0294_{\pm 0.0005}$	$0.0969_{\pm 0.0002}$	$0.1371_{\pm 0.0001}$
CPCC (CLIP)	Fitted frozen	$74.06_{\pm 0.22}$	$74.06_{\pm 0.22}$	$4.544_{\pm 0.000}$	$0.0300_{\pm 0.0002}$	$0.0832_{\pm 0.0003}$	$0.1249_{\pm 0.0001}$
FGVC-Aircraft							
CE	Learned	$78.16_{\pm 0.39}$	$78.15_{\pm 0.39}$	$0.834_{\pm 0.004}$	$0.0441_{\pm 0.0005}$	$0.2380_{\pm 0.0003}$	$0.3024_{\pm 0.0002}$
CE	Fitted frozen	$74.85_{\pm 0.10}$	$74.83_{\pm 0.10}$	$4.469_{\pm 0.000}$	$0.0477_{\pm 0.0004}$	$0.1698_{\pm 0.0002}$	$0.2386_{\pm 0.0000}$
CPCC (Unif.)	Learned	$81.38_{\pm 0.60}$	$81.35_{\pm 0.61}$	$0.733_{\pm 0.008}$	$0.0374_{\pm 0.0009}$	$0.2049_{\pm 0.0005}$	$0.2850_{\pm 0.0006}$
CPCC (Unif.)	Fitted frozen	$79.10_{\pm 0.32}$	$79.08_{\pm 0.32}$	$4.467_{\pm 0.000}$	$0.0376_{\pm 0.0001}$	$0.1613_{\pm 0.0002}$	$0.2335_{\pm 0.0002}$
CPCC (CLIP)	Learned	$79.67_{\pm 0.39}$	$79.65_{\pm 0.40}$	$0.791_{\pm 0.008}$	$0.0420_{\pm 0.0006}$	$0.2296_{\pm 0.0008}$	$0.2932_{\pm 0.0008}$
CPCC (CLIP)	Fitted frozen	$76.23_{\pm 0.30}$	$76.22_{\pm 0.31}$	$4.473_{\pm 0.000}$	$0.0449_{\pm 0.0004}$	$0.1695_{\pm 0.0006}$	$0.2388_{\pm 0.0002}$
Stanford Cars							
CE	Learned	$86.29_{\pm 0.34}$	$86.09_{\pm 0.36}$	$0.526_{\pm 0.011}$	$0.0515_{\pm 0.0009}$	$0.3368_{\pm 0.0008}$	$0.4301_{\pm 0.0006}$
CE	Fitted frozen	$81.28_{\pm 0.46}$	$81.16_{\pm 0.46}$	$5.091_{\pm 0.001}$	$0.0685_{\pm 0.0010}$	$0.2594_{\pm 0.0002}$	$0.3693_{\pm 0.0001}$
CPCC (Unif.)	Learned	$87.38_{\pm 0.48}$	$87.21_{\pm 0.46}$	$0.483_{\pm 0.008}$	$0.0459_{\pm 0.0014}$	$0.2873_{\pm 0.0002}$	$0.3990_{\pm 0.0005}$
CPCC (Unif.)	Fitted frozen	$84.94_{\pm 0.68}$	$84.85_{\pm 0.67}$	$5.088_{\pm 0.000}$	$0.0519_{\pm 0.0025}$	$0.2456_{\pm 0.0005}$	$0.3576_{\pm 0.0005}$
CPCC (CLIP)	Learned	$87.24_{\pm 0.34}$	$87.09_{\pm 0.31}$	$0.493_{\pm 0.009}$	$0.0459_{\pm 0.0012}$	$0.2886_{\pm 0.0008}$	$0.3934_{\pm 0.0006}$
CPCC (CLIP)	Fitted frozen	$84.04_{\pm 0.29}$	$83.91_{\pm 0.29}$	$5.088_{\pm 0.000}$	$0.0546_{\pm 0.0004}$	$0.2477_{\pm 0.0002}$	$0.3598_{\pm 0.0001}$

$\operatorname{rank}\big(L_{\mathcal{T}}^{\dagger}\big) = K-1$ holds since the Kron-reduced $L_{\mathcal{T}}^{\dagger}$ is the Laplacian of a connected graph (Dorfler & Bullo, 2013), so $\operatorname{rank}(N) = K-1$.

Now, consider the reverse direction. Say $\operatorname{rank}(N) = K-1$, and write $N = U\Sigma V^{\top}$ for a singular value decomposition (SVD), where $U \in \mathbb{R}^{K\times(K-1)}$, $V \in \mathbb{R}^{d\times(K-1)}$ have orthonormal columns and $\Sigma \succ 0$ is diagonal. Since $\mathbb{1}^{\top}N = \mathbb{1}^{\top}JM = 0$, $\operatorname{col}(N) \subseteq \mathbb{1}^{\perp}$; both spaces have dimension $K-1$, so in fact $\operatorname{col}(U) = \operatorname{col}(N) = \mathbb{1}^{\perp}$. Since $UU^{\top}$ is the orthogonal projection onto $\operatorname{col}(U)$, $UU^{\top} = J$. By Lemma 6's spectral form, $L_{\mathcal{T}}^{\dagger}$ is positive-definite on $\mathbb{1}^{\perp}$; thus $U^{\top}L_{\mathcal{T}}^{\dagger}U \succ 0$ and hence

$$C := \Sigma^{-1}\big(U^{\top}L_{\mathcal{T}}^{\dagger}U\big)\Sigma^{-1} \succ 0.$$

N's SVD gives us an expression for its pseudo-inverse: $N^{\dagger} = V\Sigma^{-1}U^{\top}$. Now, fix $\alpha > 0$; using $N^{\dagger} = V\Sigma^{-1}U^{\top}$ and the definition of C, write

$$A := VCV^{\top} + \alpha\big(I_d - VV^{\top}\big) = N^{\dagger}L_{\mathcal{T}}^{\dagger}(N^{\dagger})^{\top} + \alpha(I_d - VV^{\top}).$$

A is symmetric and, in fact, positive-definite: it is block-diagonal for the splitting $\mathbb{R}^d = \operatorname{col}(V) \oplus \operatorname{col}(V)^{\perp}$, acting as $C \succ 0$ on $\operatorname{col}(V)$ and $\alpha I \succ 0$ on $\operatorname{col}(V)^{\perp}$.

Now, we have

$$\begin{aligned}
NAN^{\top} &= (U\Sigma V^{\top})\left[VCV^{\top} + \alpha(I_d - VV^{\top})\right](V\Sigma U^{\top}) \\
&= U\Sigma V^{\top}VCV^{\top}V\Sigma U^{\top} + \alpha U\Sigma(V^{\top}V - V^{\top}VV^{\top}V)\Sigma U^{\top} \\
&= U\Sigma C\Sigma U^{\top} \\
&= UU^{\top}L_{\mathcal{T}}^{\dagger}UU^{\top} \\
&= JL_{\mathcal{T}}^{\dagger}J \\
&= L_{\mathcal{T}}^{\dagger}
\end{aligned}$$

where the last step follows from Lemma 6.

Now, take $W = MA$ and $b_k = -\frac{1}{2}m_k^{\top}Am_k$. Then (M, W, A, b) is a prototype rule with $JWM^{\top}J = NAN^{\top} = L_{\mathcal{T}}^{\dagger}$, hence $\mathcal{T}$-faithful by §H.2.

I Proof of Theorem 4

This appendix proves Theorem 4. We first prove a more general statement, Proposition 8, which describes the minimizers of equation 4 in closed form for any $d \geq K-1$. Theorem 4 is read off as

a corollary in §I.9. The structure of the proof is a series of reductions from one objective to another; we descend to an objective whose unique minimizer we describe in closed form; we then ascend back to the original objective, characterizing the residual non-uniqueness of its minimizers as we do. §I.3 shows that a minimizer exists; §I.4 solves for W in closed form, leaving an optimization over the Gram $G = MM^\top$; §I.5 shows that the optimal Gram G is scalar on each $L_\mathcal{T}$-eigenspace, decoupling the optimization problem into independent, mode-by-mode scalar problems; §I.6 solves the per-mode scalar problems. §I.7 shows that the optimal Gram is unique. We then move back up to the original objective in §I.9, recovering a complete description of optimal (M^*, W^*) from the unique optimal Gram.

I.1 Conventions & Notation

Throughout all of our appendices, $L_\mathcal{T} = U\Lambda U^\top$ is the eigendecomposition of $L_\mathcal{T}$ in some orthonormal eigenbasis U, where $\Lambda = \operatorname{diag}(0, \nu_2, \ldots, \nu_K)$ with $0 < \nu_2 \leq \cdots \leq \nu_K$. We write ϕ_l for the l-th column of U, matching the eigenbasis fixed in equation 6. The regularizer $S \in \{I, L_\mathcal{T}\}$ is diagonalized by the same basis,

$$S \;=\; U\Lambda_S U^\top, \qquad \Lambda_S = \operatorname{diag}(s_1, \ldots, s_K),$$

with $s_l = 1$ for $S = I$ and $s_l = \nu_l$ for $S = L_\mathcal{T}$. The only properties of S we use are that $s_l > 0$ for every $l \geq 2$ and that $s_l = s_{l'}$ whenever $\nu_l = \nu_{l'}$; we write $s_{\min} := \min_{l \geq 2} s_l > 0$. We assume $d \geq K - 1$ throughout this appendix; Theorem 4 takes $d = K - 1$. We assume that Y acts as a scalar times the identity on each of $L_\mathcal{T}$'s eigenspaces. This condition means that we can write for U (and any other eigenbasis of $L_\mathcal{T}$)

$$Y \;=\; U\Gamma U^\top, \qquad \Gamma \;=\; \operatorname{diag}(\gamma_1, \ldots, \gamma_K)$$

where $\gamma_l \in \mathbb{R}$ for all l, and $\gamma_l = \gamma_{l'}$ whenever $\nu_l = \nu_{l'}$. To simplify its presentation, §5 required $\gamma_1 = 0$ and $\gamma_l \geq 0$. We eliminate those assumptions here to prove a more general statement. Then $YY^\top = U\Gamma^2 U^\top$ shares the same block structure, with diagonal entries $\gamma_l^2 \geq 0$. By connectedness of $\mathcal{T}$, $\phi_1 = \mathbb{1}/\sqrt{K}$. We write $\phi_l(k)$ for the k-th coordinate of ϕ_l. Group the distinct eigenvalues as $0 = \nu_{(1)} < \nu_{(2)} < \cdots < \nu_{(p)}$ with multiplicities $1 = m_1, m_2, \ldots, m_p$ (so $\sum_j m_j = K$), and let $E_j := \ker(L_\mathcal{T} - \nu_{(j)} I_K)$ be the corresponding eigenspace; in particular $E_1 = \operatorname{span}\{\mathbb{1}\}$. Write $\gamma_{(j)}$ and $s_{(j)}$ for the common values of γ_l and s_l on E_j.

I.2 The Spectral Form of the Minimizers

We prove the following proposition, which is more general than Theorem 4. In §I.9, we establish Theorem 4 as a simple corollary of the following.

Proposition 8 (Spectral decomposition of minimizers)**.** *Under the conventions of §I.1, let Y be admissible and the feature dimension $d \geq K-1$. Then we can eigendecompose Y in $L_\mathcal{T}$'s orthonormal eigenbasis U; write*

$$L_\mathcal{T} \;=\; U\,\Lambda\,U^\top, \qquad \Lambda = \operatorname{diag}(0, \nu_2, \cdots, \nu_K);$$
$$Y \;=\; U\,\Gamma\,U^\top, \qquad \Gamma \;=\; \operatorname{diag}(\gamma_1, \gamma_2, \ldots, \gamma_K);$$

for the simultaneous diagonalization (so $\gamma_l = \gamma_{l'}$ whenever $\nu_l = \nu_{l'}$). Let (M^*, W^*) be any global minimizer of the class-mean objective, $\mathcal{J}_Y(M, W)$; then M^* and W^* admit the factorizations

$$M^* \;=\; U \cdot \Sigma \cdot R, \qquad W^* \;=\; U \cdot \Theta \cdot R,$$

sharing the same factors U and R, where

- *U is the orthonormal eigenbasis of $L_\mathcal{T}$ from above, with l-th column the eigenvector for ν_l;*
- *R is an arbitrary real orthogonal $d \times d$ matrix;*
- *$\Sigma \in \mathbb{R}^{K \times d}$ is the matrix carrying the singular values $\sigma_2, \ldots, \sigma_K$ on its sub-diagonal, i.e.,*

$$\Sigma \;=\; \begin{bmatrix} 0_{1\times d} \\ \operatorname{diag}(\sigma_2, \ldots, \sigma_K) \;\big|\; 0_{(K-1)\times(d-K+1)} \end{bmatrix};$$

  the singular values are $\sigma_1 = 0$ and, for $l \geq 2$,

$$\sigma_l^2 \;=\; \left[\, |\gamma_l| \cdot \sqrt{\frac{\lambda_W}{\lambda_M\, s_l}} \;-\; \lambda_W \,\right]_+ ;$$

- $\Theta \in \mathbb{R}^{K\times d}$ *shares the sub-diagonal shape of* Σ*, with the values* θ_l *replacing* σ_l*:*

$$\Theta \;=\; \begin{bmatrix} 0_{1\times d} \\ \operatorname{diag}(\theta_2,\ldots,\theta_K) \;\big|\; 0_{(K-1)\times(d-K+1)} \end{bmatrix},$$

where

$$\theta_l \;=\; \frac{\gamma_l\,\sigma_l}{\sigma_l^2+\lambda_W}, \quad 2\le l\le K.$$

In particular, M^* *and* W^* *have the same active modes:* $\theta_l = 0$ *if and only if* $\sigma_l = 0$*. Both factorizations are unique only up to rotation within each repeated-eigenvalue eigenspace of* $L_{\mathcal{T}}$ *(acting on* U*) and an arbitrary orthogonal change of feature-space basis (acting on* R*); the products* $M^*M^{*\top}$ *and* $W^*M^{*\top}$ *are unaffected by these and are hence fully determined. Conversely,* $(U\Sigma R, U\Theta R)$ *is a global minimizer for every orthogonal* $R\in\mathbb{R}^{d\times d}$*.*

I.3 Existence of a Minimizer

Lemma 9 (Existence)**.** *For admissible* Y *and* $\lambda_W,\lambda_M>0$*, the class-mean objective* $\mathcal{J}_Y$ *of equation 4 attains its minimum over* $\{(M,W): M^\top\mathbb{1}=0\}$*.*

Proof. We note that

$$\begin{aligned}\mathcal{J}_Y(M,W) &= \|MW^\top - Y\|_F^2 + \lambda_W\|W\|_F^2 + \lambda_M\,\mathrm{tr}(M^\top SM)\\ &\ge \lambda_W\|W\|_F^2 + \lambda_M\;\mathrm{tr}(M^\top SM)\\ &\ge \lambda_W\|W\|_F^2 + \lambda_M\cdot s_{\min}\cdot\|M\|_F^2\end{aligned}$$

where $s_{\min}$ is the smallest non-zero eigenvalue of S. This means that the sub-level sets $\mathcal{S}_c := \{(M,W)\mid \mathcal{J}_Y(M,W)\le c\}$ are bounded. The domain of $\mathcal{J}_Y$ is $\{(M,W): M^\top\mathbb{1}=0\}$, which is closed; since $\mathcal{J}_Y$ is continuous, its sub-level sets are closed. Sub-level sets are therefore compact by Heine-Borel, and $\mathcal{J}_Y$ attains a global minimum on non-empty sub-level sets by the Weierstrass extreme value theorem. At least one such sub-level set is non-empty, and a minimizer on any such set must be a global minimizer across the entire domain. □

I.4 Eliminating the Classifier

Our goal in this section is to reduce the optimization of $\mathcal{J}_Y(M,W)$ over the variables M and W to an optimization over $G = MM^\top$ alone. We do this in two parts. First, we eliminate the optimization variable W, showing that it is a function of M at the optimum. We then state and prove a small technical lemma, which will be used here and elsewhere throughout the appendices. Finally, we use our closed-form expression for the optimal W^* to obtain a simpler objective over G.

Eliminating the classifier. The map $W\mapsto\mathcal{J}_Y(M,W)$ is a strictly convex quadratic, with

$$\nabla_W\mathcal{J}_Y \;=\; 2(WM^\top - Y^\top)M \;+\; 2\lambda_W W.$$

Setting the gradient to zero yields the normal equation $W(M^\top M+\lambda_W I_d) = Y^\top M$. The matrix $M^\top M+\lambda_W I_d$ is positive-definite, hence invertible, so we have

$$W^* \;=\; Y^\top M\,(M^\top M+\lambda_W I_d)^{-1}. \tag{12}$$

A technical lemma. Next, we state and prove a small lemma.

