ONE SIZE DOESN'T FIT ALL: DEPTH-AWARE MANIFOLD-GATED MIXTURE-OF-EXPERTS FOR DEEP IMBALANCED REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing representation-based Deep Imbalanced Regression (DIR) methods typically enforce label–feature geometry at a fixed representation depth. This one-size-fits-all strategy can overlook depth-dependent variation in label–feature alignment. We propose Depth-aware Manifold-Gated Mixture-of-Experts (DMG-MoE) to address this depth-dependent variation. It measures Depth-wise Manifold Alignment (DMA) across selected backbone depths by comparing an input's soft feature neighborhood to its soft label neighborhood via KL divergence, and uses this signal to guide the allocation of prediction weight across depth-wise experts. We distill geometry-aware supervision from a training-time DMA teacher into a label-free student gate that produces routing weights without label access at inference. We parameterize routing via a stick-breaking scheme that sequentially allocates prediction mass across depth-wise experts. This enables sample-specific, geometry-guided depth adaptation. On AgeDB-DIR and IMDB-WIKI-DIR, DMG-MoE achieves the lowest overall MAE and GM among the compared methods. We also observe gains on the tabular benchmarks (Abalone, Airfoil) and with two additional backbones (ResNet-152, ConvNeXt-Tiny). The learned routing aligns with its DMA teacher signal, supporting the geometry-aware mechanism.

1 INTRODUCTION

Real-world regression tasks typically exhibit severely imbalanced continuous label distributions, where most training samples lie in a narrow head region and only a few fall in the tail (Zhang et al., 2023). This pattern commonly arises in visual regression (e.g., age estimation) (Zhang et al., 2024) and tabular regression tasks (Camacho & Bacao, 2024). When trained with standard objectives, deep regressors become biased toward frequent targets, producing disproportionately large errors on rare values.

Data-level remedies such as resampling and reweighting can mitigate imbalance (Avelino et al., 2024; Chen et al., 2024). However, they leave a core challenge unaddressed: learning a representation geometry that remains reliable across the entire label range (Dong et al., 2025; Keramati et al., 2023). Recent work on Deep Imbalanced Regression (DIR) has thus focused on representation-level solutions, guided by the prior that nearby labels should correspond to nearby representations. For instance, FDS (Yang et al., 2021) pioneered representation-level regularization for DIR via feature distribution smoothing, and RankSim (Gong et al., 2022) enforces label–feature consistency through ranking-based neighborhood regularization.

While these methods improve representation smoothness under imbalance, they share a common limitation: they apply constraints at a single depth, assuming a one-size-fits-all representation granularity. This is overly restrictive: local manifold reliability is depth-dependent and varies across samples. This motivates using depth-wise label–feature alignment to guide sample-specific routing.

We propose **Depth-aware Manifold-Gated Mixture-of-Experts (DMG-MoE)**, an architecture that attaches lightweight depth-wise experts to multiple backbone depths and routes each input to more reliable depths. Our key signal is **Depth-wise Manifold Alignment (DMA)**: we construct soft neighborhood distributions in label and feature spaces, and measure their discrepancy via KL diver-

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.

gence. This gives a smooth, distance-sensitive estimate of local manifold reliability at each depth. During training, a label-based DMA teacher supervises routing and is distilled into a *label-free* student gate for inference. We adopt a stick-breaking router that allocates prediction mass sequentially from shallow to deep, yielding an interpretable depth-wise mixture. Our design is motivated by a new proposed DIR principle: *each sample should be supervised at the depth where its local feature–label geometry is most reliable.*

Our contributions are summarized as follows:

- We identify a depth-dependent manifold reliability phenomenon in DIR: local feature–label alignment varies across depths and across samples. To quantify this agreement, we propose DMA, a soft, distance-sensitive metric that reflects local manifold reliability at each depth.
- We leverage this insight through a depth-wise MoE framework with stick-breaking routing, which allocates prediction mass sequentially from shallow to deep, yielding an interpretable depth-wise mixture.
- To make DMA usable at inference without labels, we distill the teacher signal into a label-free student gate, retaining per-sample depth reliability information while discarding the label dependency.
- We evaluate DMG-MoE on standard DIR benchmarks (AgeDB-DIR, IMDB-WIKI-DIR) with ResNet-50 backbone, then extend to tabular domains (Abalone, Airfoil) and diverse backbones (ResNet-152, ConvNeXt-Tiny) to demonstrate its generality. Results show the lowest overall MAE on all four evaluated datasets among the compared methods, and diagnostic analyses show that learned routing aligns with its DMA teacher signal.

2 Related Work

Distribution- and loss-level strategies for imbalanced regression. Imbalanced regression has been extensively studied in tabular domains. Data-level approaches such as SMOGN (Branco et al., 2017) and WSMOTER (Camacho & Bacao, 2024) generate synthetic samples to rebalance the label distribution, while objective-level strategies such as density-based weighting (Steininger et al., 2021) adjust per-sample loss contributions. Yang et al. (2021) formalize Deep Imbalanced Regression (DIR) in deep vision, propose feature distribution smoothing (FDS) for representation regularization, and establish the standard benchmarks (AgeDB-DIR, IMDB-WIKI-DIR). Balanced MSE (Ren et al., 2022) reweights samples by label density, and uncertainty-aware formulations (Wang & Wang, 2023) adaptively temper the loss. These methods seek to compensate for the effects of imbalance during training. Beyond this, we additionally allocate, for each sample, prediction weight across depths according to local feature–label geometry.

Label-aligned representation learning. A recurring theme in DIR is to learn representations whose geometry reflects the ordinal and neighborhood structure of continuous labels. RankSim (Gong et al., 2022) pioneers ranking-based regularization for label–feature neighborhood consistency. RNC (Zha et al., 2023) and ConR (Keramati et al., 2023) recast this as a contrastive objective: representations of nearby labels are pulled together, while those of distant labels are pushed apart. PRIME (Lim et al., 2025) shifts from ranking to distribution matching, using KL divergence between label-induced and feature-induced neighborhoods via learnable proxies. SRL (Dong et al., 2025) provides a spectral relaxation that unifies ranking, smoothing, and neighborhood constraints. These methods all enforce label-aligned geometry, but share a common limitation in their reported formulations: they operate at a single depth, typically the final representation of the backbone. We instead measure label-aligned geometry at multiple depths. The resulting per-depth alignment scores guide, for each sample, which depth is most reliable for prediction.

Expert-based decompositions for DIR. Beyond representation-level regularization, recent DIR methods explore divide-and-conquer predictors that split the label space to handle heterogeneous difficulty. HCA (Xiong & Yao, 2024) proposes hierarchical classification adjustment to handle heterogeneous label granularity. Group-DIR (Pu et al., 2025) introduces group-based decompositions with descending soft labeling. While effective, these approaches rely on predefined label partitions. We instead replace predefined partitions with geometrically-adaptive allocation: prediction responsibility is determined by per-depth manifold reliability rather than fixed label-space boundaries.

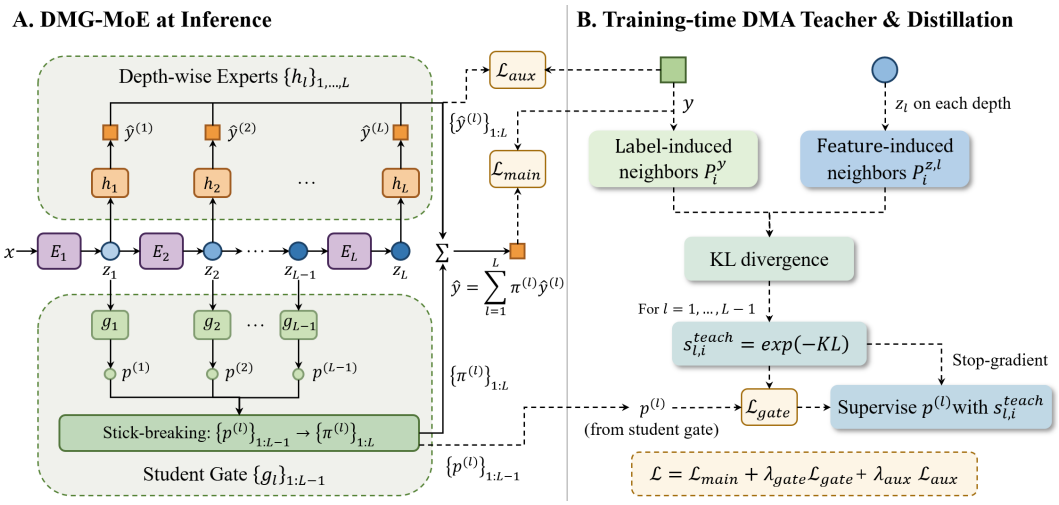


Figure 1: Overview of DMG-MoE. (A) Inference: the backbone produces depth-wise representations; depth-wise experts predict at each depth, and a label-free student gate outputs stick-breaking parameters to form depth-wise mixture weights for the final prediction. (B) Training: a DMA teacher uses labels to compute per-depth manifold alignment scores, which are distilled into the student gate. The teacher is discarded at inference.

Depth-adaptive Mixture-of-Experts. Modern Mixture-of-Experts (MoE) (Masoudnia & Ebrahimpour, 2014) enables conditional computation through sparse gating (Shazeer et al., 2017; Fedus et al., 2022). Its routing strategies are designed to balance stability and expert utilization (Zhou et al., 2022; Lee et al., 2025). Dynamic inference and early-exit networks also study depth-adaptive computation, primarily for efficiency (Huang et al., 2024; Wang et al., 2018; Royer et al., 2023). Our depth-wise MoE shares the expert-routing paradigm with these methods, but guides its learned gate with DMA-derived teacher supervision alongside task-loss gradients. This yields an interpretable parameterization in which mixture weights are informed by local geometry and shaped by the regression objective.

3 Method

3.1 Problem Setup and Overview

Deep Imbalanced Regression. We study Deep Imbalanced Regression (DIR) with a long-tailed continuous label distribution. Given $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ with $x_i \in \mathcal{X}$ and $y_i \in \mathbb{R}$, most samples lie in the head region while tail targets are sparse. We aim for accurate prediction across the entire label range, covering both dense head and sparse tail regions. Figure 1 outlines the DMG-MoE framework, and a complete notation table is provided in the Appendix.

Backbone and depth-wise representations. A backbone F with L sequential modules $E_{1:L}$ produces depth-wise representations:

$$z_1 = E_1(x), \quad z_l = E_l(z_{l-1}),\ l = 2, \dots, L. \tag{1}$$

Standard practice predicts from the deepest representation as $\hat{y} = h_L(z_L)$. However, local manifold reliability varies across depths and samples: many-shot samples can already show substantial alignment at intermediate depths, while few-shot samples often show larger alignment gains at later depths. This motivates DMG-MoE to allocate prediction weight across depths according to each sample's local geometry.

Depth-wise experts and mixture prediction. At each intermediate depth, a lightweight expert h_l predicts $\hat{y}^{(l)} = h_l(z_l)$ from the backbone's representation z_l. Instead of predicting solely from the

deepest layer, DMG-MoE forms its prediction as a depth-wise mixture across all layers:

$$\hat{y} = \sum_{l=1}^{L} \pi^{(l)} \, \hat{y}^{(l)}, \qquad \sum_{l=1}^{L} \pi^{(l)} = 1, \; \pi^{(l)} \geq 0, \tag{2}$$

where the mixture weights $\pi^{(l)}$ are produced by the routing mechanism described next.

Depth-adaptive routing. Depth-adaptive routing proceeds in two stages. During training, a Depth-wise Manifold Alignment (DMA) teacher quantifies per-depth manifold reliability by comparing label-induced and feature-induced neighborhood distributions (Section 3.2). This signal is distilled into a label-free gate, whose outputs are then converted to depth-wise mixture weights via stick-breaking (Section 3.3).

3.2 Depth-wise Manifold Alignment

Our depth-adaptive routing is guided by a training-time alignment signal that measures, at each depth, how well a sample's feature neighborhood matches its label neighborhood. For each mini-batch, we construct both neighborhood distributions. Their alignment is then quantified via the Kullback–Leibler (KL) divergence as detailed below.

Label- and feature-induced neighborhood distributions. For each mini-batch $\{(x_i, y_i)\}_{i=1}^{B}$, i indexes an anchor sample, $j \neq i$ indexes another sample in the batch, and l indexes a selected backbone depth. We define a generic neighbor distribution from logits a_{ij} via a row-wise softmax with $a_{ii} = -\infty$:

$$P_i(j) = \frac{\exp(a_{ij})}{\sum_{k \neq i} \exp(a_{ik})}, \qquad j \neq i. \tag{3}$$

Here $P_i(j)$ denotes a generic neighbor probability for anchor i and neighbor j, and a_{ij} is the corresponding softmax logit. In label space, the corresponding quantities are $P_i^y(j)$ and a_{ij}^y. In feature space at backbone depth l, they are $P_i^{z,l}(j)$ and $a_{ij}^{z,l}$.