Lemma 10 (Push-through Lemma)**.** *For any* $B\in\mathbb{R}^{a\times b}$ *and* $\lambda_W>0$*, writing* $Z := BB^\top$ *for the Gram matrix of* B*,*

$$B(B^\top B+\lambda_W I_b)^{-1}B^\top \;=\; Z(Z+\lambda_W I_a)^{-1}.$$

Proof. From $(B^\top B+\lambda_W I_b)\,B^\top \;=\; B^\top\,(BB^\top+\lambda_W I_a)$, left-multiply by $(B^\top B+\lambda_W I_b)^{-1}$ and right-multiply by $(BB^\top+\lambda_W I_a)^{-1}$ to obtain $B^\top(Z+\lambda_W I_a)^{-1} = (B^\top B+\lambda_W I_b)^{-1}B^\top$. Left-multiply by B. □

Reducing to the Gram objective. With W eliminated, we substitute the optimal W^* back into $\mathcal{J}_Y$ to obtain an objective in M alone. In fact we will show something stronger: the resulting objective depends on M *only* through its Gram matrix $G := MM^\top$. This reduction enables the rest of our analysis by exposing the influence of $L_{\mathcal{T}}$'s spectrum. Lemma 10 (applied with $B = M$, $Z = G$) is precisely the tool that rewrites every appearance of M in terms of G. We examine the three terms of $\mathcal{J}_Y$ in turn.

Data-fit term. Substituting equation 12 and applying Lemma 10,

$$MW^{*\top} \;=\; M(M^\top M + \lambda_W I_d)^{-1} M^\top Y \;=\; G(G + \lambda_W I_K)^{-1} Y.$$

Then, using $I - G(G + \lambda_W I_K)^{-1} = \lambda_W (G + \lambda_W I_K)^{-1}$ (from $(G + \lambda_W I_K) - G = \lambda_W I_K$),

$$MW^{*\top} - Y \;=\; -\lambda_W (G + \lambda_W I_K)^{-1} Y, \tag{13}$$

a closed form in G. Taking the squared Frobenius norm (via $\|A\|_F^2 = \operatorname{tr}(A^\top A)$ and symmetry of $(G + \lambda_W I_K)^{-1}$),

$$\left\| MW^{*\top} - Y \right\|_F^2 \;=\; \lambda_W^2 \operatorname{tr}\left(YY^\top (G + \lambda_W I_K)^{-2} \right).$$

Weight-decay term. Next, we examine the weight-decay term $\|W^*\|_F^2$. Transposing equation 12 and applying Lemma 10 once again,

$$W^* W^{*\top} = Y^\top M (M^\top M + \lambda_W I_d)^{-2} M^\top Y = Y^\top G (G + \lambda_W I_K)^{-2} Y,$$

so

$$\lambda_W \|W^*\|_F^2 \;=\; \lambda_W \operatorname{tr}\left(YY^\top G \, (G + \lambda_W I_K)^{-2} \right).$$

Regularizer term. Cyclicity of trace immediately yields

$$\lambda_M \operatorname{tr}(M^\top S M) = \lambda_M \operatorname{tr}(SG).$$

The reduced objective. All three terms now depend on M only through G:

$$\begin{aligned}
\left\| MW^{*\top} - Y \right\|_F^2 + \lambda_W \|W^*\|_F^2 &= \lambda_W^2 \operatorname{tr}\left(YY^\top (G + \lambda_W I_K)^{-2} \right) + \lambda_W \operatorname{tr}\left(YY^\top G (G + \lambda_W I_K)^{-2} \right) \\
&= \lambda_W \operatorname{tr}\left[\lambda_W YY^\top (G + \lambda_W I_K)^{-2} + YY^\top G (G + \lambda_W I_K)^{-2} \right] \\
&= \lambda_W \operatorname{tr}\left[YY^\top (G + \lambda_W I_K)(G + \lambda_W I_K)^{-2} \right] \\
&= \lambda_W \operatorname{tr}\left[YY^\top (G + \lambda_W I_K)^{-1} \right]
\end{aligned}$$

Using the identity $\lambda_W (G + \lambda_W I_K)^{-1} = I - G(G + \lambda_W I_K)^{-1}$, this simplifies further to

$$\left\| MW^{*\top} - Y \right\|_F^2 + \lambda_W \|W^*\|_F^2 = \operatorname{tr}(YY^\top) - \operatorname{tr}\left(YY^\top G (G + \lambda_W I_K)^{-1} \right).$$

Discarding the constant term $\operatorname{tr}(YY^\top)$ and adding the regularizer term, we get the objective

$$\min_{G \in \mathcal{G}} \mathcal{F}(G), \qquad \mathcal{F}(G) \;:=\; -\operatorname{tr}\left(YY^\top G \, (G + \lambda_W I_K)^{-1} \right) \;+\; \lambda_M \operatorname{tr}(SG), \tag{14}$$

over the convex cone $\mathcal{G} := \{G \succeq 0 : G\mathbb{1} = 0,\ G = G^\top\}$. The feasible set is exactly the set of admissible Grams. G must be positive semi-definite and symmetric because $G = MM^\top$, and $G\mathbb{1} = 0$ encodes the centering constraint $M^\top \mathbb{1} = 0$, since $\mathbb{1}^\top G \mathbb{1} = \|M^\top \mathbb{1}\|^2$.

We have thus collapsed the two-variable problem in (M, W) to a single convex-cone problem in G. For every feasible (M, W), $\mathcal{J}_Y(M, W) \geq \mathcal{J}_Y(M, W^*(M)) = \mathcal{F}(MM^\top) + \operatorname{tr}(YY^\top)$. We have equality in the first step if and only if $W = W^*(M)$, so (M, W) minimizes $\mathcal{J}_Y$ if and only if $MM^\top$ minimizes $\mathcal{F}$ over $\mathcal{G}$ and $W = W^*(M)$. We defer the solution of equation 14 to the next two sections, and return in §I.8 to recover class-mean optima (M^*, W^*) from the Gram-objective optimum.

I.5 Symmetry Reduction

The reduced problem equation 14 is convex and invariant under a natural group of orthogonal symmetries. We exploit both properties to reduce G^* to a form that is scalar on each $L_{\mathcal{T}}$-eigenspace.

Lemma 11 (Convexity). *$\mathcal{F}$ is convex on $\mathcal{G}$.*

Proof. The regularizer $\operatorname{tr}(SG)$ is linear in G. For the data-fit term, the identity $G(G+\lambda_W I_K)^{-1} = I_K - \lambda_W(G+\lambda_W I_K)^{-1}$ gives

$$-\operatorname{tr}\big(YY^\top G(G+\lambda_W I_K)^{-1}\big) \;=\; -\operatorname{tr}(YY^\top) \;+\; \lambda_W \operatorname{tr}\big(YY^\top (G+\lambda_W I_K)^{-1}\big).$$

Since $\lambda_W > 0$ and the regularizer is linear, it suffices to show that $G \mapsto \operatorname{tr}\big(YY^\top (G+\lambda_W I_K)^{-1}\big)$ is convex on $\mathcal{G}$. Writing $y_1, \dots, y_K$ for the columns of Y,

$$\operatorname{tr}\big(YY^\top (G+\lambda_W I_K)^{-1}\big) \;=\; \sum_{k=1}^{K} y_k^\top (G+\lambda_W I_K)^{-1} y_k,$$

so it is enough that each $\varphi_k(G) := y_k^\top (G+\lambda_W I_K)^{-1} y_k$ is convex. For any $A \succ 0$, completing the square in the quadratic $z \mapsto z^\top A z - 2y^\top z$ (minimized at $z = A^{-1}y$, where it equals $-y^\top A^{-1} y$) yields the identity

$$y^\top A^{-1} y \;=\; \max_{z \in \mathbb{R}^K} \big(2\, y^\top z - z^\top A z\big).$$

Taking $A = G + \lambda_W I_K \succ 0$ (valid for all $G \succeq 0$),

$$\varphi_k(G) \;=\; \max_{z \in \mathbb{R}^K} \big(2\, y_k^\top z - \lambda_W \|z\|^2 - z^\top G z\big).$$

For each fixed z the bracketed expression is affine in G: the first two terms are constant in G, and the third, $z^\top G z = \operatorname{tr}(zz^\top G)$, is linear in G. Thus φ_k is a pointwise supremum of affine functions of G, hence convex. Summing over k and adding the linear regularizer, $\mathcal{F}$ is convex on $\mathcal{G}$. □

Define the group

$$\mathcal{S}_{\mathcal{T}} \;:=\; O(m_2) \times \cdots \times O(m_p),$$

where $O(m_j) := \{Q \in \mathbb{R}^{m_j \times m_j} : Q^\top Q = I_{m_j}\}$ denotes the orthogonal group in dimension m_j. For $Q = (Q_2, \dots, Q_p) \in \mathcal{S}_{\mathcal{T}}$, let

$$P_Q \;:=\; U\,(I_1 \oplus Q_2 \oplus \cdots \oplus Q_p)\,U^\top \;\in\; O(K),$$

where $A_1 \oplus \cdots \oplus A_p$ denotes the block-diagonal matrix with blocks $A_1, \dots, A_p$ along the diagonal and zeros elsewhere. $\mathcal{S}_{\mathcal{T}}$ acts on $\mathbb{R}^{K \times K}$ via $G \mapsto G^Q$, where

$$G^Q \;:=\; P_Q\, G\, P_Q^\top.$$

The block I_1 acts trivially on $E_1 = \operatorname{span}(\phi_1) = \operatorname{span}\{\mathbb{1}\}$, and each $Q_j \in O(m_j)$ acts on E_j for $j \geq 2$. Equivalently, P_Q acts as Q_j on each non-trivial eigenspace E_j with $j \geq 2$ and as the identity on E_1.

Lemma 12 (Invariance). *For every $Q \in \mathcal{S}_{\mathcal{T}}$ and every $G \in \mathcal{G}$:* (i) $G^Q \in \mathcal{G}$; *and* (ii) $\mathcal{F}(G^Q) = \mathcal{F}(G)$.

Proof. P_Q commutes with both $S = U\Lambda_S U^\top$ and $YY^\top = U\Gamma^2 U^\top$. Recall that $P_Q = U(I_1 \oplus Q_2 \oplus \cdots \oplus Q_p)U^\top$. Then

$$P_Q S = U \cdot (I_1 \oplus Q_2 \oplus \cdots \oplus Q_p) \cdot \Lambda_S \cdot U^\top.$$

Both Λ_S and $I_1 \oplus Q_2 \oplus \cdots \oplus Q_p$ are block-diagonal on the partition of $\mathbb{R}^K$ into $L_{\mathcal{T}}$-eigenspaces; since Λ_S acts as a constant times the identity on every such eigenspace, the two matrices commute. Thus, using $U^\top U = I$, we have

$$P_Q S = U \cdot \Lambda_S \cdot (I_1 \oplus Q_2 \oplus \cdots \oplus Q_p) \cdot U^\top = U\Lambda_S U^\top U (I_1 \oplus Q_2 \oplus \cdots \oplus Q_p) U^\top = S P_Q.$$

A similar approach shows that P_Q commutes with $YY^\top$.

(i) Positive-semi-definiteness is preserved by orthogonal conjugation. For the centering constraint, observe that $\mathbb{1} = \sqrt{K}\phi_1$ lies in E_1, on which P_Q acts as the identity; in particular $P_Q^\top \mathbb{1} = \mathbb{1}$. Hence

$$G^Q\mathbb{1} \;=\; P_Q G P_Q^\top \mathbb{1} \;=\; P_Q G \mathbb{1} \;=\; 0,$$

using $G\mathbb{1} = 0$ in the last step.

(ii) For the regularizer, $\mathrm{tr}(S\,P_Q G P_Q^\top) = \mathrm{tr}(P_Q^\top S P_Q\, G) = \mathrm{tr}(SG)$ using commutation and cyclicity. For the data-fit term, use orthogonality of P_Q to get $(P_Q G P_Q^\top + \lambda_W I_K)^{-1} = P_Q(G + \lambda_W I_K)^{-1} P_Q^\top$ and hence

$$\begin{aligned}\mathrm{tr}\left(YY^\top P_Q G P_Q^\top (P_Q G P_Q^\top + \lambda_W I_K)^{-1}\right) \;&=\; \mathrm{tr}\left(P_Q^\top YY^\top P_Q\, G(G + \lambda_W I_K)^{-1}\right)\\ &=\; \mathrm{tr}\left(YY^\top G(G + \lambda_W I_K)^{-1}\right). \qquad \square\end{aligned}$$

Now, let $G^* \in \mathcal{G}$ be any minimizer of $\mathcal{F}$ (one exists by Lemma 9). The group $\mathcal{S}_\mathcal{T}$ is a finite product of orthogonal, hence compact, groups; it is therefore compact itself, and has a Haar probability measure μ. Define the symmetrization

$$G^\sharp \;:=\; \int_{\mathcal{S}_\mathcal{T}} (G^*)^Q \, d\mu(Q).$$

We claim that $G^\sharp \in \mathcal{G}$. By Lemma 12(i), the integrand $(G^*)^Q$ lies in $\mathcal{G}$ for every $Q \in \mathcal{S}_\mathcal{T}$. We have $G^\sharp\mathbb{1} = \int (G^*)^Q\mathbb{1}\, d\mu(Q) = 0$ and, for any $v \in \mathbb{R}^K$, $v^\top G^\sharp v = \int v^\top (G^*)^Q v\, d\mu(Q) \geq 0$. Moreover, $G^\sharp$ is itself a minimizer: by Lemma 12(ii) every $(G^*)^Q$ attains the minimum value, and Lemma 11 together with Jensen's inequality gives $\mathcal{F}(G^\sharp) \leq \int \mathcal{F}((G^*)^Q)\, d\mu(Q) = \min_\mathcal{G} \mathcal{F}$.

The Haar measure is translation-invariant, so $G^\sharp$ is $\mathcal{S}_\mathcal{T}$-invariant: $(G^\sharp)^Q = G^\sharp$ for every $Q \in \mathcal{S}_\mathcal{T}$. Equivalently, $P_Q G^\sharp = G^\sharp P_Q$ for all $Q \in \mathcal{S}_\mathcal{T}$. To complete this section, we state an elementary lemma:

Lemma 13 (Orbit-symmetric operators are scalar)**.** *Let $m \geq 1$ and let $A \in \mathbb{R}^{m\times m}$ be symmetric. If*

$$QAQ^\top \;=\; A \qquad \textit{for every } Q \in O(m),$$

then $A = s\, I_m$ for some $s \in \mathbb{R}$.

Proof. Since A is symmetric, it is orthogonally diagonalizable by the spectral theorem. Write $A = VDV^\top$ with $V \in O(m)$ and $D = \mathrm{diag}(d_1, \ldots, d_m)$. Suppose for the sake of contradiction that not all d_i are equal—say, $d_1 \neq d_2$ after relabeling.

Let $R \in O(m)$ be the orthogonal matrix that swaps the first two standard basis vectors and fixes the rest. Then a direct computation gives $RDR^\top = \mathrm{diag}(d_2, d_1, d_3, \ldots, d_m) \neq D$. Setting $Q := VRV^\top \in O(m)$, we have $QAQ^\top = V(RDR^\top)V^\top \neq VDV^\top = A$, contradicting the hypothesis. Thus, $d_1 = d_2 = \cdots = d_m$, and $A = s\, I_m$ for some real scalar s. $\square$

We apply the lemma block-by-block. Two preliminary observations about $G^\sharp$ are needed.

First, note that the restriction of $G^\sharp$ to each eigenspace E_j yields a well-defined operator over that eigenspace. Since $G^\sharp\mathbb{1} = 0$, $G^\sharp E_1 = \{0\} \subseteq E_1$. For each $j \geq 2$, choose $Q \in \mathcal{S}_\mathcal{T}$ with $Q_j = -I_{m_j}$ and all other $Q_{j'} = I_{m_{j'}}$; then P_Q has eigenspaces E_j (eigenvalue -1) and $E_j^\perp$ (eigenvalue 1). Since $G^\sharp$ commutes with P_Q, it maps each P_Q-eigenspace back into itself; in particular, we have $G^\sharp E_j \subseteq E_j$. As $G^\sharp$ maps vectors in E_j back to E_j, its restriction to E_j is a well-defined operator.

Second, note that $G^\sharp|_{E_j}$ is $O(m_j)$-invariant (again, for $j \geq 2$). For any $Q_j \in O(m_j)$, let $Q \in \mathcal{S}_\mathcal{T}$ with Q_j in the j-th slot and identities elsewhere. Then P_Q acts as Q_j on E_j and trivially on $E_j^\perp$. Restricting the commutation $P_Q G^\sharp P_Q^\top = G^\sharp$ to E_j yields $Q_j(G^\sharp|_{E_j})Q_j^\top = G^\sharp|_{E_j}$.