Label-induced. The label-induced logits are defined via absolute label differences with temperature $\tau_y > 0$:

$$a_{ij}^y = -\frac{|y_i - y_j|}{\tau_y}, \quad i \neq j, \tag{4}$$

yielding the label neighbor distribution P_i^y via Eq. 3.

Feature-induced at depth l. We ℓ_2-normalize features $\tilde{z}_{l,i} = z_{l,i}/\|z_{l,i}\|_2$ and define feature-induced logits via cosine similarity with temperature $\tau_z > 0$:

$$a_{ij}^{z,l} = \frac{\tilde{z}_{l,i}^\top \tilde{z}_{l,j}}{\tau_z}, \quad i \neq j, \tag{5}$$

yielding the feature neighbor distribution $P_i^{z,l}$ via Eq. 3.

Depth-wise manifold alignment score. We measure mismatch by KL divergence:

$$\begin{aligned} m_{l,i} &= \mathrm{KL}\Big(P_i^y \,\|\, P_i^{z,l}\Big) \\ &= \sum_{j \neq i} P_i^y(j) \Big(\log P_i^y(j) - \log P_i^{z,l}(j)\Big), \end{aligned} \tag{6}$$

where $\log P_i^y$ and $\log P_i^{z,l}$ are computed via log-softmax for numerical stability. We then convert $m_{l,i}$ into a bounded DMA teacher score

$$s_{l,i}^{\mathrm{teach}} = \exp(-m_{l,i}) \in (0, 1], \tag{7}$$

where larger values indicate higher local manifold reliability at depth l for sample i.

3.3 DEPTH-WISE MIXTURE WEIGHTS

The DMA teacher score $s_{l,i}^{\text{teach}}$ relies on ground-truth labels and is unavailable at inference. We therefore learn a *label-free* gate that predicts routing from intermediate features by distilling the teacher signal during training.

Gate architecture. The gate consists of $L-1$ lightweight gating networks $\{g_l\}_{l=1}^{L-1}$ that output the per-depth allocation probabilities:

$$p_i^{(l)} = \sigma\big(g_l(z_{l,i})\big) \ \in \ (0,1), \qquad l = 1, \ldots, L-1, \tag{8}$$

where σ denotes the sigmoid function. We train $p_i^{(l)}$ to match the teacher score $s_{l,i}^{\text{teach}}$, so that the gate learns to approximate the teacher's per-depth allocation without label access.

Distillation objective. We regress the teacher scores with an ℓ_2 distillation loss:

$$\mathcal{L}_{\text{gate}} = \frac{1}{B} \sum_{i=1}^{B} \sum_{l=1}^{L-1} \Big\| p_i^{(l)} - \text{stopgrad}\big(s_{l,i}^{\text{teach}}\big) \Big\|_2^2. \tag{9}$$

In implementation, we clamp teacher targets to $[\epsilon, 1-\epsilon]$ with a small ϵ for numerical stability. The stopgrad operator detaches the teacher from the distillation gradient, ensuring that DMA scores reflect genuine label–feature alignment rather than being pulled toward the gate outputs.

The depth-wise mixture weights $\{\pi_i^{(l)}\}_{l=1}^{L}$ are then obtained via stick-breaking of the per-depth gate outputs:

$$\begin{aligned} \pi_i^{(1)} &= p_i^{(1)}, \\ \pi_i^{(l)} &= p_i^{(l)} \prod_{k=1}^{l-1} \big(1 - p_i^{(k)}\big), \qquad l = 2, \ldots, L-1, \\ \pi_i^{(L)} &= \prod_{k=1}^{L-1} \big(1 - p_i^{(k)}\big). \end{aligned} \tag{10}$$

This enforces a sequential allocation from shallow to deep. Substituting these weights into Eq. 2 yields DMG-MoE's final prediction.

3.4 TRAINING OBJECTIVES

We train DMG-MoE end-to-end with a regression loss on the depth-wise mixture prediction, together with two auxiliary objectives: (i) distillation for the label-free gate, and (ii) encouraging specialization among depth-wise experts. We describe each term below and then present the full objective.

Main regression loss. For a mini-batch, DMG-MoE predicts $\hat{y}_i = \sum_{l=1}^{L} \pi_i^{(l)} \hat{y}_i^{(l)}$ (Eq. 2) and optimizes a base regression loss $\ell(\hat{y}_i, y_i)$:

$$\mathcal{L}_{\text{main}} = \frac{1}{B} \sum_{i=1}^{B} \ell(\hat{y}_i, y_i). \tag{11}$$

Route-weighted auxiliary loss for expert specialization. A naïve auxiliary loss that trains every expert on every sample tends to make depth-wise experts converge to similar mappings. To encourage specialization, we train each expert primarily on the samples it is responsible for under the current routing by using a route-weighted auxiliary loss:

$$\mathcal{L}_{\text{aux}} = \frac{1}{B} \sum_{i=1}^{B} \sum_{l=1}^{L} \text{stopgrad}\big(\pi_i^{(l)}\big)\, \ell\big(\hat{y}_i^{(l)}, y_i\big), \tag{12}$$

where $\text{stopgrad}(\pi_i^{(l)})$ blocks gradients from $\mathcal{L}_{\text{aux}}$ to the gate, so that routing is learned via $\mathcal{L}_{\text{main}}$ and $\mathcal{L}_{\text{gate}}$ while experts adapt to the current assignments.

Full objective. Assembling all these loss terms, the overall objective is:

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \lambda_{gate}\mathcal{L}_{\text{gate}} + \lambda_{aux}\mathcal{L}_{\text{aux}}, \tag{13}$$

where λ_{gate} and λ_{aux} control the strength of auxiliary terms. The complete training procedure is provided in the Appendix.

3.5 Theoretical Insights

We provide a theoretical perspective supporting two design choices: (i) using KL-based soft neighborhood alignment as a depth-wise reliability signal, and (ii) distilling this signal into a label-free gate for inference.

DMA controls local neighborhood distortion. Recall the label and feature neighbor distributions P_i^y and $P_i^{z,l}$, and define the KL mismatch $m_{l,i} = \mathrm{KL}(P_i^y \| P_i^{z,l})$. Assume labels are bounded $y \in [y_{\min}, y_{\max}]$, and define the normalized label deviation function $f_i(j) = |y_j - y_i|/(y_{\max} - y_{\min}) \in [0, 1]$. Then the discrepancy between expected deviations under the two neighborhoods is bounded by

$$\begin{aligned} \left| \mathbb{E}_{j\sim P_i^{z,l}}[f_i(j)] - \mathbb{E}_{j\sim P_i^{y}}[f_i(j)] \right| &\le 2\,\mathrm{TV}(P_i^y, P_i^{z,l}) \\ &\le \sqrt{2\,\mathrm{KL}(P_i^y \| P_i^{z,l})} \\ &= \sqrt{2\,m_{l,i}}, \end{aligned} \tag{14}$$

where $\mathrm{TV}(P, Q) = \frac{1}{2}\|P - Q\|_1$ and the second inequality follows from Pinsker's inequality. Eq. (14) implies that a small KL mismatch guarantees small distortion in local label dispersion induced by feature neighbors, motivating $s_{l,i}^{\text{teach}} = \exp(-m_{l,i})$ as a bounded depth-wise reliability score.

Distillation yields bounded routing deviation. Let $p_i = \{p_i^{(l)}\}_{l=1}^{L-1}$ be label-free gate outputs and let $s_i = \{s_{l,i}^{\text{teach}}\}_{l=1}^{L-1}$ be the corresponding teacher targets. Define gate and teacher depth-wise mixture weights by stick-breaking as $\pi_i = \mathrm{SB}(p_i)$ and $\pi_i^{\text{teach}} = \mathrm{SB}(s_i)$.

Assume expert predictions are bounded, $|\hat{y}_i^{(l)}| \le M$. The deviation between gate and teacher mixture predictions satisfies

$$\begin{aligned} \left| \hat{y}_i - \hat{y}_i^{\text{teach}} \right| &= \left| \sum_{l=1}^{L} \left(\pi_i^{(l)} - \pi_i^{\text{teach},(l)}\right) \hat{y}_i^{(l)} \right| \\ &\le M\, \|\pi_i - \pi_i^{\text{teach}}\|_1. \end{aligned} \tag{15}$$

Appendix A.2 relates this bound to the gate regression error through the stick-breaking map.

3.6 Complexity Analysis

Let B be the mini-batch size and C_l the feature dimension at tapped depth l. The DMA teacher constructs pairwise feature similarities in $\mathcal{O}(B^2 C_l)$ time per depth, followed by row-wise softmax and KL divergence in $\mathcal{O}(B^2)$. Across the tapped depths, its dominant training-time cost is $\mathcal{O}\big(B^2 \sum_{l\in\mathcal{L}_t} C_l\big)$. At inference, the teacher is discarded, leaving only the lightweight heads, gates, and stick-breaking operation in addition to the shared backbone. The parameter count and a detailed complexity comparison with RankSim are given in Appendix A.3.

4 Experiments

4.1 Setup

Benchmarks. We evaluate DMG-MoE on two standard DIR benchmarks for age estimation, AgeDB-DIR and IMDB-WIKI-DIR (Yang et al., 2021). We use the official train/val/test splits and the evaluation protocol provided by the DIR benchmark. To test generalization beyond visual regression, we also evaluate on two UCI-DIR benchmarks (Abalone and Airfoil) under the SRL protocol (Dong et al., 2025).

Table 1: MAE results on four DIR benchmarks.

Method	AgeDB-DIR				IMDB-WIKI-DIR				Airfoil				Abalone			
	All	Many	Med	Few	All	Many	Med	Few	All	Many	Med	Few	All	Many	Med	Few
Vanilla	7.67	6.66	9.30	12.61	8.03	7.16	15.48	26.11	5.66	5.11	5.03	6.75	4.57	**0.88**	2.65	7.97
LDS + FDS	7.55	7.03	8.46	10.52	7.73	7.22	12.98	23.71	5.76	4.45	4.79	7.79	5.09	0.90	3.26	9.26
RankSim	7.41	6.49	8.73	12.47	7.72	**6.92**	14.52	25.89	5.23	5.05	4.91	5.72	4.33	0.98	2.59	7.42
Bal. MSE	7.98	7.58	8.65	9.93	8.43	7.84	13.35	23.27	5.69	4.51	5.04	7.28	5.37	2.14	2.66	9.37
Ord. Ent.	7.60	6.69	8.87	12.68	8.01	7.17	15.15	26.48	6.27	4.85	5.37	8.32	6.77	2.31	4.01	11.61
ConR	7.41	6.51	8.81	12.04	7.84	7.15	14.36	25.15	–	–	–	–	–	–	–	–
SRL	7.22	6.64	8.28	9.81	7.69	7.08	**12.65**	22.78	5.10	4.83	4.75	5.69	4.16	0.89	2.42	7.19
DMG-MoE	**6.97**	**6.37**	**8.05**	**9.63**	**7.57**	6.93	12.90	**22.24**	**3.47**	**3.10**	**3.01**	**4.46**	**3.97**	2.11	**1.88**	**4.91**

*Best in **bold**, second best underlined.*

Baselines. We compare against representative DIR methods: LDS/FDS (Yang et al., 2021), RankSim (Gong et al., 2022), Balanced MSE (Ren et al., 2022), ConR (Keramati et al., 2023), and SRL (Dong et al., 2025). We follow the training and evaluation protocols of Dong et al. (2025) and adopt their reproduced baseline numbers for fair comparison. SRL omits ConR on UCI-DIR because its data augmentation is not fully established for tabular data (Dong et al., 2025).

Backbone and depth-wise experts. Following established DIR practice, we adopt ResNet-50 (He et al., 2016) as the backbone, denoted Vanilla, and tap $L = 4$ stages (conv2_x–conv5_x) as the depth-wise representations. A lightweight regression head is attached at each depth, with routing via $L-1$ stick-breaking gates. For tabular experiments, we use a 4-layer MLP backbone as in Dong et al. (2025).

Metrics. We report Mean Absolute Error (MAE) and Geometric Mean (GM) following prior DIR work (Yang et al., 2021). We report All and shot-group results from unit-width training bins. The Few/Many thresholds are $< 20/> 100$ for AgeDB-DIR and IMDB-WIKI-DIR, $< 10/> 40$ for Airfoil, and $< 100/> 400$ for Abalone, with Med. covering the remainder (Dong et al., 2025).

Training protocol. All models are trained for 90 epochs with the Adam optimizer, with the learning rate decayed by 0.1 at epochs 60 and 80. We follow dataset-specific batch sizes, learning rates, and preprocessing from Dong et al. (2025). Checkpoints are selected by validation MAE. Full implementation details are provided in the Appendix.