With these considerations in hand, Lemma 13 forces $G^\sharp|_{E_j} = t_j I_{m_j}$ for some $t_j \in \mathbb{R}$. Positive semi-definiteness of $G^\sharp$ forces $t_j \geq 0$. In tandem with $G^\sharp|_{E_1} = 0$, we get

$$G^\sharp \;=\; U\ \mathrm{diag}\big(0,\, t_2 I_{m_2},\, \ldots,\, t_p I_{m_p}\big)\, U^\top, \qquad t_j \geq 0. \tag{15}$$

Substituting into $\mathcal{F}$ reduces the optimization to a scalar problem across blocks.

I.6 PER-MODE OPTIMIZATION

By the symmetry reduction of §I.5, it suffices to minimize $\mathcal{F}$ over Gram matrices of the form equation 15. We substitute this form into Equation equation 14 and minimize the resulting expression over $t_j \geq 0$.

Decoupling. Restating equation 15, let $\widetilde{G} = \operatorname{diag}(0, t_2 I_{m_2}, \ldots, t_p I_{m_p})$ so $G^\sharp = U\widetilde{G}U^\top$. We make three observations:

- $S = U\Lambda_S U^\top$ with Λ_S's diagonal entries equal to $s_{(j)}$ on the j-th block. Then by cyclicity of trace,
$$\operatorname{tr}(SG^\sharp) = \operatorname{tr}(\Lambda_S \widetilde{G}) = \sum_{j=2}^{p} s_{(j)} m_j t_j$$
- $YY^\top = U\Gamma^2 U^\top$ with Γ^2's diagonal entries equal to $\gamma_{(j)}^2$ on the j-th block.
- For the data-fit term, we need to compute $G^\sharp(G^\sharp + \lambda_W I_K)^{-1}$. Two reductions are useful. First, the conjugation passes through cleanly. Writing $\lambda_W I_K = U(\lambda_W I_K)U^\top$ to absorb the shift into the conjugation,
$$G^\sharp + \lambda_W I_K \;=\; U\widetilde{G}U^\top + \lambda_W UU^\top \;=\; U(\widetilde{G} + \lambda_W I_K)U^\top,$$
so $(G^\sharp + \lambda_W I_K)^{-1} = U(\widetilde{G} + \lambda_W I_K)^{-1}U^\top$ by $U^{-1} = U^\top$. Multiplying by $G^\sharp = U\widetilde{G}U^\top$ and canceling the inner $U^\top U = I_K$,
$$G^\sharp(G^\sharp + \lambda_W I_K)^{-1} \;=\; U\,\widetilde{G}\,(\widetilde{G} + \lambda_W I_K)^{-1}\,U^\top.$$
Second, the inner factor is block-diagonal with scalar blocks. Since $\widetilde{G} = \operatorname{diag}(0, t_2 I_{m_2}, \ldots, t_p I_{m_p})$,
$$\widetilde{G} + \lambda_W I_K \;=\; \operatorname{diag}\big(\lambda_W I_{m_1},\, (t_2 + \lambda_W) I_{m_2},\, \ldots,\, (t_p + \lambda_W) I_{m_p}\big),$$
which is block-diagonal with each block a positive scalar multiple of the identity ($t_j + \lambda_W > 0$ since $\lambda_W > 0$ and $t_j \geq 0$). Inverting block-by-block and multiplying by $\widetilde{G}$ yields:
$$\widetilde{G}(\widetilde{G} + \lambda_W I_K)^{-1} \;=\; \operatorname{diag}\left(0,\, \frac{t_2}{t_2 + \lambda_W} I_{m_2},\, \ldots,\, \frac{t_p}{t_p + \lambda_W} I_{m_p}\right),$$
where the $j = 1$ block vanishes because the corresponding block of $\widetilde{G}$ is $0 \cdot I_{m_1}$ (the centering constraint forces $t_1 = 0$).

Combining and applying trace cyclicity to strip the outer $U, U^\top$:
$$-\operatorname{tr}\big(YY^\top G^\sharp(G^\sharp + \lambda_W I_K)^{-1}\big) \;=\; -\sum_{j=2}^{p} \gamma_{(j)}^2\, m_j\, \frac{t_j}{t_j + \lambda_W}.$$
(The $j = 1$ term vanishes because $t_1 = 0$.) Together, $\mathcal{F}$ evaluated on equation 15 decomposes as
$$\mathcal{F}(G^\sharp) \;=\; \sum_{j=2}^{p} m_j\, f_j(t_j), \qquad f_j(t) \;:=\; -\gamma_{(j)}^2\, \frac{t}{t + \lambda_W} \;+\; \lambda_M\, s_{(j)}\, t, \qquad t \geq 0. \tag{16}$$
The optimization decouples across j: minimizing $\mathcal{F}$ over Grams that are scalar on each eigenspace reduces to minimizing each f_j over $t_j \geq 0$ independently.

Per-block scalar problem. Each f_j is convex on $[0, \infty)$:
$$f_j''(t) \;=\; \frac{2\,\lambda_W\,\gamma_{(j)}^2}{(t + \lambda_W)^3} \;\geq\; 0,$$
with strict inequality (hence strict convexity) when $\gamma_{(j)} \neq 0$. When $\gamma_{(j)} = 0$, f_j reduces to the linear function $\lambda_M s_{(j)} t$, strictly increasing on $[0, \infty)$ since $\lambda_M, s_{(j)} > 0$, so the unique minimizer on $[0, \infty)$ is $t_j = 0$.

If $\gamma_{(j)} \neq 0$, setting $f_j'(t) = 0$ gives

$$(t + \lambda_W)^2 \;=\; \frac{\lambda_W\, \gamma_{(j)}^2}{\lambda_M\, s_{(j)}} \quad\Longleftrightarrow\quad t + \lambda_W \;=\; |\gamma_{(j)}| \sqrt{\frac{\lambda_W}{\lambda_M\, s_{(j)}}}, \tag{17}$$

which uses the positive root since $t + \lambda_W > 0$. The unconstrained minimizer is $t_j^\circ := |\gamma_{(j)}| \sqrt{\lambda_W / (\lambda_M\, s_{(j)})} - \lambda_W$; projecting onto $[0, \infty)$ by convexity of f_j yields $t_j = [t_j^\circ]_+$. This formula also captures the $\gamma_{(j)} = 0$ case (which gives $t_j = [-\lambda_W]_+ = 0$), so uniformly:

$$\boxed{t_j \;=\; \sigma_{(j)}^2 \;:=\; \left[\, |\gamma_{(j)}| \sqrt{\frac{\lambda_W}{\lambda_M\, s_{(j)}}} \;-\; \lambda_W \,\right]_+ \qquad (j \geq 2).} \tag{18}$$

Uniqueness of the per-block minimizer holds in both cases: by strict convexity when $\gamma_{(j)} \neq 0$, and by strict monotonicity when $\gamma_{(j)} = 0$.

Threshold characterization. The active set $J_{\text{active}} := \{j \geq 2 : \sigma_{(j)}^2 > 0\}$ is characterized by

$$j \in J_{\text{active}} \iff \lambda_W\, \lambda_M\, s_{(j)} \;<\; \gamma_{(j)}^2 \iff \gamma_{(j)} > \kappa_{(j)} \quad \text{(for admissible } Y\text{, where } \gamma_{(j)} \geq 0),$$

where (recall) $\kappa_{(j)} = \sqrt{\lambda_M \lambda_W s_{(j)}}$.

Conclusion. Substituting $t_j = \sigma_{(j)}^2$ into equation 15:

$$G^\sharp \;=\; U\, \operatorname{diag}\big(0,\, \sigma_{(2)}^2 I_{m_2},\, \ldots,\, \sigma_{(p)}^2 I_{m_p}\big)\, U^\top. \tag{19}$$

This is the unique minimizer of $\mathcal{F}$ that is scalar on each eigenspace, by strict convexity of each f_j on $[0, \infty)$. In particular, every minimizer of $\mathcal{F}$ on $\mathcal{G}$ symmetrizes to the same matrix equation 19.

I.7 Uniqueness of the Optimal Gram

Sections I.5–I.6 establish that the explicit matrix of equation 19 is a minimizer of $\mathcal{F}$ on $\mathcal{G}$, and that every minimizer $G^* \in \mathcal{G}$ symmetrizes to the same matrix. Earlier, we wrote $G^\sharp$ to denote the symmetrization of an arbitrary minimizer $G^* \in \mathcal{G}$; since we have established that every minimizer symmetrizes to the *same* matrix, we abuse notation and let $G^\sharp$ denote this unique matrix.

The purpose of this section is to establish that $G^\sharp$ is the *only* minimizer of the reduced Gram objective. That is, a minimizer G^* does not merely symmetrize to $G^\sharp$, it *is* $G^\sharp$. As a consequence, $G^\sharp$ is the *unique* minimizer of $\mathcal{F}$ over $\mathcal{G}$.

We work throughout in the fixed eigenbasis U, so both our closed form for the optimal Gram and the argument below are written in terms of U. The resulting matrix is nevertheless independent of this choice, a point we clarify at the end of this section once the necessary theory has been established. Thus, the *value* of $G^\sharp$ is independent of U even if its *expression* is not.

Let $G^* \in \mathcal{G}$ be a minimizer, and set $\Delta := G^* - G^\sharp$. We use the map Φ throughout, defined by

$$\Phi(G) \;:=\; \int_{\mathcal{S}_\mathcal{T}} G^Q \, d\mu(Q).$$

Φ sends a matrix $G \in \mathcal{G}$ to its symmetrization; if G minimizes the reduced Gram objective, we already know that $\Phi(G) = G^\sharp$.

To show that $G^* = G^\sharp$, we prove that $\Delta = 0$. The argument works along the segment $G_t := (1-t)G^\sharp + tG^*$ joining the two minimizers. Since the minimizer set is convex, $\mathcal{F}$ is convex on this segment, and reading off that constancy mode-by-mode constrains Δ. We split $\mathbb{R}^K$ into orthogonal eigenspaces and show that Δ vanishes on each. We conclude by establishing that $G^\sharp$, though written in the basis U, does not depend on that choice.

Defining $\mathcal{F}(G_t)$. Observe that both $\mathcal{G}$ and the minimizer set of $\mathcal{F}$ are convex, so $G_t := G^\sharp + t\Delta = (1-t)G^\sharp + tG^*$ is a minimizer of $\mathcal{F}(G)$ for all $t \in [0, 1]$. Thus $t \mapsto \mathcal{F}(G_t)$ is constant on $[0, 1]$. Writing $X_t := G_t + \lambda_W I_K \succ 0$,

$$\mathcal{F}(G_t) \;=\; \text{const} \;+\; \lambda_W\, \operatorname{tr}\big(YY^\top X_t^{-1}\big) \;+\; \lambda_M\, \operatorname{tr}(SG_t). \tag{20}$$

Δ vanishes on the active subspace. Recall that our regression targets Y admit a U-eigendecomposition,

$$Y = U \cdot \operatorname{diag}(\gamma_1, \gamma_2, \ldots, \gamma_K) \cdot U^\top.$$

Recall too (§I.1) that $\gamma_{(j)}$ denotes the common value of γ_l on the $L_{\mathcal{T}}$-eigenspace E_j. The *active eigenspace* is then

$$E_{\text{act}} = \bigoplus_{j:\gamma_{(j)} \neq 0} E_j.$$

Before proceeding, we state and prove a simple lemma.

Lemma 14. *Let ϕ_i be a U-eigenvector in E_{act}. Then*

$$\|X_t^{-1/2} \Delta X_t^{-1} \phi_i\|^2 = 0$$

for all $t \in [0, 1]$.

Proof. We recover $\|X_t^{-1/2} \Delta X_t^{-1} \phi_i\|^2$ from $\mathcal{F}(G_t)$. Examining equation 20, we observe that the left-hand-side is constant. On the right-hand-side, the first term is constant, and the third is affine in t. We prove the lemma by focusing on the second term on the right side, $\operatorname{tr}\left(YY^\top X_t^{-1}\right)$.

Write $YY^\top = \sum_{j=1}^K \gamma_j^2 \phi_j \phi_j^\top$ (where, recall, ϕ_j is the j-th column of U, an orthonormal eigenbasis of $L_{\mathcal{T}}$). Now, observe that

$$\operatorname{tr}\left(YY^\top X_t^{-1}\right) \;=\; \sum_{j=1}^K \gamma_j^2 \cdot \phi_j^\top X_t^{-1} \phi_j = \sum_{j=1}^K \gamma_j^2 h_j(t), \qquad h_j(t) := \phi_j^\top X_t^{-1} \phi_j.$$

Differentiation, using $\frac{d}{dt} X_t^{-1} = -X_t^{-1} \Delta X_t^{-1}$ and the symmetry of the relevant matrices, gives us

$$h_j''(t) \;=\; 2\, \phi_j^\top X_t^{-1} \Delta X_t^{-1} \Delta X_t^{-1} \phi_j \;=\; 2 \left\| X_t^{-1/2} \Delta X_t^{-1} \phi_j \right\|^2 \;\geq\; 0.$$

Note that the simplified expression for the second derivative matches the expression in the statement of the lemma. The second derivative is non-negative, so $h_j(t)$ is convex.

The data-fit term $\sum_{j=1}^K \gamma_j^2 h_j(t)$ is a linear combination of convex functions with non-negative coefficients. However, re-arranging Equation equation 20 reveals that the data-fit term (there written as $\operatorname{tr}\left(YY^\top X_t^{-1}\right)$) is affine (since every other term in that equality is affine). Since this combination is affine in t, its second derivative vanishes:

$$\sum_{j=1}^K \gamma_j^2 \, h_j''(t) = 0.$$

Every summand is non-negative, so each term with $\gamma_j \neq 0$ must satisfy $h_j''(t) = 0$.

Since $\phi_i \in E_{\text{act}}$, the above considerations imply $h_i''(t) = 0$, which completes the proof. □

We use the above lemma to prove that Δ vanishes on E_{act}. At $t = 0$, $X_0 = G^\sharp + \lambda_W I_K$ acts as the scalar $\sigma_{(j)}^2 + \lambda_W$ times the identity on each eigenspace E_j, by equation 19. Hence every U-eigenvector ϕ_i in E_j is an eigenvector of X_0^{-1}, with eigenvalue $(\sigma_{(j)}^2 + \lambda_W)^{-1} > 0$. Lemma 14 tells us that

$$0 \;=\; \left\| X_0^{-1/2} \Delta X_0^{-1} \phi_i \right\|^2 \;=\; (\sigma_{(j)}^2 + \lambda_W)^{-2} \left\| X_0^{-1/2} \Delta \phi_i \right\|^2.$$

But since $X_0^{-1/2}$ is invertible, we must have $\Delta \phi_i = 0$. Thus, Δ vanishes on E_{act}.

Δ vanishes on inactive subspaces. Consider $E_1 = \operatorname{span}\{\mathbb{1}\}$, the first eigenspace of $L_{\mathcal{T}}$. Since $G^*, G^\sharp \in \mathcal{G}$, we have $\Delta\mathbb{1} = 0$. So Δ vanishes on E_1.

Thus, if we define $E_{\text{inact}} = \bigoplus_{j:j\geq 2, \gamma_{(j)}=0} E_j$, then the range of Δ is contained in E_{inact}.

The closed-form expression for $G^\sharp$ (equation 19) tells us that $G^\sharp$ vanishes on E_{inact}. To show that Δ vanishes on this space, then, it will suffice to show that G^* vanishes there.

Before showing that G^* vanishes on E_{inact}, we state and prove a simple lemma. For E_j any $L_{\mathcal{T}}$-eigenspace, let P_{E_j} denote the orthogonal projection onto E_j. Further, for an operator A, let $A|_{E_j} = P_{E_j} A P_{E_j}$ denote the compression of A onto E_j. Recall that Φ maps a matrix onto its symmetrization.