4.2 Main Results

Vision DIR benchmarks. Table 1 reports MAE on all four benchmarks. On the two vision benchmarks, DMG-MoE achieves the lowest overall and Few-shot MAE and GM (Appendix Table A.5). Against SRL, overall MAE falls from 7.22 to 6.97 on AgeDB-DIR and from 7.69 to 7.57 on IMDB-WIKI-DIR. Few-shot MAE improves from 9.81 to 9.63 and from 22.78 to 22.24, respectively. The corresponding Few-shot GM decreases from 6.29 to 5.69 and from 15.20 to 13.97.

Tabular DIR benchmarks. On the two tabular UCI-DIR benchmarks (Abalone, Airfoil), Table 1 reports mean MAE over 3 runs under the SRL protocol. DMG-MoE achieves the lowest overall and Few-shot MAE on both datasets (e.g., Airfoil Few-shot MAE 4.46 vs. 5.69). Since tabular inputs lack spatial structure, these gains support that the method relies on depth-wise label–feature geometry rather than an image-specific inductive bias.

Backbone generality. To test whether the routing benefit transfers across architectures, we apply DMG-MoE to ResNet-152 (same family, larger scale) and ConvNeXt-Tiny (different architecture) on AgeDB-DIR. Appendix Table A.6 shows that DMG-MoE improves overall MAE and GM on all three backbones. Overall MAE decreases from 7.52 to 7.16 on ResNet-152 and from 8.35 to 7.70 on ConvNeXt-Tiny, with the largest gains in both metrics on Few-shot samples. These results extend the observed gains to a larger ResNet and a different architecture.

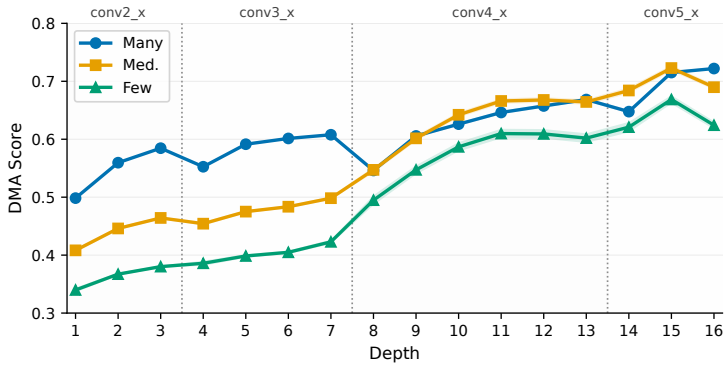


Figure 2: Depth-dependent manifold alignment in a vanilla ResNet-50 on AgeDB-DIR. DMA scores are averaged over Many/Med./Few subsets; dotted lines mark stage transitions.

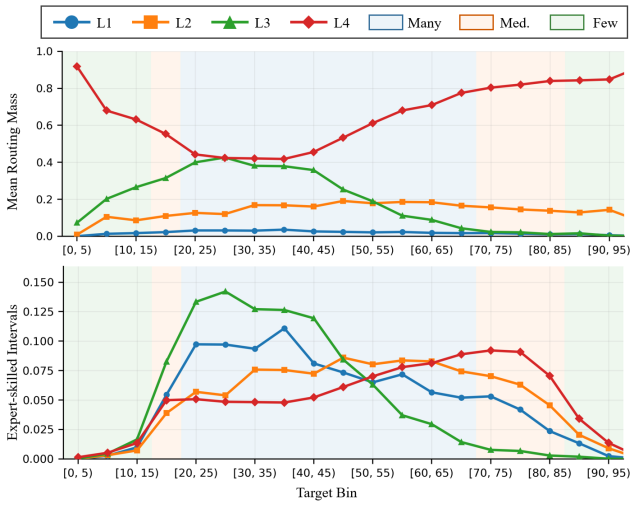


Figure 3: Per-bin routing structure on AgeDB-DIR. *Top:* conditional routing distribution over age bins. *Bottom:* per-expert normalized traffic. Shaded regions mark Many/Med./Few-shot intervals.

4.3 Mechanism Analysis

We provide mechanism evidence connecting (i) depth-dependent local manifold reliability, (ii) bin-dependent depth-adaptive routing, and (iii) the fidelity of routing w.r.t. the DMA teacher signal.

Depth-dependent manifold reliability. We diagnose how well local feature neighborhoods respect label geometry across representation depth using the DMA score defined in Section 3.2, here evaluated at all 16 bottleneck blocks for diagnosis, rather than only the $L{=}4$ routed stages. As shown in Figure 2, alignment generally improves with depth, with larger gains for few-shot samples, supporting the premise that tail regions benefit more from deeper representations. The corresponding RankSim and DMG-MoE curves are provided in the Appendix.

Per-bin routing structure. The learned routing exhibits a structured depth-wise allocation (Figure 3, top): dense regions (Many) concentrate mass at intermediate experts (L2/L3), while tail bins (Few) at both age extremes shift substantially toward the deepest expert (L4). The shallowest expert (L1) is rarely engaged. The per-expert normalized view (bottom) shows clear depth-wise specialization across the label range. Appendix Figure A.2 reports depth-wise DMA by shot group, while Figure A.6 shows per-bin expert errors alongside routing mass. For Many-shot samples, intermediate depths can already show substantial alignment. Task-loss gradients also shape the gate, so routing need not track the depth with the highest DMA.

Table 2: MAE results of key ablations on AgeDB-DIR.

Variant	All	Many	Med	Few
Multi-head + uniform mix (A1)	7.51	6.90	8.49	10.64
Learned gate, no teacher (A2)	7.16	6.52	8.17	10.32
Always deepest (A3)	7.20	6.62	8.16	10.03
Softmax routing (A4)	7.45	6.68	8.93	10.70
Gate detached from $\mathcal{L}_{\text{main}}$ (A5)	7.43	6.71	8.58	11.06
RankSim teacher (A6)	7.26	6.63	8.16	10.67
Marginal-improvement teacher (A7)	7.16	6.48	8.37	10.19
Shuffled teacher (A8)	7.37	6.66	8.56	10.01
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

*Best in **bold**.*

Per-sample visualization. Appendix Figure A.7 visualizes DMA, expert error, and routing weights across the four depths for several AgeDB-DIR validation samples. In these examples, the depth with the highest DMA also has the lowest expert error and receives the largest routing weight.

Teacher–student agreement and routing sharpness. We measure how closely the learned routing follows the DMA teacher signal using Spearman agreement between π and π^{teach}. The overall agreement is $\rho \approx 0.611$ and remains positive on Few-shot samples ($\rho \approx 0.468$), while tail routing is substantially sharper, consistent with allocating deeper capacity to sparse label regions. Full diagnostics, along with further analyses of temperature sensitivity and expert specialization, are reported in the Appendix.

Together, the depth-wise and per-bin analyses show structured variation across label regions, while the per-sample visualization illustrates how DMA, expert error, and routing can coincide at a given depth. Teacher–student agreement measures fidelity to the DMA training signal.

4.4 Ablation Study

AgeDB-DIR ablations examine capacity, DMA supervision, and routing design. All variants use the same backbone and $L{=}4$ depth-wise heads unless stated otherwise, with MAE reported for All/Many/Med./Few.

Table 2 shows that the gains are not from capacity alone. Uniform mixing (A1) barely improves Few-shot MAE, while routing without the DMA teacher (A2) improves on A1 but trails DMG-MoE. Always using the deepest representation (A3) degrades results, confirming the role of depth adaptation. Softmax routing (A4) underperforms stick-breaking, and detaching the gate from $\mathcal{L}_{\text{main}}$ (A5) confirms that DMA distillation and task gradients are complementary. Teacher variants (A6/A7) outperform the shuffled-teacher control (A8), confirming sample-specific teacher structure beyond batch-level statistics. Further head ablations appear in the Appendix.

On AgeDB-DIR, DMG-MoE adds 13.7% per-epoch time over Vanilla and runs faster per epoch than RankSim despite the DMA teacher's $\mathcal{O}(B^2)$ computation (Appendix A.9.2). Appendix A.10 reports sensitivity to loss weights, temperatures, and batch size.

5 Conclusion

DMG-MoE replaces single-depth DIR regularization with geometry-aware routing across depth-wise experts. A KL-based DMA teacher supervises a label-free stick-breaking gate that requires no labels at inference. On the evaluated datasets and backbones, DMG-MoE improves overall MAE, while routing diagnostics support the mechanism.

Limitations. DMA relies on informative local feature–label neighborhoods. Its behavior when this locality is weak remains to be evaluated.

Future work. We plan to extend manifold-guided routing to multi-dimensional targets and explore scalable neighborhood estimators for large batches or high-resolution representations.

Reproducibility Statement

An anonymous repository for this work is available at https://anonymous.4open.science/r/dmgmoe/. Implementation details and additional experiments are provided in the Appendix.

AI Use Statement

Generative AI tools were used solely to improve the readability of this manuscript. The authors reviewed all AI-assisted edits and take responsibility for the final content.

Ethical Statement

This paper advances the field of Machine Learning. We see no specific societal consequences that must be highlighted here.

References

Juscimara G Avelino, George DC Cavalcanti, and Rafael MO Cruz. Resampling strategies for imbalanced regression: a survey and empirical analysis. *Artificial Intelligence Review*, 57(4):82, 2024.

Paula Branco, Luís Torgo, and Rita P Ribeiro. Smogn: a pre-processing approach for imbalanced regression. In *First international workshop on learning with imbalanced domains: Theory and applications*, pp. 36–50. PMLR, 2017.

Luís Camacho and Fernando Bacao. Wsmoter: a novel approach for imbalanced regression. *Applied intelligence*, 54(19):8789–8799, 2024.

Wuxing Chen, Kaixiang Yang, Zhiwen Yu, Yifan Shi, and CL Philip Chen. A survey on imbalanced learning: latest research, applications and future directions. *Artificial Intelligence Review*, 57(6): 137, 2024.

Zijian Dong, Yilei Wu, Chongyao Chen, Yingtian Zou, Yichi Zhang, and Juan Helen Zhou. Improve representation for imbalanced regression through geometric constraints. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5082–5091, 2025.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Yu Gong, Greg Mori, and Frederick Tung. Ranksim: Ranking similarity regularization for deep imbalanced regression. *arXiv preprint arXiv:2205.15236*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Haiyang Huang, Newsha Ardalani, Anna Sun, Liu Ke, Hsien-Hsin S Lee, Shruti Bhosale, Carole-Jean Wu, and Benjamin Lee. Toward efficient inference for mixture of experts. *Advances in Neural Information Processing Systems*, 37:84033–84059, 2024.

Mahsa Keramati, Lili Meng, and R David Evans. Conr: Contrastive regularizer for deep imbalanced regression. *arXiv preprint arXiv:2309.06651*, 2023.

Hyunji Lee, Luca Soldaini, Arman Cohan, Minjoon Seo, and Kyle Lo. Routerretriever: Routing over a mixture of expert embedding models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11995–12003, 2025.

Jongin Lim, Sucheol Lee, Daeho Um, Sung-Un Park, and Jinwoo Shin. Prime: Deep imbalanced regression with proxies. In *Forty-second International Conference on Machine Learning*, 2025.

Saeed Masoudnia and Reza Ebrahimpour. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42(2):275–293, 2014.

Ruizhi Pu, Gezheng Xu, Ruiyi Fang, Bing-Kun Bao, Charles Ling, and Boyu Wang. Leveraging group classification with descending soft labeling for deep imbalanced regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19978–19985, 2025.

Jiawei Ren, Mingyuan Zhang, Cunjun Yu, and Ziwei Liu. Balanced mse for imbalanced visual regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7926–7935, 2022.

Amelie Royer, Ilia Karmanov, Andrii Skliar, Babak Ehteshami Bejnordi, and Tijmen Blankevoort. Revisiting single-gated mixtures of experts. *arXiv preprint arXiv:2304.05497*, 2023.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Michael Steininger, Konstantin Kobs, Padraig Davidson, Anna Krause, and Andreas Hotho. Density-based weighting for imbalanced regression. *Machine Learning*, 110(8):2187–2211, 2021.

Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 409–424, 2018.

Ziyan Wang and Hao Wang. Variational imbalanced regression: Fair uncertainty quantification via probabilistic smoothing. *Advances in Neural Information Processing Systems*, 36:30429–30452, 2023.

Haipeng Xiong and Angela Yao. Deep imbalanced regression via hierarchical classification adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23721–23730, 2024.

Yuzhe Yang, Kaiwen Zha, Yingcong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In *International conference on machine learning*, pp. 11842–11851. PMLR, 2021.

Kaiwen Zha, Peng Cao, Jeany Son, Yuzhe Yang, and Dina Katabi. Rank-n-contrast: Learning continuous representations for regression. *Advances in Neural Information Processing Systems*, 36: 17882–17903, 2023.

Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9):10795–10816, 2023.

Yiping Zhang, Yuntao Shou, Wei Ai, Tao Meng, and Keqin Li. Groupface: Imbalanced age estimation based on multi-hop attention graph convolutional network and group-aware margin optimization. *IEEE Transactions on Information Forensics and Security*, 20:605–619, 2024.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114, 2022.

A APPENDIX

A.1 NOTATION SUMMARY

Table A.1 summarizes the core notation used in the main paper and this supplement. Unless noted otherwise, $i, j, k \in \{1, \ldots, B\}$ index mini-batch samples and l indexes tapped backbone depths. Expert predictions and mixture weights use $l = 1, \ldots, L$; gate outputs and DMA teacher targets use $l = 1, \ldots, L-1$, with the final depth receiving the residual stick mass. The label-dependent DMA teacher is training-only, while inference uses the label-free student gates.

Table A.1: Summary of notation.

Symbol	Description
x_i / y_i	Input sample and continuous regression target for sample i
F	Backbone network (e.g., ResNet)
E_l	Backbone stage (depth) l; $z_{l,i}$ is the representation at depth l
$\tilde{z}_{l,i}$	ℓ_2-normalized representation $z_{l,i}/\|z_{l,i}\|_2$
h_l / $\hat{y}_i^{(l)}$	Depth-wise expert head and its prediction at depth l
$\hat{y}_i$	Final mixture prediction $\sum_{l=1}^{L} \pi_i^{(l)} \hat{y}_i^{(l)}$
g_l / $p_i^{(l)}$	Student gate and its pre-stick-breaking break probability at depth l
$\pi_i^{(l)}$	Mixture weight for expert l after stick-breaking; $\pi_i^{(l)} \geq 0$ and $\sum_l \pi_i^{(l)} = 1$
p_i / s_i	Gate-output vector $\{p_i^{(l)}\}_{l=1}^{L-1}$ and teacher-target vector $\{s_{l,i}^{\text{teach}}\}_{l=1}^{L-1}$
$\mathrm{SB}(\cdot)$	Stick-breaking map from $L-1$ break probabilities to L mixture weights
π_i / π_i^{teach}	Student and teacher mixture-weight vectors, $\mathrm{SB}(p_i)$ and $\mathrm{SB}(s_i)$
a_{ij}^{y} / $P_i^{y}(\cdot)$	Label-induced logits and neighborhood distribution in a mini-batch
$a_{ij}^{z,l}$ / $P_i^{z,l}(\cdot)$	Feature-induced logits and neighborhood distribution at depth l
$m_{l,i}$	KL mismatch $m_{l,i} = \mathrm{KL}(P_i^{y} \| P_i^{z,l})$
$s_{l,i}^{\text{teach}}$	Teacher score $s_{l,i}^{\text{teach}} = \exp(-m_{l,i})$
τ_y, τ_z	Temperatures for label/feature neighborhood softmax
$\ell(\hat{y}, y)$	Base per-sample regression loss
$\mathcal{L}_{\text{main}}$	Main regression loss on the mixture prediction
$\mathcal{L}_{\text{gate}}$	Gate distillation loss between student p and teacher targets
$\mathcal{L}_{\text{aux}}$	Route-weighted auxiliary loss for specialization
$\mathcal{L}$	Total objective $\mathcal{L}_{\text{main}} + \lambda_{gate}\mathcal{L}_{\text{gate}} + \lambda_{aux}\mathcal{L}_{\text{aux}}$
$\lambda_{gate}, \lambda_{aux}$	Weights of the gate-distillation and expert-specialization losses
B	Mini-batch size
L	Number of tapped backbone depths
C_l	Feature dimension at depth l
$\mathcal{L}_t = \{1, \ldots, L-1\}$	Depth indices evaluated by the DMA teacher during training
$\mathrm{stopgrad}(\cdot)$	Operator that treats its argument as constant during backpropagation, used for teacher targets and routing weights

A.2 TRAINING ALGORITHM

Algorithm 1 summarizes one mini-batch training step, including the DMA teacher computed from that batch.

Algorithm 1 DMG-MoE Training Step

Input: mini-batch $\{(x_i, y_i)\}_{i=1}^{B}$; backbone stages (depths) $\{E_l\}_{l=1}^{L}$; expert heads $\{h_l\}_{l=1}^{L}$; student gates $\{g_l\}_{l=1}^{L-1}$.
Hyperparameters: temperatures τ_y, τ_z; loss weights $\lambda_{gate}, \lambda_{aux}$; target clamp ϵ.
Forward backbone: $z_{1,i} \leftarrow E_1(x_i)$; $z_{l,i} \leftarrow E_l(z_{l-1,i})$ for $l = 2, \dots, L$.
Depth-wise predictions: $\hat{y}_i^{(l)} \leftarrow h_l(z_{l,i})$ for $l = 1, \dots, L$.
Label neighborhood: $P_i^y \leftarrow \mathrm{softmax}_{j \neq i}(-|y_i - y_j|/\tau_y)$ for all i.
Initialize $\mathcal{L}_{\mathrm{gate}} \leftarrow 0$.
for $l = 1$ **to** $L - 1$ **do**
  $\tilde{z}_{l,i} \leftarrow z_{l,i}/\|z_{l,i}\|_2$ for all i.
  $P_i^{z,l} \leftarrow \mathrm{softmax}_{j \neq i}\Big(\tilde{z}_{l,i}^{\top} \tilde{z}_{l,j}/\tau_z\Big)$ for all i.
  $s_{l,i}^{\mathrm{teach}} \leftarrow \mathrm{clamp}\Big(\exp\Big[-\mathrm{KL}(P_i^y \| P_i^{z,l})\Big], \epsilon, 1 - \epsilon\Big)$ for all i.
  $p_i^{(l)} \leftarrow \sigma\big(g_l(z_{l,i})\big)$ for all i.
  $\mathcal{L}_{\mathrm{gate}} \mathrel{+}= \frac{1}{B} \sum_{i=1}^{B} \Big\| p_i^{(l)} - \mathrm{stopgrad}(s_{l,i}^{\mathrm{teach}}) \Big\|_2^2$.
end for
Stick-breaking routing: $\pi_i \leftarrow \mathrm{SB}(p_i)$ for all i.
Mixture prediction: $\hat{y}_i \leftarrow \sum_{l=1}^{L} \pi_i^{(l)} \hat{y}_i^{(l)}$.
Main loss: $\mathcal{L}_{\mathrm{main}} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \ell(\hat{y}_i, y_i)$.
Route-weighted auxiliary: $\mathcal{L}_{\mathrm{aux}} \leftarrow \frac{1}{B} \sum_{i=1}^{B} \sum_{l=1}^{L} \mathrm{stopgrad}(\pi_i^{(l)})\, \ell(\hat{y}_i^{(l)}, y_i)$.
Total: $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{main}} + \lambda_{gate} \mathcal{L}_{\mathrm{gate}} + \lambda_{aux} \mathcal{L}_{\mathrm{aux}}$.
Update: apply one Adam step to all trainable parameters using $\nabla \mathcal{L}$.

A.3 Additional Theoretical Analysis

This appendix expands the theoretical discussion in Section 3.5 of the main paper.

Proof of the neighborhood-distortion bound (Eq. 14 in the main paper). Assume the label range is non-degenerate, i.e., $y_{\max} > y_{\min}$, so that $f_i(j) = |y_j - y_i|/(y_{\max} - y_{\min}) \in [0, 1]$ is well defined. Let $\mathrm{TV}(P, Q) = \frac{1}{2}\|P - Q\|_1$. For any bounded function f,

$$|\mathbb{E}_P[f] - \mathbb{E}_Q[f]| \leq \|f\|_\infty \, \|P - Q\|_1 = 2\|f\|_\infty \, \mathrm{TV}(P, Q).$$

Because $\|f_i\|_\infty \leq 1$,

$$\Big|\mathbb{E}_{P_i^{z,l}}[f_i] - \mathbb{E}_{P_i^y}[f_i]\Big| \leq 2\,\mathrm{TV}(P_i^y, P_i^{z,l}).$$

Applying Pinsker's inequality, $\mathrm{TV}(P, Q) \leq \sqrt{\frac{1}{2}\mathrm{KL}(P\|Q)}$, gives

$$\Big|\mathbb{E}_{P_i^{z,l}}[f_i] - \mathbb{E}_{P_i^y}[f_i]\Big| \leq \sqrt{2\,\mathrm{KL}(P_i^y \| P_i^{z,l})} = \sqrt{2\, m_{l,i}},$$

which proves Eq. 14 in the main paper.

Lipschitz continuity of stick-breaking. Let $\Delta_L = \{q \in \mathbb{R}_{\geq 0}^{L} : \sum_{l=1}^{L} q^{(l)} = 1\}$. The stick-breaking equations in the main paper define a map $\mathrm{SB} : [0, 1]^{L-1} \to \Delta_L$ from break probabilities $p = \{p^{(l)}\}_{l=1}^{L-1}$ to mixture weights $\pi = \{\pi^{(l)}\}_{l=1}^{L}$. Let $J_{\mathrm{SB}}(p) \in \mathbb{R}^{L \times (L-1)}$ denote its Jacobian and define $a_k(p) = \prod_{r=1}^{k-1}(1 - p^{(r)})$, with $a_1(p) = 1$. For the k-th Jacobian column, outputs before depth k have zero derivative, while the derivative at depth k and the absolute derivatives of all later outputs satisfy

$$\left|\frac{\partial \pi^{(k)}}{\partial p^{(k)}}\right| = a_k(p), \qquad \sum_{l=k+1}^{L} \left|\frac{\partial \pi^{(l)}}{\partial p^{(k)}}\right| = a_k(p).$$

The second identity follows because the later stick-breaking weights sum to the residual mass. Thus every Jacobian column has absolute sum $2a_k(p) \leq 2$, giving $\|J_{\mathrm{SB}}(p)\|_{1 \to 1} \leq 2$. For

$p, p' \in [0,1]^{L-1}$, integrating the Jacobian along the line segment between them yields

$$\begin{aligned} \|\mathrm{SB}(p) - \mathrm{SB}(p')\|_1 &\le \int_0^1 \left\| J_{\mathrm{SB}}(p' + t(p - p')) \right\|_{1\to 1} \|p - p'\|_1 \, dt \\ &\le 2 \|p - p'\|_1 \le L \|p - p'\|_1, \end{aligned} \tag{A.1}$$

where the final inequality uses $L \ge 2$. This Lipschitz bound relates gate regression error to changes in routing weights. It also gives the strengthened routing-deviation bound

$$\left| \hat{y}_i - \hat{y}_i^{\mathrm{teach}} \right| \le 2M \|p_i - s_i\|_1 \le 2M\sqrt{L-1} \|p_i - s_i\|_2.$$

A.4 Detailed Complexity and Comparison to RankSim

This appendix provides a detailed complexity discussion and a comparison to RankSim (Gong et al., 2022). We focus on RankSim because both methods construct batch-wise pairwise relations among samples, making it the most direct reference for comparing their additional training cost. Their uses of these relations nevertheless differ: RankSim imposes a ranking-based representation objective, whereas DMA uses neighborhood alignment as routing supervision.

A.4.1 DMA Complexity

Training-time compute. Among the DMG-MoE-specific training operations, the pairwise DMA computation is the asymptotically dominant term; the heads and gates scale linearly with B. The label-neighborhood distribution, which is independent of depth, costs $\mathcal{O}(B^2)$ and can be computed once per mini-batch. At each depth $l \in \mathcal{L}_t$, feature normalization costs $\mathcal{O}(BC_l)$, forming the cosine-similarity matrix costs $\mathcal{O}(B^2 C_l)$, and row-wise (log-)softmax and KL evaluation cost $\mathcal{O}(B^2)$. The resulting DMA time complexity is

$$\mathcal{O}\left(B^2 \sum_{l \in \mathcal{L}_t} C_l + B^2 |\mathcal{L}_t| + B^2 \right) = \mathcal{O}\left(B^2 \sum_{l \in \mathcal{L}_t} C_l \right),$$

where the final form suppresses lower-order terms because $C_l \ge 1$.

Comparison to RankSim. Both DMA and RankSim construct a batch-wise feature-similarity matrix, so both incur the common $\mathcal{O}(B^2 C_l)$ term at a given depth. Their relation-matching stages differ: RankSim obtains row-wise ranks using sorting, which costs $\mathcal{O}(B^2 \log B)$ over a batch, whereas DMA applies row-wise (log-)softmax and KL in $\mathcal{O}(B^2)$. Thus DMA removes the sorting stage and uses dense matrix, softmax, and reduction operations. This comparison concerns the relation-matching stage; when C_l is large, pairwise feature-similarity construction can dominate the total cost of either method. For DMA at multiple depths, the cost scales with $|\mathcal{L}_t|$ and the corresponding feature dimensions C_l.