Lemma 15. *For any $L_{\mathcal{T}}$-eigenspace E_j, we have*

$$\operatorname{tr}(\Phi(\Delta)|_{E_j}) = \operatorname{tr}(\Delta|_{E_j}).$$

Proof. The trace of a compression simplifies: for any M, cyclicity of trace and $P_{E_j}^2 = P_{E_j}$ give

$$\operatorname{tr}(M|_{E_j}) = \operatorname{tr}(P_{E_j} M P_{E_j}) = \operatorname{tr}(M P_{E_j}).$$

For $Q \in \mathcal{S}_{\mathcal{T}}$, the matrix P_Q maps E_j onto itself, and therefore P_Q commutes with P_{E_j}. Since P_Q is orthogonal, it fixes P_{E_j} under conjugation:

$$P_Q^\top P_{E_j} P_Q \;=\; P_Q^\top P_Q\, P_{E_j} \;=\; P_{E_j}.$$

Combining these facts yields

$$\operatorname{tr}\big((\Delta^Q)|_{E_j}\big) \;=\; \operatorname{tr}(P_Q \Delta P_Q^\top P_{E_j}) \;=\; \operatorname{tr}\big(\Delta\, P_Q^\top P_{E_j} P_Q\big) \;=\; \operatorname{tr}(\Delta P_{E_j}) \;=\; \operatorname{tr}(\Delta|_{E_j})$$

Compression and trace are both linear in Δ and therefore commute with integration. The above equality gives

$$\operatorname{tr}\big(\Phi(\Delta)|_{E_j}\big) \;=\; \int_{\mathcal{S}_{\mathcal{T}}} \operatorname{tr}\big((\Delta^Q)|_{E_j}\big)\, d\mu(Q) \;=\; \int_{\mathcal{S}_{\mathcal{T}}} \operatorname{tr}(\Delta|_{E_j})\, d\mu(Q) \;=\; \operatorname{tr}(\Delta|_{E_j}). \tag{21}$$

□

Now, we can continue arguing that G^* vanishes on E_{inact}. We have

$$\operatorname{tr}(G^*|_{E_j}) \;=\; \operatorname{tr}(G^\sharp|_{E_j}) + \operatorname{tr}(\Delta|_{E_j}) \;=\; 0 + 0 \;=\; 0, \qquad (j \in E_{\text{inact}}).$$

The closed-form expression for $G^\sharp$ equation 19 tells us that the first term is zero. For the second term, note that since Φ is linear and sends every minimizer to $G^\sharp$,

$$\Phi(\Delta) \;=\; \Phi(G^*) - \Phi(G^\sharp) \;=\; G^\sharp - G^\sharp \;=\; 0.$$

Lemma 15 then gives $\operatorname{tr}(\Delta|_{E_j}) = \operatorname{tr}(\Phi(\Delta)|_{E_j}) = 0$.

The above suffices to show that the diagonal blocks of $G^*|_{E_{\text{inact}}}$ are zero. It remains to show that its off-diagonal blocks, $P_{E_{j'}} G^* P_{E_j}$ for distinct inactive modes j and j', are also zero. This is forced by the zero diagonal blocks and positive semi-definiteness of $G^*|_{E_{\text{inact}}}$. Writing $A = G^*|_{E_{\text{inact}}}$, take any $\phi_j \in E_j$, $\phi_{j'} \in E_{j'}$ and $s \in \mathbb{R}$. Then $v := s\,\phi_j + \phi_{j'} \in E_{\text{inact}}$ gives

$$0 \;\leq\; v^\top A v \;=\; s^2\, \phi_j^\top A \phi_j \;+\; 2s\, \phi_j^\top A \phi_{j'} + \phi_{j'}^\top A \phi_{j'} = 2s \phi_j^\top A \phi_{j'}$$

where the diagonal-block terms $\phi_j^\top A \phi_j = \phi_{j'}^\top A \phi_{j'} = 0$. A linear function of s is non-negative for all $s \in \mathbb{R}$ only if its slope vanishes, so $\phi_j^\top A \phi_{j'} = 0$. Since $\phi_j, \phi_{j'}$ are arbitrary elements of E_j and $E_{j'}$, the entire off-diagonal block $P_{E_j} G^* P_{E_{j'}}$ is 0. Thus $G^*|_{E_{\text{inact}}} = 0$, and hence $\Delta|_{E_{\text{inact}}} = 0$.

Conclusion. Δ vanishes on E_1, E_{act}, and E_{inact}. Since $\mathbb{R}^K$ is an orthogonal decomposition of these three spaces, and Δ is symmetric, it vanishes everywhere on $\mathbb{R}^K$. Thus, $G^\sharp$ is the unique minimizer of $\mathcal{F}$ on $\mathcal{G}$, as claimed. The uniqueness of $G^\sharp$ is used to characterize the uniqueness of minimizers (M^*, W^*) of equation 4.

Independence from the eigenbasis. The argument above fixes an eigenbasis U and shows that the minimizer set equals the singleton $\{G^\sharp\}$, with $G^\sharp$ given by equation 19. The minimizer set, however, is intrinsic to the problem equation 14, and is determined independently of any choice of eigenbasis. Running the same argument with any other eigenbasis U' of $L_\mathcal{T}$ identifies this *same* set with the singleton $\{(G^\sharp)'\}$, where $(G^\sharp)'$ is equation 19 written in U'. Two singletons that coincide as sets share their element, so $G^\sharp = (G^\sharp)'$: while the *expression* for the optimal Gram is written in terms of U, its *value* does not depend on that choice.

I.8 Recovering M^* and W^*

The preceding two sections show that the Gram objective (equation 14) has a unique minimizer,

$$G^\sharp \;=\; U\,\mathrm{diag}\left(0,\,\sigma_{(2)}^2 I_{m_2},\,\ldots,\,\sigma_{(p)}^2 I_{m_p}\right) U^\top, \tag{22}$$

with $\sigma_{(j)}^2$ given by equation 18. By the equivalence recorded at the end of §I.4, (M, W) minimizes $\mathcal{J}_Y$ if and only if $MM^\top = G^\sharp$ and $W = W^*(M)$. We must therefore recover M^* and W^* from $G^\sharp$ and characterize the residual non-uniqueness.

Recovering M^*. Write $r = \mathrm{rank}(G^\sharp)$. A factor $M^* \in \mathbb{R}^{K\times d}$ with $M^*M^{*\top} = G^\sharp$ exists if and only if $d \geq r$. We have $r \leq K-1$, so under our standing assumption of $d \geq K-1$ (§I.1), this condition always holds.

We now construct a matrix M with $MM^\top = G^\sharp$. Define the matrix

$$\Sigma \;:=\; \left[\begin{array}{c} 0_{1\times d} \\ \hline \mathrm{diag}(\sigma_2,\ldots,\sigma_K) \;\big|\; 0_{(K-1)\times(d-K+1)} \end{array}\right] \in \mathbb{R}^{K\times d},$$

where for each $i \geq 2$, σ_i equals $\sigma_{(j)}$ for the unique mode j with $i \in E_j$. Note that $\Sigma\Sigma^\top$ is the $K \times K$ matrix $\mathrm{diag}\,(0, \sigma_2^2, \ldots, \sigma_K^2)$.

Direct computation shows that if we take $M_0 = U\Sigma$, then $M_0 M_0^\top = G^\sharp$. With one Gram factorization in hand, we can describe the *complete* set of such factorizations (Horn & Johnson, 2012, Theorem 7.3.11). Every matrix M^* with $M^*M^{*\top} = G^\sharp$ must factor as

$$M^* = U \cdot \Sigma \cdot R, \qquad R \in O(d), \tag{23}$$

where $O(d)$ is the group of real orthogonal $d \times d$ matrices. Conversely, $(U\Sigma R)(U\Sigma R)^\top = G^\sharp$ for every $R \in O(d)$, so every such M^* is the class-mean factor of a minimizer.

Recovering W^*. From equation 12, $W^* = Y^\top M^*(M^{*\top}M^* + \lambda_W I_d)^{-1}$. Substituting equation 23, and using $Y = Y^\top = U\Gamma U^\top$ (§I.1):

$$W^* \;=\; U\Gamma U^\top \cdot U\Sigma R \cdot (R^\top\Sigma^\top\Sigma R + \lambda_W I_d)^{-1} \;=\; U\,\Gamma\Sigma\,(\Sigma^\top\Sigma + \lambda_W I_d)^{-1}\,R,$$

where we used $U^\top U = I_K$ and $(R^\top A R + \lambda_W I_d)^{-1} = R^\top(A + \lambda_W I_d)^{-1}R$ for any A and orthogonal R. For $2 \leq i \leq K$, define

$$\theta_i = \frac{\gamma_i\,\sigma_i}{\sigma_i^2 + \lambda_W};$$

then the matrix $\Gamma\Sigma(\Sigma^\top\Sigma + \lambda_W I_d)^{-1}$ is

$$\left[\begin{array}{c} 0_{1\times d} \\ \hline \mathrm{diag}(\theta_2,\ldots,\theta_K) \;\big|\; 0_{(K-1)\times(d-K+1)} \end{array}\right].$$

For active modes ($i \geq 2$, $\sigma_i > 0$), the per-mode condition $\sigma_i^2 + \lambda_W = |\gamma_i|\sqrt{\lambda_W/(\lambda_M s_i)}$ from §I.6 gives

$$\theta_i \;=\; \frac{\gamma_i\,\sigma_i}{\sigma_i^2 + \lambda_W} \;=\; \mathrm{sign}(\gamma_i)\,\sigma_i\sqrt{\frac{\lambda_M\,s_i}{\lambda_W}}.$$

We now have the following expression for W^*:

$$W^* \;=\; U\left[\begin{array}{c} 0_{1\times d} \\ \hline \mathrm{diag}(\theta_2,\ldots,\theta_K) \;\big|\; 0_{(K-1)\times(d-K+1)} \end{array}\right] R. \tag{24}$$

M^* and W^* share the same left singular basis U and the same right rotation R; their singular values are linked by $|\theta_i| = \sigma_i\sqrt{\lambda_M s_i/\lambda_W}$, with the sign of θ_i inherited from γ_i.

Residual symmetry. By equation 23 and equation 24, the minimizer set of $\mathcal{J}_Y$ is exactly

$$\big\{\,(U\Sigma R,\ U\Theta R)\ :\ R\in O(d)\,\big\}.$$

This suggests that there are two sources of non-uniqueness: the choice of eigenbasis U on the left, and an orthogonal transformation R on the right. In fact, the set of minimizers does *not* depend on the choice of U. Take another orthonormal $L_{\mathcal{T}}$ eigenbasis $U' = U\,(I_1 \oplus Q_2 \oplus \cdots \oplus Q_p)$ with each $Q_j \in O(m_j)$ an orthogonal transformation. Since σ_l and θ_l are constant on each eigenspace, we have

$$(I_1\oplus Q_2\oplus\cdots\oplus Q_p)\,\Sigma \;=\; \Sigma\,Q', \qquad (I_1\oplus Q_2\oplus\cdots\oplus Q_p)\,\Theta \;=\; \Theta\,Q', \qquad Q' := (Q_2\oplus\cdots\oplus Q_p)\oplus I_{d-K+1} \in O(d).$$

Selecting a different eigenbasis therefore reparameterizes the minimizer set without changing it. The *sole* source of residual non-uniqueness is in the choice of R.

Conclusion. The optimal Gram $G^\sharp$ is uniquely determined by equation 22, and the pair (M^*, W^*) is uniquely determined modulo a common right-multiplication by some $R \in O(d)$. This completes the proof of Proposition 8.

I.9 The Decision Metric and Parts (i)–(iii)

We now derive Theorem 4 from Proposition 8. That $\mathcal{J}_Y$ attains its minimum is Lemma 9, and uniqueness up to a common orthogonal transformation was established at the end of §I.8. It remains to construct A and verify parts (i)–(iii).

The metric A. For any target Y satisfying the hypotheses of Proposition 8, define the matrix

$$A \;:=\; R^\top D\,R \;\in\; \mathbb{R}^{d\times d}, \qquad D \;:=\; \operatorname{diag}\left(\frac{\gamma_2}{\sigma_2^2+\lambda_W},\ldots,\frac{\gamma_K}{\sigma_K^2+\lambda_W},\underbrace{0,\ldots,0}_{d-K+1}\right) \;\in\; \mathbb{R}^{d\times d} \tag{25}$$

where $\Gamma = \operatorname{diag}(\gamma_l)$ is Y's spectrum in the U-basis and R is the same matrix as in the factorizations of M^* and W^*. We write $D_l := \gamma_l/(\sigma_l^2+\lambda_W)$ for the diagonal entries; in the notation of the main text, $A = \sum_{l\ge 2} D_l\, r_l r_l^\top$ when $d = K-1$. When $\gamma_l \ge 0$ for all l, A is symmetric positive semi-definite, and the classifier's argmax rule satisfies

$$\arg\max_k \langle w_k^*, h\rangle \;=\; \arg\min_k \Big(\|h-m_k^*\|_A^2 \;-\; m_k^{*\top} A\, m_k^*\Big), \tag{26}$$

i.e., argmax-logit is nearest-class-mean under the Mahalanobis (semi-)metric A, up to a per-class bias correction $-\,m_k^{*\top} A\, m_k^*$.

Positive-definiteness. When $d = K-1$ the zero block of D is empty, and $D_l > 0$ exactly when $\gamma_l > 0$; thus $A \succ 0$ whenever $\gamma_l > 0$ for every $l \ge 2$. Admissibility permits $\gamma_l = 0$, in which case $\sigma_l = \theta_l = 0$, so both M^* and W^* vanish along r_l. The value of A on such directions (and on the $d-K+1$ padding directions when $d > K-1$) is then immaterial to parts (i)–(iii) below, whose derivations use only $M^*A = W^*$ and $M^*AM^{*\top}$: replacing each such zero entry of D by any positive number (say λ_W^{-1}) leaves all three conclusions intact and makes A symmetric positive-definite. This is the A that Theorem 4 asserts exists.

Part (i): $W^* = M^*A$. The logit at feature h for class k is $w_k^{*\top} h = (e_k^\top W^*)h$. We claim

$$\langle w_k^*, h\rangle \;=\; (m_k^*)^\top A\, h. \tag{27}$$

To see this, write $m_k^{*\top} = e_k^\top M^* = e_k^\top U\Sigma R$ and $w_k^{*\top} = e_k^\top W^* = e_k^\top U\Theta R$. Then

$$(m_k^*)^\top A\, h \;=\; e_k^\top U\Sigma R\cdot R^\top D R\cdot h \;=\; e_k^\top U\,\Sigma\, D\, R\, h,$$

using $RR^\top = I_d$. The middle product ΣD retains the sub-diagonal form of Σ: since D's leading slots align with Σ's sub-diagonal columns, right-multiplication scales each non-zero entry σ_l to $\sigma_l\cdot\gamma_l/(\sigma_l^2+\lambda_W)$, which is exactly θ_l (Proposition 8). Hence the middle factor *equals* Θ, and $(m_k^*)^\top A\, h = e_k^\top U\Theta R\, h = w_k^{*\top} h$, establishing equation 27. Since equation 27 holds for every k and h, we have $W^* = M^*A$, which is part (i).

Part (ii): the decision rule. Expand the squared Mahalanobis (semi-)distance:

$$\|h - m_k^*\|_A^2 \;=\; h^\top A h \;-\; 2\, m_k^{*\top} A\, h \;+\; m_k^{*\top} A\, m_k^*.$$

The first term $h^\top A h$ is independent of k and drops out of the decision rule. Substituting the logit identity equation 27 and using the expansion above,

$$\arg\max_k \langle w_k^*, h\rangle \;=\; \arg\max_k\; m_k^{*\top} A\, h \;=\; \arg\min_k\big(-2\, m_k^{*\top} A\, h\big) \;=\; \arg\min_k\big(\|h - m_k^*\|_A^2 - m_k^{*\top} A\, m_k^*\big),$$

which is equation 26; the final equality adds the k-independent constant $h^\top A h$. Observing that $m_k^{*\top} A m_k^* = \|m_k^*\|_A^2$, we have established part (ii).

Part (iii): the A-Gram of the class means. We first record two identities relating D_l and σ_l to the target spectrum.