Memory. Both RankSim and DMA involve dense $B \times B$ relation matrices. Because the DMA teacher is detached and depth-specific matrices can be evaluated and released sequentially, its peak additional relation-matrix memory is $\mathcal{O}(B^2)$ rather than $\mathcal{O}(|\mathcal{L}_t| B^2)$. This is an auxiliary-memory bound and excludes the backbone activations required by ordinary training.

Parameter cost. DMG-MoE reuses a single backbone and adds depth-wise heads and gates with parameter cost

$$\mathcal{O}\left(\sum_{l=1}^{L} \mathrm{params}(h_l) + \sum_{l=1}^{L-1} \mathrm{params}(g_l) \right).$$

Top-k neighborhood approximation (optional). For large B, an optional approximation is to retain only k neighbors per anchor before evaluating the (log-)softmax and KL terms. If a dense similarity matrix is still formed, this reduces the subsequent distribution-matching computation and retained neighbor distributions from $\mathcal{O}(B^2)$ to $\mathcal{O}(Bk)$, but the similarity computation remains $\mathcal{O}(B^2 C_l)$ and its peak dense storage remains $\mathcal{O}(B^2)$. Reducing similarity construction toward $\mathcal{O}(BkC_l)$ requires an approximate or sparse neighbor-search procedure that avoids evaluating all pairs. This approximation is not used in the experiments reported in this paper.

A.4.2 Soft Ranking and Neighborhood Regularizers

To isolate the effect of replacing hard, non-differentiable ranking, we construct two differentiable relation-matching regularizers: a soft-ranking variant of RankSim and a neighborhood-alignment variant that repurposes DMA's alignment principle as a representation objective.

Soft RankSim (fully vectorized). Given a mini-batch, we first construct a class-balanced sampled set by discretizing labels into bins and sampling k instances per bin (optionally weighted by w). Let the sampled size be N. We ℓ_2-normalize features so that dot product equals cosine similarity. We further min–max normalize sampled labels to $[0, 1]$ to avoid vanishing gradients in the sigmoid comparisons.

We define the feature and label similarity matrices:

$$S^z_{ab} = \frac{\cos(z_a, z_b)}{\tau_{\text{sim}}}, \qquad S^y_{ab} = -\left|y_a - y_b\right|,$$

where $a, b \in \{1, \ldots, N\}$ and τ_{sim} is a similarity scaling temperature.

Soft-rank operator. For a score vector $s \in \mathbb{R}^N$, we approximate the rank of element j via pairwise sigmoid comparisons:

$$\widetilde{\text{rk}}_{\tau_r}(s)_j = \frac{1}{N} \sum_{k=1}^{N} \sigma\left(\frac{s_k - s_j}{\tau_r}\right),$$

where $\sigma(\cdot)$ is the sigmoid and τ_r is the ranking temperature. The self-comparison term $k = j$ adds the same offset $0.5/N$ to all ranks and therefore does not affect their differences.

We apply $\widetilde{\text{rk}}_{\tau_r}(\cdot)$ row-wise to obtain rank matrices $R^z = \widetilde{\text{rk}}_{\tau_r}(S^z)$ and $R^y = \widetilde{\text{rk}}_{\tau_r}(S^y)$, and minimize

$$\mathcal{L}_{\text{SoftRS}} = \frac{1}{N^2} \left\|R^z - R^y\right\|_F^2.$$

Although this operator avoids explicit sorting and is efficiently vectorized for the sampled batch sizes used here, applying all pairwise comparisons to every similarity row entails $\mathcal{O}(N^3)$ arithmetic.

Soft Neighborhood Alignment. Alternatively, we directly match neighborhood distributions using a softmax over $j \neq i$:

$$P^y_i(j) = \text{softmax}_{j \neq i}\left(-\frac{|y_i - y_j|}{\tau_y}\right), \quad P^z_i(j) = \text{softmax}_{j \neq i}\left(\frac{\cos(z_i, z_j)}{\tau_z}\right),$$

$$\mathcal{L}_{\text{SoftNbr}} = \frac{1}{N} \sum_{i=1}^{N} \text{KL}(P^y_i \| P^z_i).$$

This formulation requires $\mathcal{O}(N^2C + N^2)$ arithmetic for feature dimension C and avoids both sorting and the cubic all-pairs soft-rank comparisons.

A.4.3 Training-time Comparison of Ranking Regularizers

To compare the measured efficiency of the alternative relation-matching formulations, we benchmark (i) Vanilla, (ii) RankSim, (iii) Soft RankSim, and (iv) Soft Neighborhood Alignment under identical settings.

Table A.2 shows that RankSim incurs $1.41\times$ the Vanilla time in this setting due to its per-anchor ranking operations. Soft RankSim takes $1.05\times$ the Vanilla time despite its higher arithmetic complexity, while Soft Neighborhood Alignment takes approximately the same time as Vanilla. Soft Neighborhood Alignment also improves overall MAE and GM relative to RankSim in this controlled comparison. These measurements are specific to the stated batch size, hardware, and implementation; they do not replace the asymptotic analysis above.

Training environment and reproducibility. All training timing experiments are conducted with DistributedDataParallel on four NVIDIA RTX 4090D GPUs using bf16 precision. We replace backbone BatchNorm with SyncBN and employ NVIDIA DALI for GPU-side decoding/augmentation. The training set contains 12,208 samples with batch size 64. Wall-clock time per epoch measures end-to-end training time including the input pipeline under identical settings.

Table A.2: Regularizer trade-off (AgeDB-DIR, ResNet-50). We report MAE/GM (lower is better) and wall-clock time under identical settings. Best results in each column are bold, and second-best results are underlined. Soft RankSim replaces RankSim's hard ranking with a differentiable soft-rank operator, whereas Soft Neighborhood Alignment repurposes DMA's neighborhood-alignment signal as representation regularization rather than routing supervision.

Method	MAE ↓				GM ↓				Time ↓	Rel.
	All	Many	Med.	Few	All	Many	Med.	Few	(s/epoch)	
Vanilla	7.2605	**6.1986**	9.0127	12.4443	4.6330	**3.8856**	6.5326	9.4376	<u>12.8</u>	<u>1.00×</u>
RankSim	7.2681	6.3848	8.5297	<u>12.1219</u>	4.6820	4.1128	<u>5.8120</u>	<u>8.7548</u>	18.1	1.41×
Soft RankSim	**7.1480**	<u>6.2744</u>	<u>8.4275</u>	**11.8602**	**4.4756**	<u>3.9102</u>	**5.6546**	**8.3936**	13.5	1.05×
Soft Neighborhood Alignment	<u>7.1494</u>	6.3068	**8.1843**	12.2452	<u>4.6114</u>	4.0238	5.8242	8.7607	**12.6**	≈**1.00**×

A.5 Implementation, Architectures, and Hyperparameters

Unless otherwise stated, this section describes the ResNet-50 setup used for the AgeDB-DIR and IMDB-WIKI-DIR image benchmarks.

Data preprocessing. We use the official train/validation/test splits and follow the DIR benchmark preprocessing in Yang et al. (2021). All RGB images are resized to 224×224. During training, we apply zero-padding (16 pixels), random 224×224 cropping, and horizontal flipping with probability 0.5. Each RGB channel is then normalized with mean 0.5 and standard deviation 0.5. Validation and test images use only resizing and normalization.

Optimization. For both image benchmarks, the models used in the main comparisons are trained for 90 epochs with Adam, weight decay 10^{-4}, and a step schedule that multiplies the learning rate by 0.1 at epochs 60 and 80. We use batch size 64 and learning rate 2.5×10^{-4} for AgeDB-DIR, and batch size 256 and learning rate 1×10^{-3} for IMDB-WIKI-DIR. For evaluation, we select the checkpoint with the lowest validation MAE.

Depth-wise experts and routing. We tap $L = 4$ depths and use stick-breaking routing with $L-1$ gates; the last depth receives residual mass. All four heads are evaluated for every input and combined by soft routing; DMG-MoE does not use early exiting. Layer dimensions and the shared head/gate design pattern are given in Table A.3.

DMA teacher and gate distillation. For each gated depth $l \in \{1, \dots, L-1\}$, we construct label-induced and feature-induced neighborhoods by temperature-scaled softmax over pairwise distances/similarities (Section 3.2 of the main paper), compute $m_{l,i} = \mathrm{KL}(P_i^y \| P_i^{z,l})$, and define teacher targets $s_{l,i}^{\text{teach}} = \exp(-m_{l,i})$. We evaluate KL using log-softmax and clamp each target to $[\epsilon, 1-\epsilon]$ with $\epsilon = 10^{-6}$ for numerical stability. The student gates are supervised by $\mathcal{L}_{\text{gate}}$ and also receive task gradients through the mixture prediction in $\mathcal{L}_{\text{main}}$. The label-dependent teacher is evaluated only during training and is discarded for validation, testing, and inference.

Losses and hyperparameters. We use Focal-R as the regression loss ℓ in both $\mathcal{L}_{\text{main}}$ and $\mathcal{L}_{\text{aux}}$ (Yang et al., 2021). Default loss weights and teacher temperatures are summarized in Table A.4; their sensitivity is examined in Section A.10.

A.5.1 Model Components and Default Settings

Depth-wise heads and gates. We attach L lightweight regression heads $\{h_l\}_{l=1}^{L}$ to four backbone stages (conv2_x–conv5_x for ResNet-50, $L = 4$). Each head h_l takes the globally average pooled feature $z_l \in \mathbb{R}^{C_l}$ and outputs a scalar prediction. The student router uses $L-1$ gates $\{g_l\}_{l=1}^{L-1}$ to produce one stick-breaking logit per gated depth, with the final depth receiving the residual mass.

Concretely, each head is a single linear layer, while each gate is a one-hidden-layer MLP with hidden dimension 64, BatchNorm, and ReLU, without dropout.

Teacher temperatures. The DMA teacher constructs label- and feature-induced neighborhood distributions using temperature-scaled label distances and feature similarities. Table A.4 reports the

Table A.3: Architectures of depth-wise heads and gates. GAP denotes global average pooling. The final depth uses the residual mass in stick-breaking (no gate).

Depth (stage)	C_l	Head h_l	Gate g_l
conv2_x	256	GAP → Linear(256→1)	GAP → Linear(256→64)→ BN → ReLU → Linear(64→1)
conv3_x	512	GAP → Linear(512→1)	GAP → Linear(512→64)→ BN → ReLU → Linear(64→1)
conv4_x	1024	GAP → Linear(1024→1)	GAP → Linear(1024→64)→ BN → ReLU → Linear(64→1)
conv5_x	2048	GAP → Linear(2048→1)	residual mass (no gate)

default settings used for the image-benchmark results; controlled deviations are stated in the corresponding sensitivity analyses.

Table A.4: Default teacher temperatures and loss hyperparameters for the image benchmarks.

Hyperparameter	Value	Description
τ_y	10	label-space temperature for P_i^y (teacher)
τ_z	0.08	feature-space temperature for $P_i^{z,l}$ (teacher)
β	0.25	Focal-R error-scaling parameter
γ	2.0	Focal-R focusing exponent
λ_{gate}	0.01	gate distillation weight (main-paper Eq. 13)
λ_{aux}	0.1	auxiliary specialization weight (main-paper Eq. 13)

Notes on reproducibility. All backbone, head, and gate parameters are optimized jointly under the same Adam schedule. Within $\mathcal{L}_{\text{aux}}$, routing weights are detached so that this term updates the experts but not the gates. At validation and test time, targets are used only to compute evaluation metrics and are never passed to the model.

A.6 Additional Benchmark and Transfer Results

This section consolidates additional benchmark and transfer results that complement the main experiments. Unless noted otherwise, the experiments use the same training protocol as the main paper (Section 4).

The GM comparison on AgeDB-DIR and IMDB-WIKI-DIR is reported in Table A.5.

Table A.5: GM results on AgeDB-DIR and IMDB-WIKI-DIR. Lower is better.

Method	AgeDB-DIR				IMDB-WIKI-DIR			
	All	Many	Med	Few	All	Many	Med	Few
Vanilla	4.85	4.17	6.51	8.98	4.54	4.14	10.84	18.64
LDS + FDS	4.86	4.57	5.38	6.75	4.40	4.17	7.87	15.77
RankSim	4.71	4.15	5.74	8.92	4.29	**3.92**	9.72	18.02
Bal. MSE	5.01	4.83	5.46	6.30	4.93	4.68	7.90	15.51
Ord. Ent.	4.91	4.28	6.20	9.29	4.47	4.07	10.56	21.11
ConR	4.70	4.13	5.91	8.59	4.43	4.05	9.91	18.55
SRL	<u>4.50</u>	<u>4.12</u>	<u>5.37</u>	<u>6.29</u>	<u>4.28</u>	4.03	**7.28**	<u>15.20</u>
DMG-MoE	**4.39**	**4.10**	**5.07**	**5.69**	**4.27**	<u>3.99</u>	<u>7.74</u>	**13.97**

*Best in **bold**, second best <u>underlined</u>.*

A.6.1 UCI Tabular DIR Benchmarks

Main-paper Table 1 reports the UCI tabular DIR comparison under the SRL protocol. Following Dong et al. (2025), the four-layer MLP comprises three hidden layers with widths 20, 30, and 10 (d–20–30–10–1).