Lemma 16. *For a target with mode-l eigenvalue γ_l and induced singular value σ_l (Proposition 8), on any active mode ($l \geq 2$ with $\sigma_l > 0$),*

$$D_l \;=\; \operatorname{sign}(\gamma_l)\sqrt{\frac{\lambda_M\, s_l}{\lambda_W}}, \qquad \textit{and} \qquad D_l\, \sigma_l^2 \;=\; \gamma_l \;-\; \operatorname{sign}(\gamma_l)\sqrt{\lambda_M \lambda_W\, s_l}. \tag{28}$$

In particular, the decision-metric eigenvalue D_l depends on the target only through the sign of γ_l, and not through its magnitude. For admissible Y, on every mode $l \geq 2$ (active or not),

$$D_l\, \sigma_l^2 \;=\; \big[\,\gamma_l - \kappa_l\,\big]_+. \tag{29}$$

Proof. By Proposition 8, on an active mode $\sigma_l^2 \;=\; |\gamma_l|\sqrt{\lambda_W/(\lambda_M s_l)} - \lambda_W$, so $\sigma_l^2 + \lambda_W \;=\; |\gamma_l|\sqrt{\lambda_W/(\lambda_M s_l)}$. Substituting into $D_l = \gamma_l/(\sigma_l^2 + \lambda_W)$ and cancelling $|\gamma_l|$ gives the first identity. For the second,

$$D_l\, \sigma_l^2 \;=\; \operatorname{sign}(\gamma_l)\sqrt{\tfrac{\lambda_M s_l}{\lambda_W}}\Big(|\gamma_l|\sqrt{\tfrac{\lambda_W}{\lambda_M s_l}} - \lambda_W\Big) \;=\; \gamma_l \;-\; \operatorname{sign}(\gamma_l)\sqrt{\lambda_M \lambda_W\, s_l},$$

using $\operatorname{sign}(\gamma_l)\,|\gamma_l| = \gamma_l$ in the first term. For equation 29, admissibility gives $\gamma_l \geq 0$. On an active mode, $\gamma_l > \kappa_l$ by the threshold characterization of §I.6, and the second identity reads $D_l\sigma_l^2 = \gamma_l - \kappa_l = [\gamma_l - \kappa_l]_+$. On an inactive mode, $\sigma_l = 0$ and $\gamma_l \leq \kappa_l$, so both sides vanish. □

Recall that

$$A \;=\; R^\top D R, \qquad D = \operatorname{diag}\big(D_2, \ldots, D_K, 0, \ldots, 0\big)$$

and R is the same orthogonal factor as in $M^* = U\Sigma R$ (Proposition 8). The squared semi-distance is

$$\|m_i^* - m_j^*\|_A^2 \;=\; (e_i - e_j)^\top M^* A\, M^{*\top}(e_i - e_j).$$

Substituting both factorizations and using $RR^\top = I_d$ to cancel the feature-space rotation,

$$M^* A\, M^{*\top} \;=\; U\Sigma R \cdot R^\top D R \cdot R^\top \Sigma^\top U^\top \;=\; U\,\Sigma D \Sigma^\top\, U^\top = U\ \operatorname{diag}\big(0, \sigma_2^2 \cdot D_2, \ldots, \sigma_K^2 \cdot D_K\big)\ U^\top.$$

This is an eigendecomposition in an $L_{\mathcal{T}}$-eigenbasis: the closed forms for σ_l^2 and γ_l depend on l only through ν_l (the tree Laplacian's eigenvalues, §I.1), so $M^*AM^{*\top}$ acts as a scalar multiple of the identity on each $L_{\mathcal{T}}$-eigenspace. Thus, $M^*AM^{*\top}$ commutes with $L_{\mathcal{T}}$. Since the l-th coordinate of $U^\top(e_i - e_j)$ is $\phi_l(i) - \phi_l(j)$,

$$\|m_i^* - m_j^*\|_A^2 \;=\; \sum_{l=2}^{K} D_l\, \sigma_l^2\, \big(\phi_l(i) - \phi_l(j)\big)^2. \tag{30}$$

Substituting equation 29 into the eigendecomposition above,

$$M^* A M^{*\top} \;=\; \sum_{l \geq 2} \big[\gamma_l - \kappa_l\big]_+\, \phi_l \phi_l^\top,$$

which is part (iii) with $c_l = [\gamma_l - \kappa_l]_+$. Note that padding D as described above does not change $M^*AM^{*\top}$, since Σ vanishes on the padded directions. This completes the proof of Theorem 4.

J HIERARCHICAL NEURAL COLLAPSE: HNC1–HNC4

This appendix states and proves the hierarchical analogues HNC1–HNC4 of the classical neural collapse phenomena (§5.1). HNC1 concerns the per-sample training loss and justifies the reduction to the class-mean objective equation 4 claimed in §5; HNC2–HNC4 are read off Proposition 8 and Theorem 4.

J.1 NOTATION & THE PER-SAMPLE OBJECTIVES

We consider K-class classification with exactly n training samples per class and feature dimension $d \geq K-1$. Let $h_{i,k} \in \mathbb{R}^d$ denote the penultimate-layer feature of the i-th sample in class k, and $m_k = \frac{1}{n}\sum_{i=1}^{n} h_{i,k}$ its class-k mean. The class-mean matrix $M \in \mathbb{R}^{K\times d}$ collects these means as rows (k-th row $m_k^\top$), and the linear head is $W \in \mathbb{R}^{K\times d}$. Strictly, the optimization variables of the empirical UFM are the per-sample features $H = (h_{i,k})$ and W; however, we will establish a within-class collapse result that allows us to work instead with M. Regression targets are collected in $Y \in \mathbb{R}^{K\times K}$, with the k-th row $y_k^\top$ assigned to class k. We write $\bar{m} := \frac{1}{nK}\sum_{i,k} h_{i,k}$ for the global feature mean and $\hat{Y} := \frac{1}{K}\sum_k y_k$ for the mean target vector, and retain the conventions of §I.1.

We begin by stating the *bias-carrying empirical objective*, which represents the loss actually minimized during training:

$$\begin{aligned}\mathcal{J}_Y^{(\text{emp,bias})}(H,W,b) \;=\; & \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n}\|Wh_{i,k}+b-y_k\|^2 \\ & + \lambda_W\|W\|_F^2 + \lambda_M \operatorname{tr}\big(M^\top S M\big), \qquad \lambda_W,\lambda_M>0,\end{aligned} \tag{31}$$

where $b \in \mathbb{R}^K$ is a bias, and $y_k \in \mathbb{R}^K$ the regression target for class k. As in §5, $S \in \{I, L_{\mathcal{T}}\}$, and in both cases the regularizer acts on the class means $M = M(H)$. The UFM framework (§5.1) does not allow for the bias. However, the optimal bias is a function of the optimal features and weights; substituting this optimal bias and centering the targets yields the *bias-free empirical objective*,

$$\begin{aligned}\mathcal{J}_{\bar{Y}}^{(\text{emp})}(H,W) \;=\; & \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n}\big\|Wh_{i,k}-\bar{y}_k\big\|^2 + \lambda_W\,\|W\|_F^2 + \lambda_M\,\operatorname{tr}\big(M^\top S\,M\big), \\ \text{s.t.}\quad & \frac{1}{nK}\sum_{i,k} h_{i,k} = 0, \quad \lambda_W,\lambda_M>0,\end{aligned} \tag{32}$$

where $\bar{Y}$ and $\bar{y}_k$ denote *centered* regression targets (i.e., the original targets shifted such that they sum to zero). The bias-carrying and bias-free objectives share the same minimizers up to a global translation of the features, a correspondence §J.2 makes precise. We study the bias-free objective throughout, with the understanding that its minimizers recover the bias-carrying objective's minimizers up to translation. Because classes are balanced, the centering constraint $\frac{1}{nK}\sum_{i,k} h_{i,k} = 0$ is exactly $M^\top \mathbb{1} = 0$. If every sample sits at its class mean, $h_{i,k} = m_k$, the data-fit term of equation 32 becomes $\sum_k \|Wm_k - \bar{y}_k\|^2 = \|MW^\top - \bar{Y}\|_F^2$, and equation 32 coincides with the class-mean objective $\mathcal{J}_{\bar{Y}}(M,W)$ of equation 4. HNC1 (§J.3) shows that the minimizers of equation 32 can always be taken in this form.

The objective we present is a generalization of the MSE loss studied in the UFM by Zhou et al. (2022). In particular, our Laplacian regularizer $\lambda_M \operatorname{tr}(M^\top L_{\mathcal{T}} M)$ reduces to feature weight decay when $S = I$. (Zhou et al. (2022) apply weight decay to each sample feature, $\sum_{i,k}\|h_{i,k}\|^2$, rather than to the class means. Our analysis covers that variant as well: per-sample decay is strictly convex in each $h_{i,k}$ and dominates $n\operatorname{tr}(M^\top M)$, with equality if and only if every sample sits at its class mean, so the Jensen argument of Lemma 19 goes through with equality only at collapsed configurations, and HNC1's uniqueness clause holds.)

J.2 BIAS-ELIMINATION AND TARGET CENTERING

This section examines the relationship between the bias-carrying empirical objective equation 31 and the bias-free empirical objective equation 32. The *downward* direction eliminates the bias b in

closed form, after which the two objectives coincide as functions of (H, W); the *upward* direction restores it, lifting each bias-free optimum to a one-parameter family of bias-carrying optima when $S = L_{\mathcal{T}}$, and a single point when $S = I$. Both directions are packaged in Lemma 17 below.

We show that, after centering H, every minimizer (H, W) of the bias-carrying objective is a minimizer of the bias-free objective; conversely, every minimizer (H, W) of the bias-free objective lifts to a minimizer of the bias-carrying objective under arbitrary translations of H when $S = L_{\mathcal{T}}$, and under the trivial translation only when $S = I$. In the bias-carrying objective, the optimal bias b is uniquely determined by the optimal H and W. Formally, we establish the following:

Lemma 17. *For any target $Y \in \mathbb{R}^{K\times K}$, (H, W, b) minimizes $\mathcal{J}_Y^{(emp,bias)}$ if and only if $(H - \mathbb{1}\bar{m}^\top, W)$ minimizes $\mathcal{J}_{\bar{Y}}^{(emp)}$, with optimal bias $b = \hat{Y} - W\bar{m}$ and, when $S = I$, additionally $\bar{m} = 0$ (so that $b = \hat{Y}$). Here, $\bar{m}$ denotes the mean feature across all of H and $\hat{Y}$ the mean target vector.*

We proceed in three steps. Step 1 shows that the optimal bias is a function of H and W, allowing us to eliminate it. Further, after eliminating the bias, the objectives with and without centering coincide as functions of H and W, and hence share the same minimizers. Step 2 establishes the correspondence between minimizers of the bias-carrying centered form and minimizers of the bias-free constrained form via a centering map on H. Step 3 is a conclusion.

Throughout, we write $m_k = \frac{1}{n}\sum_i h_{i,k}$ for class k's mean feature, and M is the matrix of these class means. As in the statement of the lemma, $\bar{m} = \frac{1}{nK}\sum_{i,k} h_{i,k}$ is the global feature mean.

Step 1: centering Y does not affect the bias-carrying minimizer set. The bias-carrying objective is a convex quadratic in b, with

$$\nabla_b \mathcal{J}_Y^{(\mathrm{emp,bias})} \;=\; 2K\big(W\bar{m} + b - \hat{Y}\big)$$

Setting this to zero yields the minimizer $b^\star(H, W) = \hat{Y} - W\bar{m}$. Substituting $b^\star$ into the data-fit term, each residual becomes

$$Wh_{i,k} + b^\star - y_k \;=\; W(h_{i,k} - \bar{m}) - (y_k - \hat{Y}) \;=\; W\tilde{h}_{i,k} - \bar{y}_k,$$

with $\tilde{h}_{i,k} = h_{i,k} - \bar{m}$ and $\bar{y}_k = y_k - \hat{Y}$. Hence

$$\min_b \mathcal{J}_Y^{(\mathrm{emp,bias})}(H, W, b) \;=\; \frac{1}{n}\sum_{k,i}\big\|W\tilde{h}_{i,k} - \bar{y}_k\big\|^2 \;+\; \lambda_W\|W\|_F^2 \;+\; \lambda_M \operatorname{tr}\big(M^\top S M\big), \quad (33)$$

which depends on Y only through $\bar{Y}$. Repeating the calculation with $\bar{Y}$ in place of Y yields the same right-hand side of equation 33. Therefore

$$\min_b \mathcal{J}_Y^{(\mathrm{emp,bias})}(H, W, b) \;=\; \min_b \mathcal{J}_{\bar{Y}}^{(\mathrm{emp,bias})}(H, W, b) \qquad \text{for all } (H, W).$$

Thus, eliminating the bias yields two objectives which coincide as functions of H and W. As a consequence, they share minimizers.

Step 2: minimizer correspondence between bias-carrying centered and bias-free constrained forms. Fix any minimizer (H, W, b) of the bias-carrying empirical objective with centered targets. We exhibit (H', W') such that

$$\mathcal{J}_{\bar{Y}}^{(\text{emp, bias})}(H, W, b) = \mathcal{J}_{\bar{Y}}^{(\mathrm{emp})}(H', W'),$$

where (recall) the bias-free objective on the right requires centered features. Since the bias-carrying objective has centered targets, each residual in its data-fit term re-writes as

$$Wh_{i,k} + b - \bar{y}_k \;=\; W(h_{i,k} - \bar{m}) - \bar{y}_k + (b + W\bar{m}).$$

Since (W, H, b) is a minimizer, step 1 tells us that the bias must be $b = \hat{\bar{Y}} - W\bar{m}$. Since $\bar{Y}$ is already centered, its mean $\hat{\bar{Y}}$ is zero, and so $b = -W\bar{m}$. In the above, this tells us that the residuals are simply

$$W(h_{i,k} - \bar{m}) - \bar{y}_k.$$

Collect the centered features $h_{i,k} - \bar{m}$ into H', and take $W' = W$. Then the objectives $\mathcal{J}_{\bar{Y}}^{(\text{emp, bias})}(H, W, b)$ and $\mathcal{J}_{\bar{Y}}^{(\text{emp})}(H', W')$ have the same residuals in their data-fit terms, and hence their data-fit terms are equal. The other two terms of the objectives are equal, too. For the weight decay term, this is trivial since $W = W'$. For the Laplacian regularizer, note that $L_{\mathcal{T}}\mathbb{1} = 0$ kills the shift $M \mapsto M - \mathbb{1}\bar{m}^\top$ in $\operatorname{tr}(M^\top L_{\mathcal{T}} M)$. The $S = I$ case requires more care. Writing $M = M' + \mathbb{1}\bar{m}^\top$ with $M'^\top \mathbb{1} = 0$ the class means of H', we have $\operatorname{tr}(M^\top M) = \operatorname{tr}(M'^\top M') + K\|\bar{m}\|^2$, and hence

$$\mathcal{J}_{\bar{Y}}^{(\text{emp, bias})}(H, W, b) \;=\; \mathcal{J}_{\bar{Y}}^{(\text{emp})}(H', W') + \lambda_M K\|\bar{m}\|^2 \;\geq\; \mathcal{J}_{\bar{Y}}^{(\text{emp})}(H', W').$$

But $(H', W', 0)$ is itself feasible for the bias-carrying objective, with value $\mathcal{J}_{\bar{Y}}^{(\text{emp})}(H', W')$ (its residuals are the same, and its class means are M'). Minimality of (H, W, b) therefore forces $\bar{m} = 0$, so that $H' = H$, $b = 0$, and the two objective values agree in this case too.

Step 3: Conclusion. Composing the two steps proves the lemma. By Step 1, after the bias is eliminated, the bias-carrying objectives with regression targets Y and $\bar{Y}$ coincide as functions of (H, W); they therefore share their minimizers, and at any such minimizer the optimal bias is $b^\star = \hat{Y} - W\bar{m}$. By Step 2, the centering map $H \mapsto H' = H - \mathbb{1}\bar{m}^\top$ places the minimizers of the bias-carrying centered objective $\mathcal{J}_{\bar{Y}}^{(\text{emp,bias})}$ in value-preserving correspondence with the minimizers of the bias-free objective $\mathcal{J}_{\bar{Y}}^{(\text{emp})}$, whose feasible features are centered. Chaining the two correspondences, (H, W, b) minimizes $\mathcal{J}_{Y}^{(\text{emp,bias})}$ if and only if $(H - \mathbb{1}\bar{m}^\top, W)$ minimizes $\mathcal{J}_{\bar{Y}}^{(\text{emp})}$, with the bias pinned to $b = \hat{Y} - W\bar{m}$ and, when $S = I$, additionally $\bar{m} = 0$ (so that $b = \hat{Y}$); without the latter condition the "if" direction fails for $S = I$, since translating a bias-free minimizer by $c \neq 0$ costs $\lambda_M K\|c\|^2$. This is exactly Lemma 17.

J.3 HNC1: Collapse to Class Means

Classical NC1 states that, in the terminal phase of training, every training sample is mapped to its class' mean feature. Its hierarchical analogue, HNC1, is essentially the *same* statement about a *different* loss. Admissible targets are already centered ($\hat{Y} = 0$, so $\bar{Y} = Y$); when the bias-carrying loss is trained against an uncentered target such as one-hot $Y = I_K$, Lemma 17 replaces it with $\bar{Y} = J$, which is admissible with $\gamma_l = 1$ for $l \geq 2$. We therefore state HNC1 for admissible Y and the bias-free objective equation 32, and write Y for $\bar{Y}$.

Theorem 18 (Intra-class Variability Collapse, HNC1)**.** *Let Y be admissible and $\lambda_W, \lambda_M > 0$. Then there exists a global minimizer (H^*, W^*) of the bias-free empirical objective equation 32 with*

$$h_{i,k}^* = m_k^*,$$

where m_k^* is the mean class-k feature. Moreover, if the feature dimension d equals $K - 1$ and $\operatorname{rank}(M^*) = K - 1$, then any global minimizer must have $h_{i,k}^* = m_k^*$ for all i and k.

Theorem 18 consists of two claims about the bias-free empirical objective equation 32:

1. *Existence.* There exists a global minimizer (H^*, W^*) with $h_{i,k}^* = m_k^*$ for every i, k.
2. *Uniqueness.* If $d = K - 1$ and $\operatorname{rank}(M^*) = K - 1$, then any global minimizer has $h_{i,k}^* = m_k^*$.

The *existence* claim lets us step down from the per-sample objective to the class-mean objective: every empirical optimum can be collapsed onto its class means without increasing the objective, so we can move down to the class-mean objective. The *uniqueness* claim is the reverse direction: when $d = K - 1$ and $\operatorname{rank}(M^*) = K - 1$, the class-mean optimum yields a *unique* empirical optimum, allowing us to step back up to the per-sample objective.

We organize the argument as follows. We begin with a useful technical lemma in §J.3.1. Then, we establish the *existence* claim in §J.3.2, and finally establish the *uniqueness* claim in §J.3.3.

J.3.1 Applying Jensen's Inequality

Say we have features $H = (h_{i,k})_{i,k}$ (with K classes of n samples each), and write $m_k(H) := \frac{1}{n}\sum_{i=1}^{n} h_{i,k}$ for the induced class means and $M(H) \in \mathbb{R}^{K\times d}$ for the class-mean matrix whose k-th row is $m_k(H)^\top$. Let Π be the map that maps each feature $h_{i,k}$ to $m_k(H)$, its class' mean. Then Π preserves the class-mean matrix: $M(\Pi(H)) = M(H)$. In particular, Π preserves the centering constraint $M^\top \mathbb{1} = 0$.

Our main result here is the following straightforward application of Jensen's inequality:

Lemma 19. *For every (H, W),*

$$\mathcal{J}_Y^{(\mathrm{emp})}(\Pi(H, W)) \le \mathcal{J}_Y^{(emp)}(H, W),$$

with equality if and only if, for every class k, $Wh_{1,k} = Wh_{2,k} = \cdots = Wh_{n,k}$.

Proof. We examine how applying the map $\Pi(\cdot)$ to H affects the empirical objective. Since $\Pi(M) = M$, this does not affect the regularizer $\lambda_M \operatorname{tr}(M^\top S M)$. The weight-decay term only depends on W, and is similarly unaffected by the action of Π. Thus, the only objective term that Π can affect is the data-fit term. For all i and k, the function $h_{i,k} \mapsto \|Wh_{i,k} - y_k\|^2$ is convex in $h_{i,k}$, with Hessian $2W^\top W \succeq 0$. Thus, for each class k, we can apply Jensen's inequality to obtain

$$\frac{1}{n}\sum_{i=1}^{n} \|Wh_{i,k} - y_k\|^2 \;\ge\; \frac{1}{n}\sum_{i=1}^{n} \|W \cdot \Pi(h_{i,k}) - y_k\|^2. \tag{34}$$

Summing over classes yields the inequality

$$\frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n} \|Wh_{i,k} - y_k\|^2 \ge \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n} \|W \cdot \Pi(h_{i,k}) - y_k\|^2,$$

from which the desired inequality

$$\mathcal{J}_Y^{(\mathrm{emp})}(\Pi(H, W)) \le \mathcal{J}_Y^{(\mathrm{emp})}(H, W)$$

now follows. It remains to establish the equality condition. Recall that Jensen's inequality tells us that the inequality equation 34 is tight if and only if the convex function $h_{i,k} \mapsto \|Wh_{i,k} - y_k\|^2$ is affine on the convex hull of $\{h_{1,k}, \ldots, h_{n,k}\}$. Since the function has Hessian $2W^\top W$, this holds if and only if $W^\top W(h_{i,k} - h_{j,k}) = 0$ for all $i \ne j$; equivalently, if and only if $h_{i,k} - h_{j,k} \in \ker(W^\top W) = \ker W$; equivalently, if and only if $Wh_{i,k} = Wh_{j,k}$ for all i, j. □

J.3.2 Existence of an HNC1 minimizer

We now establish the *existence* claim by showing that the empirical objective has a minimizer. Lemma 19 tells us that replacing every sample feature $h_{i,k}$ with its class' mean feature m_k will not increase the value of the objective. Thus, to show that the empirical objective has a global minimizer with $h^*_{i,k} = m^*_k$ for all i and k, we need only show that the class mean objective (equation 4) has a global minimizer. It does, by Lemma 9.

Let (M^*, W^*) attain the class-mean objective's global minimum. Define H^* by $h^*_{i,k} := m^*_k$ for all i, k. This configuration has class-mean matrix M^*, so $\mathcal{J}_Y^{(\mathrm{emp})}(H^*, W^*) = \mathcal{J}_Y(M^*, W^*)$. By Lemma 19, this is a global minimizer of the empirical objective. In particular the two objectives share their minimum value: $\min \mathcal{J}_Y^{(\mathrm{emp})} = \min \mathcal{J}_Y$.

J.3.3 Conditions for the Uniqueness of HNC1 Minimizers

Now, we establish the *uniqueness* claim. First, note that if $n = 1$ (i.e., we have one sample per class) then $h_{i,k} = m_k$ holds trivially, and the result is true. Hereafter, suppose that $n \ge 2$. Let (H^*, W^*) be a global minimizer of $\mathcal{J}_Y^{(\mathrm{emp})}$, and let $M^* = M(H^*)$ be the induced class-mean matrix.

We begin by noting that the map $W \mapsto \mathcal{J}_Y^{(\text{emp})}(H^*, W)$ is a strictly convex quadratic with first-order optimality condition $W^*\big(\frac{1}{n}H^{*\top}H^* + \lambda_W I_d\big) = Y^\top M^*$, where $H^* \in \mathbb{R}^{nK\times d}$ stacks the sample features as rows. Since $\frac{1}{n}H^{*\top}H^* + \lambda_W I_d \succeq \lambda_W I_d \succ 0$ is positive-definite, hence invertible,

$$W^* \;=\; Y^\top M^* \left(\tfrac{1}{n}H^{*\top}H^* + \lambda_W I_d\right)^{-1}, \tag{35}$$

and hence $\operatorname{rank}(W^*) \le \operatorname{rank}(M^*) \le K-1$, where the second inequality is the centering constraint $M^{*\top}\mathbb{1} = 0$.

Now, say $d = K-1$ and $\operatorname{rank}(M^*) = K-1$. Then $\operatorname{rank}(W^*) \le K-1$. From the closed form of W^* (specifically, the invertibility of its right factor), we have $\operatorname{rank}(W^*) = \operatorname{rank}(Y^\top M^*)$. We claim that $\operatorname{rank}(Y^\top M^*) = \operatorname{rank}(M^*) = K-1$. By symmetry, $\ker(Y^\top) = \ker(Y)$; as shown above, $\operatorname{rank}(M^*) = K-1$ forces $\ker(Y) = \operatorname{span}\{\mathbb{1}\}$. Centering gives $\operatorname{range}(M^*) \subseteq \mathbb{1}^\perp$, so $\ker(Y^\top) \cap \operatorname{range}(M^*) = \{0\}$. Thus, $\operatorname{rank}(W^*) = \operatorname{rank}(Y^\top M^*) = \operatorname{rank}(M^*) = K-1$.

Since $d = K-1$, the map $W^* : \mathbb{R}^d \to \mathbb{R}^K$ has rank equal to its domain's dimension, and hence $\ker W^* = \{0\}$. As (H^*, W^*) is a global minimizer, $\mathcal{J}_Y^{(\text{emp})}(\Pi(H^*, W^*)) \ge \mathcal{J}_Y^{(\text{emp})}(H^*, W^*)$, while Lemma 19 gives the reverse inequality; the two therefore agree, so the lemma's equality condition holds: $W^*h^*_{i,k} = W^*h^*_{j,k}$ for all i, j and every class k. Since $\ker W^* = \{0\}$, this forces $h^*_{i,k} = h^*_{j,k}$ for all i, j, so every class-k sample feature equals the class mean m^*_k, as claimed.

For the converse, we establish the contrapositive: if $d > K-1$ or $\operatorname{rank}(M^*) < K-1$, then $\mathcal{J}_Y^{(\text{emp})}$ admits a non-HNC1 global minimizer. In the first case, $\operatorname{rank}(W^*) \le K-1 < d$, so $\ker W^* \subseteq \mathbb{R}^d$ is non-trivial. In the second, $\operatorname{rank}(W^*) \le \operatorname{rank}(M^*) < K-1 \le d$, again giving $\ker W^* \neq \{0\}$. Pick any $v \in \ker W^* \setminus \{0\}$ and any $\epsilon_1, \dots, \epsilon_n \in \mathbb{R}$ with $\sum_i \epsilon_i = 0$ and not all ϵ_i zero (possible since $n \ge 2$). The perturbed sample features $h'_{i,k} = h^*_{i,k} + \epsilon_i v$ preserve class means and $W^*h'_{i,k} = W^*h^*_{i,k}$, so $\mathcal{J}_Y^{(\text{emp})}(H', W^*) = \mathcal{J}_Y^{(\text{emp})}(H^*, W^*)$. The configuration (H', W^*) is thus a distinct global minimizer.

J.4 HNC2: Optimal Class-Mean Layout

Classical NC2 is a statement about the layout of the class' mean features at the optimum. HNC2 is a description of the same for our hierarchy-aware objective.

Claim. At a minimizer (M^*, W^*) of the class-mean objective, the pairwise feature-space distances admit the spectral decomposition

$$\|m^*_i - m^*_j\|_2^2 \;=\; \sum_{l\ge 2} \sigma_l^2 \,(\phi_l(i) - \phi_l(j))^2, \tag{36}$$

where m^*_k is the k-th row of M^*. (See §I.1 for the definition of $\phi_l(k)$.)

Derivation. The factorization $M^* = U\Sigma R$ yields $G^* = U\Sigma\Sigma^\top U^\top$, where $\Sigma\Sigma^\top \in \mathbb{R}^{K\times K}$ is diagonal with entries σ_l^2. Thus,

$$G^* = U\,\operatorname{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_K^2)\,U^\top.$$

This is an eigendecomposition of G^* in an $L_{\mathcal{T}}$-eigenbasis. G^* shares its eigenspaces with $L_{\mathcal{T}}$ and hence commutes with it. Moreover, the closed-form expressions for σ_l^2 (equation 18) show that $\sigma_l = \sigma_{l'}$ whenever $\nu_l = \nu_{l'}$. Thus, when restricted to any eigenspace E_j of $L_{\mathcal{T}}$, G^* acts as the scalar σ_j^2 times the identity; equivalently, the E_j-block of G^* in the U-basis is $\sigma^2_{(j)} I_{m_j}$.

For the pairwise decomposition, write $m^*_i - m^*_j = (M^*)^\top(e_i - e_j)$, so

$$\|m^*_i - m^*_j\|_2^2 \;=\; (e_i - e_j)^\top G^*(e_i - e_j) \;=\; (e_i - e_j)^\top U\,\operatorname{diag}(\sigma_l^2)U^\top(e_i - e_j).$$

Setting $v := U^\top(e_i - e_j) \in \mathbb{R}^K$, whose l-th coordinate is $\phi_l^\top(e_i - e_j) = \phi_l(i) - \phi_l(j)$, the right-hand side becomes $\sum_{l=1}^K \sigma_l^2(\phi_l(i) - \phi_l(j))^2$. The $l = 1$ term drops because $\sigma_1 = 0$, yielding equation 36.

J.5 HNC3: Spectral Self-Duality

Classical NC3 establishes that a trained classifier W and the centered class-mean matrix align up to a single scalar: each classifier row is proportional to its corresponding class mean feature, with a proportionality constant independent of k. The hierarchical analogue weakens this in a structured way. The classifier and class means still align, sharing left and right factors U and R (Proposition 8), but the alignment now takes the form

$$W^*(M^*)^\top \;=\; U\,\mathrm{diag}\!\left(\frac{\sigma_l^2}{\sigma_l^2+\lambda_W}\right)\Gamma\,U^\top, \tag{37}$$

where Γ is Y's spectrum in the U-basis (§I.1). In particular, W^* and M^* have the same active-mode support: $\theta_l = 0$ iff $\sigma_l = 0$.

Derivation. We establish equation 37 through direct computation. By Proposition 8's factorizations $M^* = U\Sigma R$ and $W^* = U\Theta R$,

$$W^*(M^*)^\top \;=\; U\Theta R\cdot R^\top\Sigma^\top U^\top \;=\; U(\Theta\Sigma^\top)U^\top,$$

where $RR^\top = I_d$. The product $\Theta\Sigma^\top$ is diagonal with l-th diagonal entry $\theta_l\sigma_l$ for $1 \le l \le K$. Substituting Proposition 8's expression $\theta_l = \gamma_l\sigma_l/(\sigma_l^2+\lambda_W)$:

$$\theta_l\sigma_l \;=\; \frac{\gamma_l\,\sigma_l^2}{\sigma_l^2+\lambda_W} \;=\; \frac{\sigma_l^2}{\sigma_l^2+\lambda_W}\cdot\gamma_l.$$

Writing the result as a product of two diagonal matrices,

$$\Theta\Sigma^\top \;=\; \mathrm{diag}\!\left(\frac{\sigma_l^2}{\sigma_l^2+\lambda_W}\right)\cdot\Gamma,$$

recovers equation 37 after conjugating by U.

For the active-mode support claim: θ_l is proportional to σ_l (with proportionality $\gamma_l/(\sigma_l^2+\lambda_W)$). On active modes ($\sigma_l > 0$), Proposition 8's threshold characterization requires $\gamma_l \neq 0$, so the proportionality is non-zero and $\theta_l \neq 0$. On inactive modes ($\sigma_l = 0$), $\theta_l = 0$ by direct substitution. Hence $\theta_l = 0$ iff $\sigma_l = 0$.

The exact form. Theorem 4(i) sharpens equation 37: the alignment is exact, $W^* = M^*A$, once the classifier is read through the decision metric A of equation 25. On active modes A's eigenvalues $D_l = \sqrt{\lambda_M s_l/\lambda_W}$ do not depend on the target at all (Lemma 16), so the shrinkage factors in equation 37 are exactly what remains once A is folded into M^*; indeed the diagonal entries of equation 37 are $\gamma_l\sigma_l^2/(\sigma_l^2+\lambda_W) = D_l\sigma_l^2 = c_l$, and $W^*M^{*\top} = M^*AM^{*\top}$ is the A-Gram of Theorem 4(iii). For $S = I$ with every mode active, A is a multiple of the identity and equation 37 reduces to classical NC3.

J.6 HNC4: Tree-Aware Nearest-Class-Mean Classification

Classical NC4 establishes that, at the optimum, a trained classifier performs nearest-class-mean classification in feature space using the Euclidean distance. Its hierarchical analogue retains this structure, with a key difference: the Euclidean distance is replaced by a Mahalanobis semi-distance,

$$\arg\max_k \langle w_k^*, h\rangle \;=\; \arg\min_k\big(\|h-m_k^*\|_A^2 - m_k^{*\top}A\,m_k^*\big), \tag{38}$$

where $A = R^\top DR$ is the metric of equation 25, with $D_l = \gamma_l/(\sigma_l^2+\lambda_W)$, and

$$\|x-y\|_A^2 = (x-y)^\top A(x-y).$$

Here, R is the same matrix as in the factorization of M^* and W^*. Part (ii) of Theorem 4 is equation 38, proved in §I.9. For $d = K-1$ and $\gamma_l > 0$ on every mode, A is positive-definite and $\|\cdot\|_A$ is a genuine norm; otherwise it is a semi-norm unless padded as described there. When the tree is *leaf-transitive*—that is, every class is exchangeable with every other class under a weight-preserving symmetry—the bias term is constant across classes and cancels from the decision rule. We prove this next.

Bias constancy under leaf-transitivity. Adopting the notation of §I.1, the calculation

$$m_k^{*\top} A\, m_k^* \;=\; \sum_{l\geq 2} \phi_l(k)^2 \cdot \frac{\gamma_l\, \sigma_l^2}{\sigma_l^2 + \lambda_W} \tag{39}$$

shows the bias's only k-dependence is through the eigenvector coordinates $\phi_l(k)$. The per-mode weights $\gamma_l\sigma_l^2/(\sigma_l^2+\lambda_W)$ depend on l only through ν_l (since both γ_l and σ_l are functions of ν_l under Proposition 8's hypotheses), so they are constant on each $L_{\mathcal{T}}$-eigenspace E_j; the bias therefore reduces to

$$m_k^{*\top} A\, m_k^* \;=\; \sum_{j\geq 2} c_j \cdot \sum_{l\in E_j} \phi_l(k)^2 \;=\; \sum_{j\geq 2} c_j \cdot \|P_{E_j} e_k\|^2,$$

where c_j is the common weight value on E_j and P_{E_j} is the orthogonal projection onto E_j (using $\sum_{l\in E_j}\phi_l(k)^2 = \langle e_k, P_{E_j} e_k\rangle$).