A.6.2 Benchmark Label Distributions

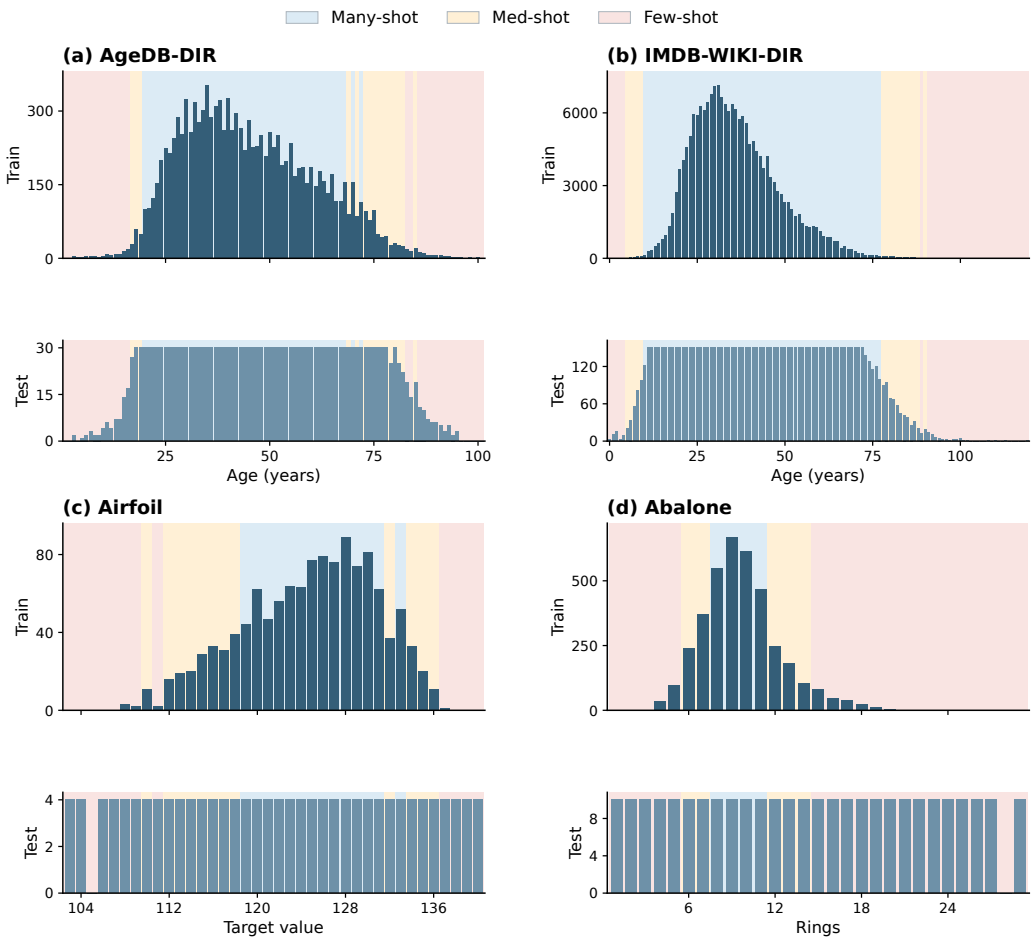


Figure A.1: Training (top) and test (bottom) label counts on the four DIR benchmarks. Background colors mark Many-shot, Med-shot, and Few-shot bins according to training-label frequency using the dataset-specific thresholds in main Section 4.1. Train and test panels use separate count scales. Train/test sample counts are 12,208/2,140 (AgeDB-DIR), 191,494/11,022 (IMDB-WIKI-DIR), 1,233/148 (Airfoil), and 3,784/280 (Abalone).

A.6.3 Backbone Transfer

Table A.6 provides the backbone-transfer MAE and GM results on AgeDB-DIR across ResNet-50, ResNet-152, and ConvNeXt-Tiny.

DMG-MoE improves every MAE column across all three backbones and nearly all GM columns, with only a marginal 0.0060 increase on ResNet-50 Many-shot GM.

Table A.6: Backbone-transfer results on AgeDB-DIR. Best values within each backbone are bold. Gain denotes the absolute reduction from Vanilla; positive values indicate improvement.

Backbone	Method	MAE ↓				GM ↓			
		All	Many	Med.	Few	All	Many	Med.	Few
ResNet-50	Vanilla	7.2008	6.6188	8.1644	10.0334	4.5441	**4.0908**	5.4800	7.3034
	DMG-MoE	**6.9669**	**6.3671**	**8.0524**	**9.6290**	**4.3918**	4.0968	**5.0715**	**5.6878**
	Gain	0.2339	0.2517	0.1120	0.4044	0.1523	−0.0060	0.4085	1.6156
ResNet-152	Vanilla	7.5230	6.9731	8.2396	10.7347	4.7322	4.3697	5.5622	6.4242
	DMG-MoE	**7.1616**	**6.6585**	**7.9692**	**9.6800**	**4.5246**	**4.1810**	**5.3409**	**6.0299**
	Gain	0.3614	0.3146	0.2704	1.0547	0.2076	0.1887	0.2213	0.3943
ConvNeXt-Tiny	Vanilla	8.3492	7.4423	9.8284	12.8227	5.2689	4.6733	6.4110	9.5083
	DMG-MoE	**7.7037**	**6.9053**	**9.2237**	**11.0405**	**4.9336**	**4.4221**	**6.1452**	**7.5575**
	Gain	0.6455	0.5370	0.6047	1.7822	0.3353	0.2512	0.2658	1.9508

A.7 Additional Mechanism Evidence

A.7.1 Depth-dependent Manifold Alignment Curves

We report depth-wise alignment curves to diagnose how local manifold reliability varies across representation depth. For each ResNet bottleneck block depth l (3–4–6–3 blocks, totaling 16 depths), we compute the neighborhood mismatch $m_{l,i} = \mathrm{KL}(P_i^y \| P_i^{z,l})$ as in Eq. (6) of the main paper and report the diagnostic DMA score $\exp(-m_{l,i})$, using the same transform as the teacher score in Eq. (7) of the main paper. Scores are averaged within the Many, Med., and Few subsets. Unlike the training-time teacher, which supervises the $L-1$ gates, this diagnostic is evaluated at *all* 16 bottleneck blocks. DMG-MoE routes over $L=4$ tapped stages.

Evaluation protocol. All statistics are computed on the AgeDB-DIR test split. We form label-induced neighborhoods P_i^y and feature-induced neighborhoods $P_i^{z,l}$ over the evaluated set (full-set neighborhoods), using the default temperatures $(\tau_y, \tau_z) = (10, 0.08)$. Specifically, P_i^y is obtained by applying a temperature-scaled softmax over pairwise label distances (e.g., $-|y_i - y_j|/\tau_y$), and $P_i^{z,l}$ is obtained by a temperature-scaled softmax over feature similarities at depth l (e.g., $\langle \tilde{z}_{l,i}, \tilde{z}_{l,j} \rangle / \tau_z$ with ℓ_2-normalized features). This full-set evaluation is feasible here since the test split is small, though the computation scales quadratically with the evaluated-set size.

Stage transitions and local dips. We mark the four ResNet stages (3–4–6–3) in the background and indicate stage boundaries (dotted lines) where the network performs stride-2 downsampling that halves the spatial resolution. We observe small local dips in alignment around these boundaries. These dips coincide with resolution changes and are followed by alignment recovery in subsequent blocks.

Vanilla backbone. For a vanilla backbone (Figure A.2A), alignment generally increases with depth across all shot groups, suggesting that deeper blocks induce neighborhoods that better respect label geometry. Many-shot samples reach relatively high alignment already at early/intermediate depths, whereas few-shot samples start with substantially lower alignment and show a larger rise toward later blocks.

RankSim comparison. We further compare against RankSim-style representation regularization (Figure A.2B). Its alignment rises through the intermediate blocks but drops in the final stage, most noticeably for Med. and Few samples. The late-stage decline contrasts with the vanilla and DMG-MoE curves.

DMG-MoE. In contrast, DMG-MoE (Figure A.2C) maintains alignment through the later depths and yields stronger deep alignment for tail samples. Together, the curves show that manifold alignment varies with both depth and shot group, motivating sample-dependent allocation across depth-wise experts.

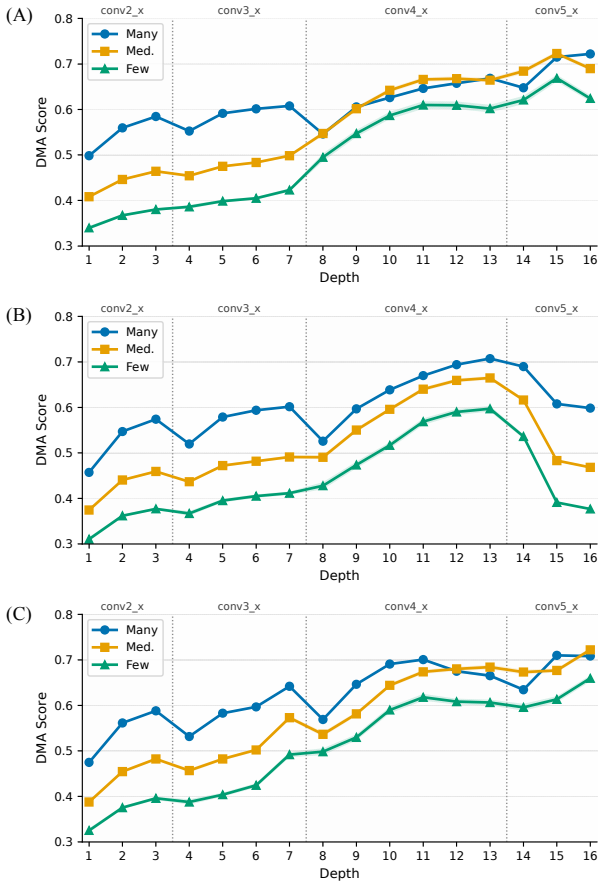


Figure A.2: Depth-dependent manifold alignment. DMA score $\exp(-\mathrm{KL}(P_i^y \| P_i^{z,l}))$ versus ResNet bottleneck depth (3–4–6–3 blocks, 16 depths), averaged over the Many, Med., and Few subsets on the AgeDB-DIR test split. Shaded regions indicate ResNet stages and dotted lines mark stage transitions with stride-2 downsampling. (A) Vanilla backbone. (B) RankSim. (C) DMG-MoE.

A.7.2 Routing Dynamics Over Training

We inspect how DMG-MoE allocates prediction mass across depths as training progresses. We track the expected depth $\mathbb{E}[d] = \sum_{l=1}^{L} l\, \pi^{(l)}$ (overall and by shot group) across epochs.

Figure A.3 shows an initial move toward deeper experts, followed by a sustained *deep-to-shallow* migration. The overall expected depth peaks at approximately 3.15 at epoch 10 and then decreases. By epoch 90, Many samples have shifted to $\mathbb{E}[d]_{\text{Many}} \approx 2.59$, whereas Few samples remain deeper at $\mathbb{E}[d]_{\text{Few}} \approx 2.98$. The persistent gap shows that the learned router retains more prediction mass on deeper experts for tail samples; together with the alignment curves in Section A.7.1, this is consistent with depth-dependent routing.

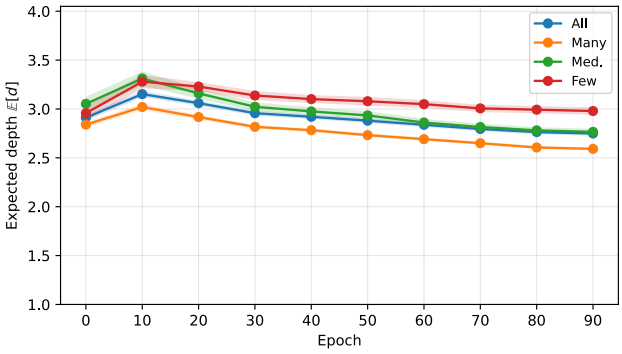


Figure A.3: Routing dynamics over training (AgeDB-DIR). Expected routed depth $\mathbb{E}[d] = \sum_l l\,\pi^{(l)}$ (mean $\pm$ SE) versus training epoch. After peaking at epoch 10, routing gradually shifts shallower, with Few samples retaining a larger expected depth than Many samples.