A graph automorphism of $\mathcal{T}$ is a bijection π of its vertices that preserves adjacency and edge weights. On the leaf sets, π restricts to a permutation, and we can represent its action on $\mathbb{R}^K$ as a permutation matrix $P_\pi \in O(K)$ satisfying $P_\pi e_k = e_{\pi(k)}$. Because π preserves adjacency and edge weights, P_π commutes with the Laplacian: $P_\pi L_{\mathcal{T}} P_\pi^\top = L_{\mathcal{T}}$. Commutation with $L_{\mathcal{T}}$ implies commutation with the spectral projections P_{E_j} onto each eigenspace of $L_{\mathcal{T}}$. (Commutation $P_\pi L_{\mathcal{T}} = L_{\mathcal{T}} P_\pi$ means P_π maps $L_{\mathcal{T}}$-eigenspaces to $L_{\mathcal{T}}$-eigenspaces of the same eigenvalue, i.e., preserves each E_j setwise, which is the same as commuting with the projection onto E_j.)

Orthogonality of P_π and commutation with P_{E_j} now give

$$\|P_{E_j} e_{\pi(k)}\|^2 \;=\; \|P_{E_j} P_\pi e_k\|^2 \;=\; \|P_\pi P_{E_j} e_k\|^2 \;=\; \|P_{E_j} e_k\|^2.$$

Each summand in the bias is therefore invariant under an automorphism's action on k. Leaf-transitivity — the existence, for any pair of leaves (k, k'), of an automorphism mapping k to k' — then forces $m_k^{*\top} A\, m_k^*$ to be constant in k, and the bias drops from $\arg\max_k$.

The bias for decision-faithful targets. The main text observes that under the decision-faithful targets the per-class offsets are the diagonal of $L_{\mathcal{T}}^\dagger$, and interprets that diagonal as an average taxonomic distance. We justify that interpretation with the following lemma.

Lemma 20. *Let Y be the decision-faithful target equation 5 for the chosen S, and let (M^*, W^*) be any minimizer of $\mathcal{J}_Y$ with decision metric A. Then $M^* A M^{*\top} = L_{\mathcal{T}}^\dagger$, so the offset of class k in equation 38 is*

$$\|m_k^*\|_A^2 \;=\; \big(L_{\mathcal{T}}^\dagger\big)_{kk} \;=\; \frac{1}{K}\sum_{v=1}^{K} d_{\mathcal{T}}(k,v) \;-\; \frac{1}{K}\,\mathrm{tr}\big(L_{\mathcal{T}}^\dagger\big).$$

In particular the offsets are constant across classes if and only if every class has the same average taxonomic distance to the others.

Proof. For the decision-faithful targets $c_l = [\gamma_l - \kappa_l]_+ = \nu_l^{-1}$ for every $l \geq 2$ (§I.9), so Theorem 4(iii) gives $M^* A M^{*\top} = \sum_{l\geq 2} \nu_l^{-1}\phi_l\phi_l^\top = L_{\mathcal{T}}^\dagger$ (Lemma 6). The offset is the k-th diagonal entry. For the second equality, Lemma 6 gives $d_{\mathcal{T}}(k,v) = (L_{\mathcal{T}}^\dagger)_{kk} + (L_{\mathcal{T}}^\dagger)_{vv} - 2(L_{\mathcal{T}}^\dagger)_{kv}$; summing over v,

$$\sum_v d_{\mathcal{T}}(k,v) \;=\; K\,(L_{\mathcal{T}}^\dagger)_{kk} + \mathrm{tr}\big(L_{\mathcal{T}}^\dagger\big) - 2\,\big(L_{\mathcal{T}}^\dagger \mathbb{1}\big)_k \;=\; K\,(L_{\mathcal{T}}^\dagger)_{kk} + \mathrm{tr}\big(L_{\mathcal{T}}^\dagger\big),$$

since $L_{\mathcal{T}}^\dagger \mathbb{1} = 0$. Rearranging gives the claim; the trace term is independent of k. □

K Realizing Hierarchical Neural Collapse Empirically

This appendix supplements our neural collapse discussion (§5). The main text of the paper includes theoretical results obtained by analyzing a certain training objective in the unconstrained features model (UFM). The UFM is a simplified analytic model of real encoders, and any results obtained

Table 10: Summary of our neural-collapse runs. η is the peak learning rate (scaled linearly in batch size, relative to 128). Training accuracy is measured at the final epoch.

Dataset	Encoder	Target Y	S	λ_M	λ_W	η	Epochs	Train acc. (%)
CIFAR-100	ResNet-18	Dec. faithful	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.09	350	99.97
CIFAR-100	ResNet-18	Eucl. faithful	$L_{\mathcal{T}}$	10^{-3}	10^{-4}	0.028	350	99.98
CIFAR-100	ResNet-18	One-hot	$L_{\mathcal{T}}$	10^{-3}	10^{-4}	0.09	350	99.69
CIFAR-100	ResNet-18	Resistance	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.14	350	99.95
CIFAR-100	DenseNet-250	Dec. faithful	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.05	350	99.86
FGVC-Aircraft	ResNet-18	Dec. faithful	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.13	2500	100.00
Stanford Cars	ResNet-18	Dec. faithful	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.13	2500	99.18
CIFAR-100	ResNet-18	Dec. faithful	I	10^{-4}	10^{-4}	0.10	350	99.97
FGVC-Aircraft	ResNet-18	Dec. faithful	I	10^{-4}	10^{-4}	0.13	2500	99.1

through it are only meaningful if their predicted outcomes occur in real models. Following existing neural collapse works (Mixon et al., 2020), we empirically confirm that the predicted feature and classifier head geometries arise in real models, across multiple datasets and encoder architectures. Our main neural collapse result (Theorem 4) is stated in terms of arbitrary admissible regression targets Y and either $S = I$ or $S = L_{\mathcal{T}}$; we additionally confirm that our result predicts the outcomes across multiple choices of these objects.

Choices of Y. Theorem 4 is stated in terms of generic admissible targets Y, and equation 5 describes *decision-faithful* regression targets. We consider two other choices. *One-hot* targets are a centered (hence admissible) version of the standard one-hot regression targets. *Euclidean-faithful* targets arrange class feature means such that the Euclidean distances between pairs are exactly taxonomic distances; these are not necessarily decision-faithful since the classifier's decision rule does not always read distances using the Euclidean metric.

Runs. Table 10 lists nine runs. The first seven use the $S = L_{\mathcal{T}}$ form of the feature regularizer; we vary the dataset, backbone encoder, and regression targets Y. The final two use the $S = I$ regularizer; we realize neural collapse using two datasets here.

Training recipe. To realize neural collapse, we follow the optimization recipe of (Papyan et al., 2020). We use SGD with momentum 0.9, a step schedule that divides the learning rate by a factor of 10 at one-third and two-thirds of training, and no data augmentation. CIFAR-100 models use the standard stem for that dataset, while Aircraft and Cars use the standard ImageNet stem. We were able to induce neural collapse in CIFAR-100 models in only 350 epochs, but found that as many as 2500 epochs were necessary for Aircraft and Cars; the latter two datasets have far fewer training samples than CIFAR-100, so they require more epochs to take more steps. Every model was trained from scratch with a batch size 512, a linear warmup over the first 5–12 epochs, and sometimes gradient-norm clipping at 1.0 to avoid early instability. We disable weight decay across the backbone and apply regularization directly to features, just as is done in our training objective.

K.1 Reproducibility

These experiments require a different experimental methodology than the rest, so their reproducibility section (§B) does not apply here. We are testing whether the predicted geometry is realized in the terminal phase of training (the regime where train error disappears but loss minimization continues), rather than comparing any methods' empirical performance. Thus, we simply tweaked training hyperparameters in an ad-hoc manner for each taxonomy/architecture to eliminate train error partway through training. For these runs, no train augmentation was used. As discussed in §5.2, weight decay was disabled across the encoder backbone. CIFAR-100 has many more train images than Aircraft or Cars, so training CIFAR-100 models takes more steps per epoch than the other two datasets. We used 350 epochs for CIFAR-100 but 2500 for Aircraft and Cars to ensure that they got enough steps to realize collapse fully. Information about each neural collapse run is summarized in Table 11.

Table 11: Configuration of every neural collapse run in §K. Dec., Eucl., and Res. respectively mean "decision-faithful", "Euclidean-faithful", and "resistance", configurations defined in §K. LR is the learning rate per batch size 128.

Dataset	Encoder	Target Y	S	λ_M	λ_W	LR	Warm-up (ep.)	Grad. clip	Epochs
CIFAR-100	ResNet-18	Dec.	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.13	12	1	350
CIFAR-100	ResNet-18	Eucl.	$L_{\mathcal{T}}$	10^{-3}	10^{-4}	0.028	5	–	350
CIFAR-100	ResNet-18	One-hot	$L_{\mathcal{T}}$	10^{-3}	10^{-4}	0.09	5	–	350
CIFAR-100	ResNet-18	Res.	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.14	5	1	350
CIFAR-100	DenseNet-250	Dec.	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.05	12	1	350
FGVC-Aircraft	ResNet-18	Dec.	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.13	12	1	2500
Stanford Cars	ResNet-18	Dec.	$L_{\mathcal{T}}$	10^{-4}	10^{-4}	0.13	12	1	2500
CIFAR-100	ResNet-18	Dec.	I	10^{-4}	10^{-4}	0.10	12	1	350
FGVC-Aircraft	ResNet-18	Dec.	I	10^{-4}	10^{-4}	0.13	12	1	2500

Table 12: Spectral alignment on CIFAR-100. We report mean $\pm$ standard deviation of empirical eigenvalues within both eigenspaces, alongside closed-form predictions from the theory.

Run	ν	σ^2 (emp.)	σ^2 (closed)	θ (emp.)	θ (closed)
ResNet-18, dec. faithful, $S = L_{\mathcal{T}}$	$1/6$	14.90 ± 0.14	14.70	1.555 ± 0.007	1.565
	1	0.990 ± 0.014	1.000	1.001 ± 0.005	1.000
ResNet-18, Eucl. faithful, $S = L_{\mathcal{T}}$	$1/6$	6.10 ± 0.08	6.00	3.135 ± 0.021	3.162
	1	0.988 ± 0.111	1.000	3.105 ± 0.344	3.162
ResNet-18, one-hot, $S = L_{\mathcal{T}}$	$1/6$	0.742 ± 0.013	0.775	1.147 ± 0.009	1.136
	1	0.306 ± 0.004	0.316	1.753 ± 0.006	1.778
ResNet-18, resistance, $S = L_{\mathcal{T}}$	$1/6$	5.85 ± 0.10	6.00	1.015 ± 0.008	1.000
	1	0.986 ± 0.011	1.000	1.004 ± 0.005	1.000
DenseNet-250, dec. faithful, $S = L_{\mathcal{T}}$	$1/6$	14.60 ± 0.67	14.70	1.574 ± 0.036	1.565
	1	1.030 ± 0.128	1.000	0.989 ± 0.070	1.000
ResNet-18, dec. faithful, $S = I$	$1/6$	6.21 ± 0.05	6.00	2.408 ± 0.009	2.449
	1	0.979 ± 0.024	1.000	1.008 ± 0.012	1.000

Unlike the other experiments, these figures do not need a validation split. Thus, they train on the entire native training split without holding out any samples. Each run discussed here was trained on a single NVIDIA L40S GPU.

K.2 Spectral Alignment

We begin by confirming the geometric content of Theorem 5 and Proposition 8. The latter predicts that the class-mean Gram matrix $G = MM^\top$ and the classifier head Gram matrix $WW^\top$ are simultaneously diagonalized by the $L_{\mathcal{T}}$-eigenbasis, with per-mode values of σ_l^2 and θ_l given by equation 18. For each $L_{\mathcal{T}}$-eigenspace, we project the empirical Grams onto E_j and compute eigenvalues; the theory predicts that these are constant and gives an expression for them in closed-form. CIFAR-100 only has two distinct eigenspaces, so we offer a compact summary of the per-eigenspace values in Table 12. The per-mode relative error for every run is given in Table 13. Across all CIFAR-100 models, we find that the empirical σ_l^2 values always land within 1–4% of the predicted eigenvalues, and within-eigenspace spread is modest. The classifier-head values θ_l also track the predicted values closely.

K.3 HNC1: Collapse to Class Means

Theorem 18 predicts that every training sample's feature will sit at its class' mean feature. We measure this with the same NC1 statistic used by Papyan et al. (2020): $\mathrm{tr}(\Sigma_W \Sigma_B^\dagger)/K$, where Σ_W and Σ_B are respectively the within-class and between-class feature covariance matrices over the training set. We also report the trace ratio $\mathrm{tr}\,\Sigma_W / \mathrm{tr}\,\Sigma_B$ and class-distance-normalzied variance $\mathrm{CDNV} = \mathrm{mean}_{u \neq v} (\mathrm{Var}_u + \mathrm{Var}_v)/(2\|m_u - m_v\|^2)$. Table 14 lists final-epoch values and the epoch at which $\mathrm{tr}(\Sigma_W \Sigma_B^\dagger)/K$ first falls below 10^{-2}. Figures 5 and 6 (top rows) chart trajectories over training.

Table 13: Per-mode spectral error over all $K-1$ non-trivial modes: median and 90th percentile of $|\sigma_l^2/\sigma_{l,\text{closed}}^2 - 1|$ and of $|\theta_l/\theta_{l,\text{closed}} - 1|$.

Run	S	Modes	σ^2 err. (med. / p90)	θ err. (med. / p90)
CIFAR-100, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	99	0.011 / 0.024	0.003 / 0.010
CIFAR-100, ResNet-18, Eucl. faithful	$L_{\mathcal{T}}$	99	0.008 / 0.020	0.006 / 0.011
CIFAR-100, ResNet-18, one-hot	$L_{\mathcal{T}}$	99	0.033 / 0.048	0.014 / 0.018
CIFAR-100, ResNet-18, resistance	$L_{\mathcal{T}}$	99	0.014 / 0.033	0.004 / 0.015
CIFAR-100, DenseNet-250, dec. faithful	$L_{\mathcal{T}}$	99	0.062 / 0.184	0.032 / 0.080
FGVC-Aircraft, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	99	0.030 / 0.084	0.014 / 0.038
Stanford Cars, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	195	0.022 / 0.045	0.018 / 0.031
CIFAR-100, ResNet-18, dec. faithful	I	99	0.020 / 0.050	0.007 / 0.021
FGVC-Aircraft, ResNet-18, dec. faithful	I	99	0.027 / 0.086	0.014 / 0.040

Table 14: HNC1 at the final epoch, and the first evaluated epoch (evaluations occur every 10 epochs) at which $\operatorname{tr}(\Sigma_W\Sigma_B^\dagger)/K < 10^{-2}$.

Run	S	$\operatorname{tr}(\Sigma_W\Sigma_B^\dagger)/K$	$\operatorname{tr}\Sigma_W/\operatorname{tr}\Sigma_B$	CDNV	Epoch $< 10^{-2}$
CIFAR-100, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	5.7×10^{-4}	2.1×10^{-4}	1.1×10^{-4}	61
CIFAR-100, ResNet-18, Eucl. faithful	$L_{\mathcal{T}}$	2.4×10^{-4}	1.7×10^{-4}	8.6×10^{-5}	51
CIFAR-100, ResNet-18, one-hot	$L_{\mathcal{T}}$	4.9×10^{-3}	4.9×10^{-3}	2.4×10^{-3}	111
CIFAR-100, ResNet-18, resistance	$L_{\mathcal{T}}$	9.3×10^{-4}	6.1×10^{-4}	3.1×10^{-4}	51
CIFAR-100, DenseNet-250, dec. faithful	$L_{\mathcal{T}}$	5.9×10^{-3}	2.1×10^{-3}	1.1×10^{-3}	221
FGVC-Aircraft, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	6.9×10^{-5}	1.3×10^{-5}	7.6×10^{-6}	407
Stanford Cars, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	1.7×10^{-3}	4.0×10^{-4}	2.4×10^{-4}	402
CIFAR-100, ResNet-18, dec. faithful	I	4.9×10^{-4}	2.8×10^{-4}	1.4×10^{-4}	61
FGVC-Aircraft, ResNet-18, dec. faithful	I	4.8×10^{-5}	2.0×10^{-5}	1.1×10^{-3}	321

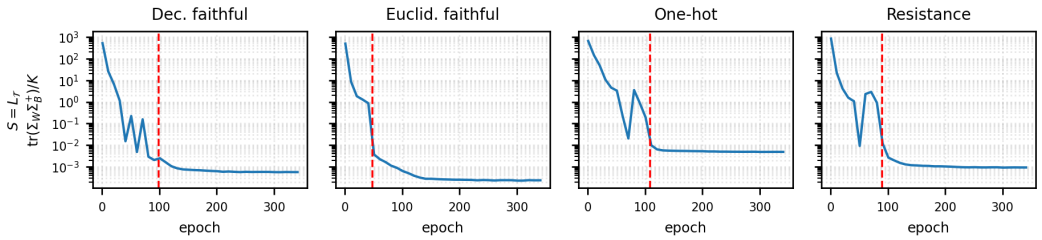
The NC1 statistic falls by three to five orders of magnitude in every run, confirming the theory's prediction. The Aircraft trajectories see the primary HNC1 statistic jump by three orders of magnitude partway through training, before recovering. We note that the other two HNC1 statistics, $\operatorname{tr}\Sigma_W/\operatorname{tr}\Sigma_B$ and the CDNV, do *not* exhibit a similar spike. As such, we suspect that the spike is caused by the pseudo-inverse $\Sigma_B^\dagger$; classes with nearly coincident feature means can give Σ_B eigenvalues close to zero, so corresponding eigenvalues in its psuedo-inverse $\Sigma_B^\dagger$ explode and $\operatorname{tr}(\Sigma_W\Sigma_B^\dagger)/K$ spikes.