A.7.3 Teacher–Student Agreement and Routing Sharpness

Setup. To evaluate how faithfully the student router captures the geometry-aware teacher signal, we form a teacher mixture distribution $\pi_i^{\text{teach}} = \mathrm{SB}(s_i)$, where $s_i = \{s_{l,i}^{\text{teach}}\}_{l=1}^{L-1}$ are the teacher targets for the $L-1$ gates (Section 3.3 of the main paper) and $\mathrm{SB}(\cdot)$ is the stick-breaking map (main-paper Eq. 10).

Correlation-based audit (Spearman). Given two length-L vectors $u, v \in \mathbb{R}^L$, we compute their mid-rank vectors $\mathrm{rk}(u), \mathrm{rk}(v)$ (ties receive averaged ranks) and define Spearman correlation as

$$\rho(u,v) = \frac{\left\langle \mathrm{rk}(u) - \overline{\mathrm{rk}(u)},\ \mathrm{rk}(v) - \overline{\mathrm{rk}(v)} \right\rangle}{\left\| \mathrm{rk}(u) - \overline{\mathrm{rk}(u)} \right\|_2 \left\| \mathrm{rk}(v) - \overline{\mathrm{rk}(v)} \right\|_2}, \tag{A.2}$$

where $\overline{\mathrm{rk}(u)}$ denotes the mean rank. We report $\rho(\pi_i, \pi_i^{\text{teach}})$ per sample and aggregate by mean$\pm$standard error (SE).

Diagnostics. We report three complementary diagnostics, overall and per bin when applicable: (i) teacher–student agreement via $\rho(\pi_i, \pi_i^{\text{teach}})$; (ii) deviation from uniform routing via entropy $H(\pi_i)$ and $\mathrm{KL}(\pi_i \| \mathrm{Unif})$; and (iii) top-1 routing mass $\max_l \pi_i^{(l)}$. Here, $H(\pi_i) = -\sum_l \pi_i^{(l)} \log \pi_i^{(l)}$.

For $L{=}4$, uniform routing yields $H(\pi) = \log 4 \approx 1.386$ and $\mathrm{KL}(\pi \| \mathrm{Unif}) = 0$. Sharper routing corresponds to smaller entropy and larger KL/top-1 mass. Table A.7 aggregates the diagnostics by shot group, and Figure A.4 resolves them across age bins. Teacher–student agreement is substantial overall ($\rho \approx 0.611$) and decreases from Many to Few, while routing becomes more concentrated on the sparser subsets (lower $H(\pi)$ and higher KL/top-1 mass). These results show that the student preserves a substantial portion of the teacher's depth ordering and that routing sharpness varies across the label distribution.

Subset	n	$\rho(\pi, \pi^{\text{teach}}) \uparrow$	$H(\pi) \downarrow$	$\mathrm{KL}(\pi \| \mathrm{Unif}) \uparrow$	$\max_l \pi^{(l)} \uparrow$
ALL	12160	0.611$\pm$0.004	0.891$\pm$0.002	0.495$\pm$0.002	0.609$\pm$0.001
Many	11111	0.622$\pm$0.004	0.925$\pm$0.002	0.462$\pm$0.002	0.589$\pm$0.001
Med.	827	0.506$\pm$0.014	0.572$\pm$0.007	0.814$\pm$0.007	0.797$\pm$0.004
Few	222	0.468$\pm$0.027	0.412$\pm$0.014	0.974$\pm$0.014	0.875$\pm$0.006

Table A.7: Teacher–student agreement and routing sharpness (AgeDB-DIR). We report mean$\pm$SE over samples. Higher ρ indicates better alignment; lower entropy indicates sharper routing.

Figure A.4 reports the same diagnostics across age bins. Agreement is highest over much of the central age range and becomes lower and more variable near sparse extremes, while entropy decreases and top-1 mass increases toward the extremes.

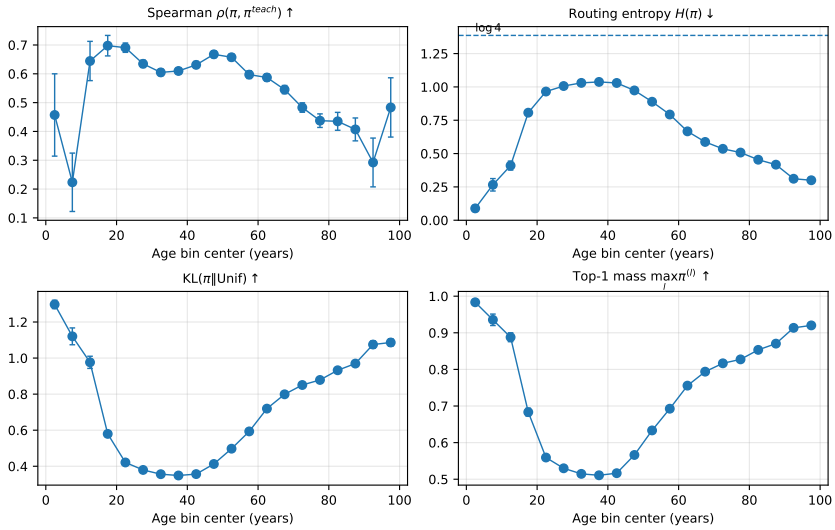


Figure A.4: Teacher–student agreement and routing sharpness across age bins (AgeDB-DIR). Points show bin means and error bars show SE where displayed.

A.7.4 Oracle Single-Depth Selection Protocol

We define a diagnostic that compares the learned soft mixture with per-sample selection of a single expert. For each test sample i, the oracle selects the depth that minimizes the per-head error:

$$l_i^\star = \arg\min_l \ \ell(\hat{y}_i^{(l)}, y_i). \tag{A.3}$$

We then evaluate the oracle prediction $\hat{y}^{(l_i^\star)}$ and compare it with the learned mixture prediction $\hat{y} = \sum_l \pi^{(l)} \hat{y}^{(l)}$. This provides a diagnostic comparison between per-sample single-expert selection and the learned soft mixture. Figure A.5 reports the resulting MAE and GM differences across age bins. Both gaps remain positive across all age bins and follow nearly identical trends. They are relatively stable in the lower and central age ranges, then widen steadily beyond age 40, with the largest separation observed in the oldest bins.

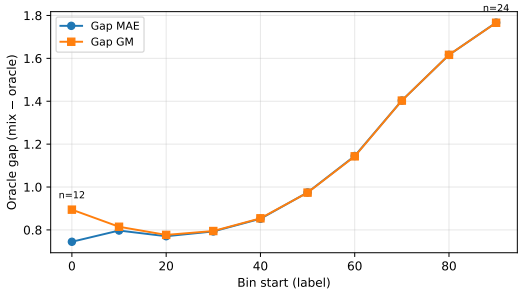


Figure A.5: Oracle single-depth diagnostic across age bins (AgeDB-DIR). The curves show MAE and GM differences between the learned mixture and per-sample oracle single-depth selection. Positive values indicate lower error for oracle selection.

A.7.5 Depth-wise Expert Specialization

To examine whether depth-wise heads learn distinct error profiles rather than collapsing to similar predictors, we analyze bin-aggregated head errors and routing allocation. In this section we instantiate $\ell(\hat{y}, y) = |\hat{y} - y|$.

For each target bin b and expert depth l, we compute: (i) **Per-head error** $\bar{e}_{b,l} = \mathbb{E}_{i\in b}[\ell(\hat{y}_i^{(l)}, y_i)]$, (ii) **Routing mass** $\bar{\pi}_{b,l} = \mathbb{E}_{i\in b}[\pi_i^{(l)}]$, (iii) **Routed-error diagnostic** $\bar{c}_{b,l} = \mathbb{E}_{i\in b}[\pi_i^{(l)}\,\ell(\hat{y}_i^{(l)}, y_i)]$, and (iv) **Allocation ratio** $\bar{r}_{b,l} = \bar{c}_{b,l} / \sum_k \bar{c}_{b,k}$. We use $\bar{c}_{b,l}$ and $\bar{r}_{b,l}$ as routing-weighted diagnostics; the mixture loss itself is computed from the final mixture prediction.

Figure A.6 provides evidence of specialization: the identity of the lower-error head varies across label regions (A), and routing mass follows a structured, bin-dependent pattern (B). The routed-error views (C–D) further show that the deepest head h_4 accounts for most of the contribution in the youngest and oldest bins, whereas the contribution is distributed more broadly across depths in several central bins. Thus, the routing-weighted error allocation varies systematically across both expert depth and target region.

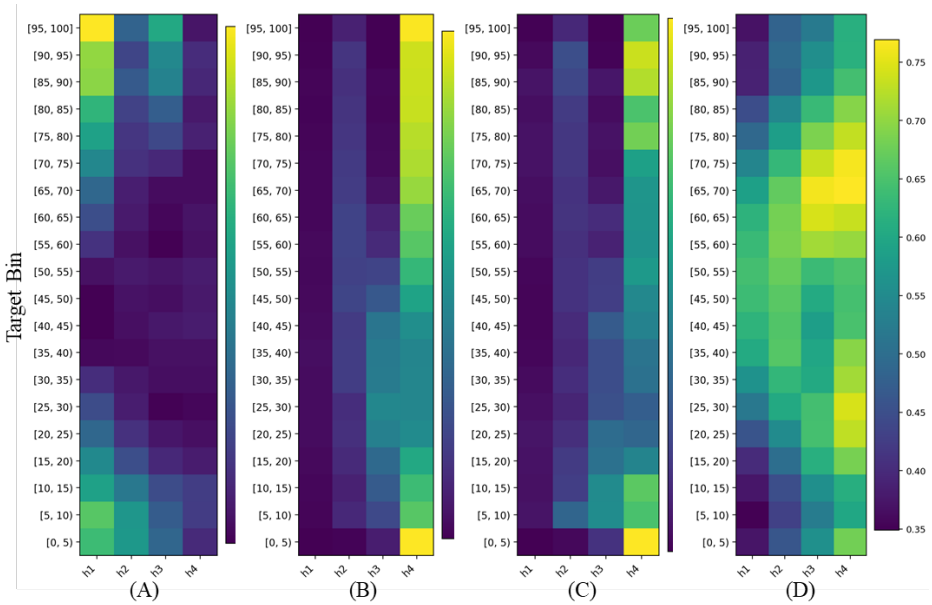


Figure A.6: Depth-wise specialization diagnostics. Heatmaps over target bins (rows) and depth experts (columns): (A) per-head error $\bar{e}_{b,l}$, (B) routing mass $\bar{\pi}_{b,l}$, (C) routed-error diagnostic $\bar{c}_{b,l}$, and (D) allocation ratio $\bar{r}_{b,l}$.

A.7.6 Per-sample Depth Analysis

Figure A.7 compares DMA, expert error, and routing weights across depths for several AgeDB-DIR validation samples.

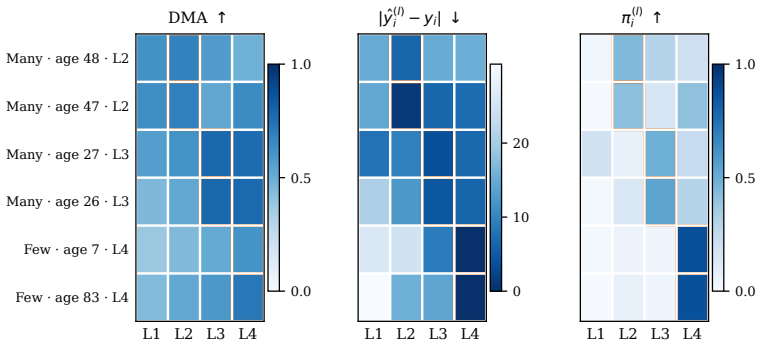


Figure A.7: Per-sample depth analysis on AgeDB-DIR. Each row shows one validation sample across four experts. Darker blue indicates higher DMA, lower expert error, or higher routing weight; orange outlines highlight the corresponding depth.

A.8 Extended Ablations

This appendix provides the full set of ablations to disentangle: (i) capacity vs. guidance, (ii) whether always using the deepest depth is sufficient, (iii) routing parameterization (stick-breaking vs. softmax), (iv) whether task gradients into the gate matter, (v) teacher variants and shuffled-teacher controls, (vi) auxiliary-loss weighting variants for specialization, and (vii) the number of routed expert heads. All experiments in this section use AgeDB-DIR; Sections A.8.1–A.8.6 report MAE, and Section A.8.7 additionally reports GM.

A.8.1 Capacity vs. Guidance

We first compare DMG-MoE against (A1) a multi-head model with uniform mixing ($\pi^{(l)} = 1/L$), which retains the same backbone and four depth-wise heads but removes sample-wise routing, and (A2) a learned gate trained without the DMA teacher. Table A.8 shows stepwise improvements from A1 to A2 and then to full DMG-MoE in every reported subset. Together with the deepest-only result in Table A.9, these comparisons show that adding multiple heads without effective sample-wise routing is insufficient.