K.4 HNC2: Class-Mean Layout

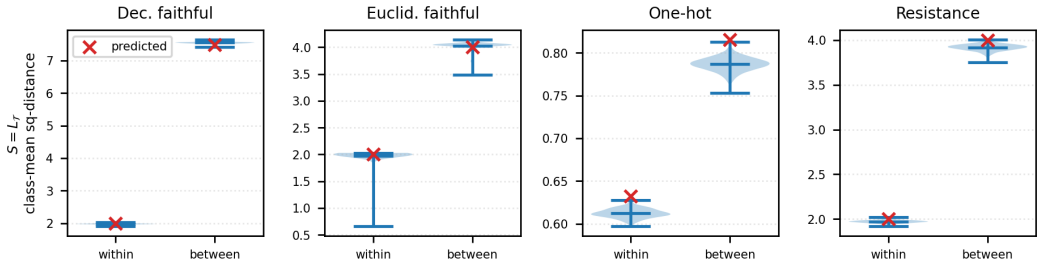
HNC2 (§J.4) predicts the squared Euclidean distances between every pair of class feature means equation 36. We group pairs of classes by their taxonomic distances, and compare the distribution of empirical distances against the predicted values in Table 15. We also report the relative Frobenius distance between empirical and predicted distance matrices. We find that the predicted layout is realized in every run: group means fall within 1–5% of predicted values and the relative Frobenius error of the entire distance matrix is 1–6%. We stress that the theory predicts very different distances for different choices of Y, and *all* of these are realized in practice.

K.5 HNC3: Spectral Self-Duality

HNC3 (§J.5) predicts the head–feature alignment matrix $WM^\top$ in closed-form equation 37. The predicted matrix is symmetric and diagonal in the $L_{\mathcal{T}}$-eigenbasis, with the per-mode eigenvalues c_l given in Theorem 4(iii). We measure the relative Frobenius error between the symmetrized empirical and predicted $WM^\top$ matrices; the empirical matrix's off-diagonal leakage $\|\text{offdiag}(U^\top WM^\top U)\|_F/\|U^\top WM^\top U\|_F$, which is zero if $WM^\top$ is diagonal in the $L_{\mathcal{T}}$-eigenbasis; the antisymmetric part $\|(WM^\top - MW^\top)/2\|_F$ relative to the prediced norm; and the per-eigenspace eigenvalues against c_l. We report all four measurements in Table 16, confirming that the empirical $WM^\top$ matrices are consistently aligning with the predicted matrices.



(a) HNC1: $\mathrm{tr}(\Sigma_W \Sigma_B^\dagger)/K$ over training (log scale). The dashed line marks the first epoch of sustained $\geq 99\%$ training accuracy.



(b) HNC2: final-epoch squared Euclidean distances between class means, split by taxonomic distance (within vs. across superclasses). Red crosses mark the closed-form prediction equation 36.

Figure 5: The four admissible targets on CIFAR-100 with a ResNet-18 and $S = L_\mathcal{T}$.

Table 15: HNC2: squared Euclidean distances between class means at the final epoch, grouped by taxonomic distance $d_\mathcal{T}(u, v)$ (mean $\pm$ standard deviation over class pairs) against the closed-form prediction. The last column is $\|D_{\text{emp}} - D_{\text{pred}}\|_F / \|D_{\text{pred}}\|_F$ over the full matrix of pairwise distances.

Run	S	$d_\mathcal{T}$	Pairs	Empirical	Predicted	Rel. Frob. err.
CIFAR-100, ResNet-18, dec. faithful	$L_\mathcal{T}$	2	200	1.98 ± 0.02	2.00	0.009
		4	4750	7.54 ± 0.03	7.48	
CIFAR-100, ResNet-18, Eucl. faithful	$L_\mathcal{T}$	2	200	1.98 ± 0.15	2.00	0.026
		4	4750	4.02 ± 0.10	4.00	
CIFAR-100, ResNet-18, one-hot	$L_\mathcal{T}$	2	200	0.612 ± 0.005	0.632	0.037
		4	4750	0.787 ± 0.008	0.816	
CIFAR-100, ResNet-18, resistance	$L_\mathcal{T}$	2	200	1.97 ± 0.02	2.00	0.022
		4	4750	3.92 ± 0.04	4.00	
CIFAR-100, DenseNet-250, dec. faithful	$L_\mathcal{T}$	2	200	2.06 ± 0.16	2.00	0.026
		4	4750	7.49 ± 0.19	7.48	
FGVC-Aircraft, ResNet-18, dec. faithful	$L_\mathcal{T}$	2	62	2.00 ± 0.16	2.00	0.022
		4	333	6.72 ± 0.80	6.62	
		6	4555	13.7 ± 2.1	13.5	
Stanford Cars, ResNet-18, dec. faithful	$L_\mathcal{T}$	2	772	2.01 ± 0.15	2.00	0.057
		4	18338	8.77 ± 1.50	8.35	
CIFAR-100, ResNet-18, dec. faithful	I	2	200	1.96 ± 0.04	2.00	0.015
		4	4750	4.05 ± 0.03	4.00	
FGVC-Aircraft, ResNet-18, dec. faithful	I	2	62	1.93 ± 0.25	2.00	0.029
		4	333	3.96 ± 0.11	4.00	
		6	4555	6.04 ± 0.16	6.00	

K.6 HNC4: Nearest-Class-Mean Decision Rule

HNC4 (§J.6) says that the trained head is a nearest-class-mean classifier under the metric A with per-class offsets $\|m_k\|_A^2$ (equation 38). HNC1 tells us that every training sample sits on its class' feature mean, so we evaluate HNC4 over a held-out test set. For each test set, we contain the classifier head's prediction $\arg\max_k \langle w_k, h \rangle + b_k$ against the Euclidean nearest class mean (the classical NC4 decision rule) and the HNC4 rule $\arg\min_k \left(\|h - m_k\|_A^2 - \|m_k\|_A^2 \right)$, with A diagonal

Table 16: HNC3 at the final epoch. Rel. Frob. err. is $\|\mathrm{sym}(WM^\top) - WM_{\text{pred}}^\top\|_F / \|WM_{\text{pred}}^\top\|_F$; leakage and antisymmetry are defined in §K.5. The last column is the largest deviation, over $L_{\mathcal{T}}$-eigenspaces, of the mean empirical eigenvalue of $\mathrm{sym}(WM^\top)$ from the predicted c_l.

Run	S	Rel. Frob. err.	Off-diag. leakage	Antisymmetry	Max eigenspace dev.
CIFAR-100, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	0.002	0.002	0.001	0.4%
CIFAR-100, ResNet-18, Eucl. faithful	$L_{\mathcal{T}}$	0.072	0.070	0.001	1.8%
CIFAR-100, ResNet-18, one-hot	$L_{\mathcal{T}}$	0.028	0.006	0.001	3.0%
CIFAR-100, ResNet-18, resistance	$L_{\mathcal{T}}$	0.005	0.003	0.001	0.4%
CIFAR-100, DenseNet-250, dec. faithful	$L_{\mathcal{T}}$	0.006	0.006	0.003	0.3%
FGVC-Aircraft, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	0.002	0.001	0.001	0.5%
Stanford Cars, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	0.031	0.029	0.002	2.7%
CIFAR-100, ResNet-18, dec. faithful	I	0.002	0.002	0.001	0.3%
FGVC-Aircraft, ResNet-18, dec. faithful	I	0.024	0.022	0.001	3.9%

Table 17: HNC4 on the test set. Agreement (%) between the classifier head and the two nearest-class-mean rules (Euclidean and HNC4) over all samples and samples misclassified by at least one rule.

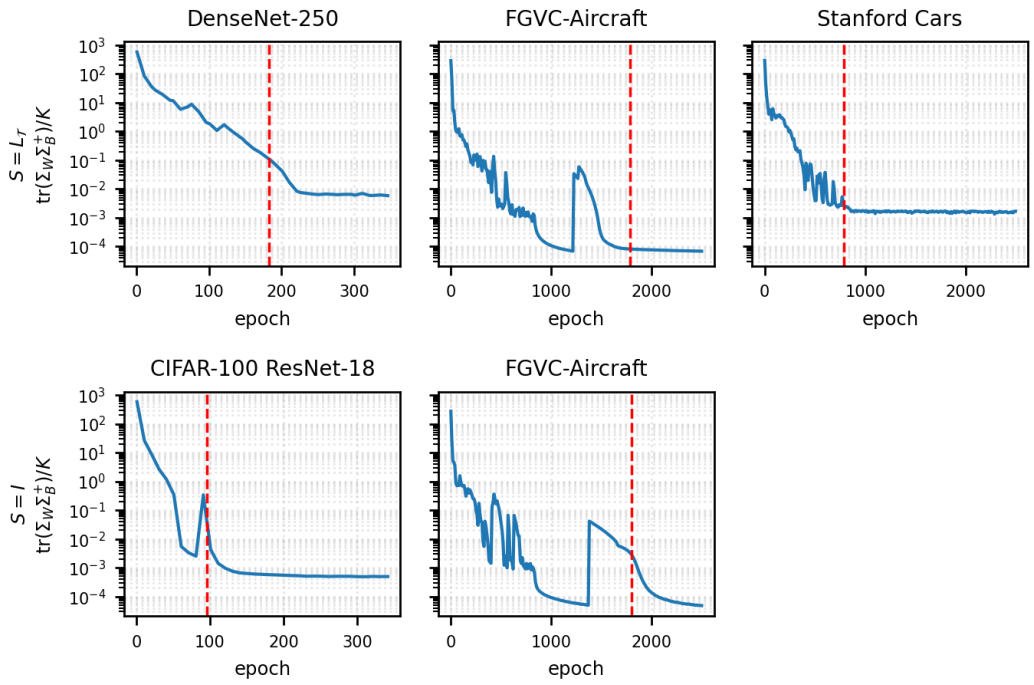
		Head vs. Euclidean		Head vs. A-rule	
Run	S	All	On errors	All	On errors
CIFAR-100, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	97.1	94.0	99.2	98.2
CIFAR-100, ResNet-18, Eucl. faithful	$L_{\mathcal{T}}$	95.7	90.4	99.0	97.8
CIFAR-100, ResNet-18, one-hot	$L_{\mathcal{T}}$	95.5	91.4	98.7	97.6
CIFAR-100, ResNet-18, resistance	$L_{\mathcal{T}}$	95.8	91.4	98.6	97.2
CIFAR-100, DenseNet-250, dec. faithful	$L_{\mathcal{T}}$	97.9	95.1	95.0	88.4
FGVC-Aircraft, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	93.3	89.6	96.4	94.4
Stanford Cars, ResNet-18, dec. faithful	$L_{\mathcal{T}}$	47.1	44.6	95.9	95.7
CIFAR-100, ResNet-18, dec. faithful	I	99.2	98.4	98.5	97.0
FGVC-Aircraft, ResNet-18, dec. faithful	I	92.8	89.0	96.1	94.0

in the $L_{\mathcal{T}}$-eigenbasis with per-mode weights $D_l = \gamma_l/(\sigma_l^2 + \lambda_W)$ computed using the empirical σ_l^2 values (equation 25).

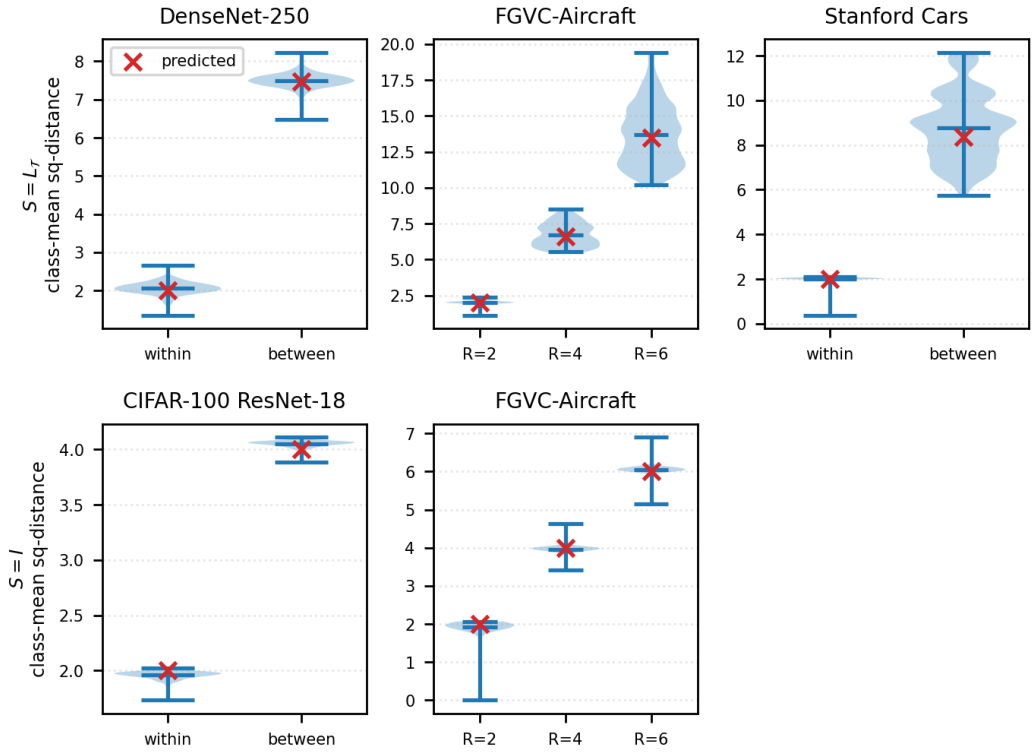
with A built from the $D_l = \gamma_l/(\sigma_l^2 + \lambda_W)$ on the empirical mode directions (equation 25).

HNC4 (§J.6) says the trained head is a nearest-class-mean classifier under the metric A with per-class offsets $\|m_k\|_A^2$ (equation 38). The four training-set statistics above cannot test this, since every training feature sits on its class mean; we therefore evaluate it on the held-out test set. For each test feature h we compare the head's prediction $\arg\max_k \langle w_k, h\rangle + b_k$ against (a) the Euclidean nearest class mean, $\arg\min_k \|h - m_k\|^2$, i.e. classical NC4, and (b) the HNC4 rule $\arg\min_k \big(\|h - m_k\|_A^2 - \|m_k\|_A^2\big)$, with A built from the closed-form mode weights $D_l = \gamma_l/(\sigma_l^2 + \lambda_W)$ on the empirical mode directions (equation 25). Table 17 reports test accuracy under each rule and the fraction of test samples on which the head agrees with each rule, overall and restricted to samples that at least one rule misclassifies (agreement on correctly classified samples is trivially high).

The comparison between HNC4 and the Euclidean rule (i.e., classical NC4) varies by datasets. The CIFAR-100 taxonomy is leaf-transitive; on this dataset, the HNC4 and the Euclidean rule align about equally well with the classifier head. On the more irregular Aircraft and Cars taxonomies, though, we observe that the HNC4 rule agrees with the classifier head much more often than the Euclidean rule.



(a) HNC1 over training.



(b) HNC2 at the final epoch. Aircraft's three-level taxonomy gives three taxonomic distances ($R \in \{2, 4, 6\}$).

Figure 6: Decision-faithful targets across encoders, datasets, and regularizers. Top row of each panel: $S = L_{\mathcal{T}}$ (DenseNet-250 on CIFAR-100; ResNet-18 on FGVC-Aircraft and Stanford Cars); bottom row: $S = I$ (ResNet-18 on CIFAR-100 and FGVC-Aircraft). Conventions as in Figure 5.