Variant	All	Many	Med.	Few
Multi-head + uniform mix (A1)	7.51	6.90	8.49	10.64
Learned gate (no teacher) (A2)	7.16	6.52	8.17	10.32
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

Table A.8: Capacity and guidance ablation. Uniform mixing, unguided routing, and full DMG-MoE.

A.8.2 Is Deep Always Better?

A natural baseline is to always use the final representation, i.e., (A3) fixing $\pi = [0, 0, 0, 1]$. Table A.9 shows that DMG-MoE consistently outperforms this deepest-only variant, reducing All and Few MAE by 0.23 and 0.40, respectively. This result supports adapting the prediction depth instead of using the final representation for every sample.

Variant	All	Many	Med.	Few
Always deepest (A3)	7.20	6.62	8.16	10.03
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

Table A.9: Deepest-only ablation. Always using the final representation vs. full DMG-MoE.

A.8.3 Routing Parameterization: Softmax vs. Stick-breaking

We next study routing parameterization by replacing stick-breaking with a standard softmax mixture over all depths (A4). Softmax routing performs markedly worse in every reported subset (Table A.10), with the largest gap on *Few*. Stick-breaking differs structurally by allocating mass sequentially from shallow to deep, whereas softmax normalizes all depth logits jointly. The result establishes that the ordered parameterization is beneficial in this setting.

Variant	All	Many	Med.	Few
Softmax routing (A4)	7.45	6.68	8.93	10.70
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

Table A.10: Routing-parameterization ablation. Softmax routing vs. stick-breaking routing.

A.8.4 Do Gates Need Task Gradients?

While the student gate is primarily trained by distilling the DMA teacher, the routing weights π also participate in the final mixture prediction and therefore receive task gradients from the main

regression loss. We ablate a *teacher-only gate* variant (A5), in which π is detached from the mixture prediction (i.e., the main task loss does not backpropagate into the gate). As shown in Table A.11, removing task gradients degrades every subset, increasing All and Few MAE by 0.46 and 1.43, respectively. DMA supervision and task gradients are therefore complementary: the former provides geometry-based routing targets, while the latter optimizes routing through the final prediction error.

Variant	All	Many	Med.	Few
Teacher-only gate (A5)	7.43	6.71	8.58	11.06
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

Table A.11: Task-gradient ablation. Teacher-only gate vs. full DMG-MoE.

A.8.5 Teacher Variants and Shuffled-teacher Control

We evaluate how the design of the training-time teacher affects routing. In addition to our default DMA teacher (KL alignment between soft label/feature neighborhoods), we consider (A6) a Soft RankSim teacher based on the per-sample soft-rank discrepancy underlying Section A.4.2, and (A7) a marginal-improvement teacher with target $\sigma(5[m_{l+1,i} - m_{l,i}])$, where $m_{l,i}$ is the KL mismatch at depth l.

As a sanity check, we perform a control experiment using a *shuffled teacher* (A8), where teacher scores are randomly permuted within each batch before distillation. This preserves the score distribution but destroys sample-wise geometric information. If gains were mainly due to regularizing the gate with a particular score distribution, shuffling would have limited impact. Table A.12 shows that A6 and A7 outperform the shuffled control on All, Many, and Med., although the shuffled control has lower Few MAE than either alternative. The full DMA teacher nevertheless yields the best value in every reported column, including Few, and is therefore our default.

Variant	All	Many	Med.	Few
Soft RankSim teacher (A6)	7.26	6.63	8.16	10.67
Marginal-improvement teacher (A7)	7.16	6.48	8.37	10.19
Shuffled teacher (A8)	7.37	6.66	8.56	10.01
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**

Table A.12: Teacher-design ablation. DMA, Soft RankSim, marginal-improvement, and shuffled teachers.

A.8.6 Auxiliary Weighting Variants for Specialization

We compare the unweighted auxiliary objective (A9), the sample-weighted objective used by DMG-MoE, and a sample-weighted plus depth-balanced variant (A10).

Let $\ell(\hat{y}, y)$ be the per-sample regression loss used in $\mathcal{L}_{\text{aux}}$.

(A9) Unweighted auxiliary.

$$\mathcal{L}_{\text{aux}}^{\text{unw}} = \frac{1}{B} \sum_{i=1}^{B} \sum_{l=1}^{L} \ell(\hat{y}_i^{(l)}, y_i).$$

Default sample-weighted auxiliary. This is the default objective in main-paper Eq. (12), which weights each sample's auxiliary supervision by its routed mass:

$$\mathcal{L}_{\text{aux}}^{\text{sw}} = \frac{1}{B} \sum_{i=1}^{B} \sum_{l=1}^{L} \text{stopgrad}(\pi_i^{(l)})\, \ell(\hat{y}_i^{(l)}, y_i).$$

No additional depth-level rebalancing is applied, depth preferences emerge from the learned routing.

(A10) Sample-weighted + depth-balanced auxiliary. To further rebalance auxiliary supervision across depths, we rescale each depth by the inverse of its mean routed mass in the batch:

$$\bar{\pi}^{(l)} = \frac{1}{B}\sum_{i=1}^{B} \mathrm{stopgrad}(\pi_i^{(l)}), \qquad \alpha_l = \frac{1}{\bar{\pi}^{(l)} + \epsilon},$$

and use

$$\mathcal{L}_{\text{aux}}^{\text{sw-db}} = \frac{1}{B}\sum_{i=1}^{B}\sum_{l=1}^{L} \mathrm{stopgrad}(\alpha_l\, \pi_i^{(l)})\, \ell(\hat{y}_i^{(l)}, y_i).$$

Compared with the unweighted A9 variant, the default objective reduces MAE by 0.32, 0.30, 0.03, and 1.36 on All, Many, Med., and Few, respectively. A10 improves Few over A9 by 0.30, but slightly degrades All and Many and increases Med. MAE by 0.24. Overall, the default route-weighted objective performs best across all subsets, indicating that additional depth-level rebalancing provides no further benefit in this setting.

Variant	All	Many	Med.	Few
Unweighted auxiliary (A9)	<u>7.29</u>	<u>6.67</u>	<u>8.08</u>	10.99
DMG-MoE (full)	**6.97**	**6.37**	**8.05**	**9.63**
Sample-weighted + depth-balanced (A10)	7.33	6.68	8.32	<u>10.69</u>

Table A.13: Auxiliary-weighting ablation. Unweighted, sample-weighted, and sample-weighted plus depth-balanced objectives.

A.8.7 Expert-Head Count Ablation

Table A.14 ablates the number of routed depth-wise experts. The full four-depth configuration achieves the best MAE and GM across all subsets. Removing L_1 increases All MAE by 0.8399 and degrades Few MAE and GM by 2.1266 and 2.0398, respectively. Removing both L_1 and L_2 further degrades All, Many, and Med. MAE and all GM metrics, although Few MAE improves by 0.3653 relative to removing L_1 alone. These results show that DMG-MoE benefits from retaining the complete shallow-to-deep expert hierarchy rather than relying only on deeper experts, yielding more robust performance across data-density regions.

Variant	MAE ↓				GM ↓			
	All	Many	Med.	Few	All	Many	Med.	Few
DMG-MoE (full)	**6.9669**	**6.3671**	**8.0524**	**9.6290**	**4.3918**	**4.0968**	**5.0715**	**5.6878**
DMG-MoE (without L_1)	7.8068	7.0561	8.9424	11.7556	4.9628	4.5231	5.8048	7.7276
DMG-MoE (without L_1–L_2)	8.0904	7.4205	9.1850	11.3903	5.2636	4.8122	6.1440	8.0028

Table A.14: Expert-head-count ablation. Full DMG-MoE uses L_1–L_4; the variants progressively remove shallower experts.

A.9 Efficiency and Profiling Details

A.9.1 Inference Overhead

DMG-MoE augments a standard backbone with lightweight depth-wise heads and small gating MLPs, and discards the DMA teacher at inference. Thus, deployment-time overhead comes solely from evaluating L heads and $L-1$ label-free gates.

Profiling protocol. We report parameter count, profiler-estimated FLOPs, and *pure model forward* latency under identical single-GPU settings. Latency excludes dataloader and batch-unpacking overhead. All measurements use bf16 inputs of resolution $3\times224\times224$; after warm-up iterations, we average repeated runs with GPU synchronization.

Method	Params (M)	FLOPs (G)	Latency (ms/img)
Backbone	23.53	91.06	7.771
DMG-MoE	23.65	91.06	9.237

Table A.15: Inference overhead at batch size 1. Latency measures pure model forward time on a single GPU.

As shown in Table A.15, DMG-MoE adds only 0.12M parameters (0.5%) and increases latency from 7.771 to 9.237 ms/image (18.9%). This measured increase reflects the additional routing and multi-head evaluation at deployment time.

Inference overhead at batch size 64. Table A.16 provides the corresponding batched-inference measurements.

Method	Params (M)	FLOPs (G)	Latency (ms/iter; ms/img)
Backbone	23.53	91.06	7.614 ms/iter (0.119 ms/img)
DMG-MoE	23.65	91.06	9.312 ms/iter (0.146 ms/img)

Table A.16: Inference overhead at batch size 64. Latency measures pure model forward time on a single GPU.

At batch size 64, per-iteration latency increases from 7.614 to 9.312 ms (22.3%), corresponding to 0.119 versus 0.146 ms/image. The two batch sizes therefore exhibit similar relative latency overhead.

Remarks on FLOPs estimation. Profiler-reported FLOPs may not account for all PyTorch operators, which can lead to identical estimates even when runtime differs. We therefore treat the identical 91.06G estimates as approximate and use measured latency as the primary deployment metric.

A.9.2 End-to-End Training Wall-Clock Time

Table A.17 reports the training-time comparison and clarifies its measurement basis. The values come from complete AgeDB-DIR runs with ResNet-50 using distributed training on four NVIDIA RTX 4090D GPUs. These timing runs are separate from the 90-epoch main-result runs and use the same validation-MAE early-stopping rule for all methods, without a fixed epoch cap. Because the runs stop at different epochs, their total elapsed times reflect different run lengths and are not directly comparable; overhead is computed from seconds per epoch using the logged, unrounded elapsed times.

Method	Early Stop Epoch	Total Time (s)	s/Epoch	Overhead
Vanilla	80	540.14	6.75	$+0.0\%$
LDS	82	742.84	9.06	$+34.2\%$
FDS	38	535.03	14.08	$+108.5\%$
RankSim	84	1014.53	12.08	$+78.9\%$
DMG-MoE	96	737.07	7.68	$+13.7\%$

Table A.17: End-to-end wall-clock time on AgeDB-DIR. Overhead is relative to Vanilla seconds per epoch.

DMG-MoE requires 7.68 s/epoch, corresponding to 13.7% overhead over Vanilla (6.75 s/epoch). It is also the fastest of the compared imbalance-aware methods and reduces per-epoch time by 36.4% relative to RankSim (12.08 s/epoch). These results show that the training-time DMA teacher adds limited end-to-end overhead despite its pairwise mini-batch computations.

A.10 Sensitivity Analyses

This section examines sensitivity to three components of the experimental setup on AgeDB-DIR: mini-batch size, DMA teacher temperatures, and loss weights.

A.10.1 BATCH-SIZE SENSITIVITY

Batch Size	All	Many	Med.	Few
16	7.14	6.55	8.12	9.91
32	7.03	6.40	8.10	9.96
64	**6.97**	**6.37**	**8.05**	**9.63**
128	7.44	6.72	8.38	11.64
256	8.35	7.41	10.01	12.64

Table A.18: Batch-size sensitivity on AgeDB-DIR (MAE). Best values are bold.

Table A.18 shows that performance is similar for batch sizes 16–64, with the default batch size of 64 yielding the best value in every subset. At 128 and 256, All MAE increases to 7.44 and 8.35, while Few MAE increases from 9.63 to 11.64 and 12.64. Because batch size also changes the optimization dynamics and number of parameter updates per epoch, this sweep does not attribute the differences solely to neighborhood estimation.

A.10.2 TEACHER TEMPERATURE SENSITIVITY

We study the DMA teacher temperatures (τ_y, τ_z) using independently trained temperature-specific checkpoints evaluated on the 2,140-sample AgeDB-DIR test set. We report (i) teacher–student agreement $\rho(\pi, \pi^{\text{teach}})$ (Spearman; Section A.7.3), (ii) student routing sharpness via entropy $H(\pi)$, (iii) teacher target sharpness via entropy $H(\pi^{\text{teach}})$, and (iv) downstream regression performance (MAE/GM). Section A.7.3 instead reports a separate 12,160-sample mechanism audit, so its overall agreement ($\rho = 0.611$) is not directly comparable to the default sweep value reported here ($\rho = 0.504$).

Sweep protocol. The primary evaluation consists of two one-dimensional slices through the default setting. First, we fix $\tau_y = 10$ and use

$$\tau_z \in \{0.01, 0.03, 0.05, 0.08, 0.10, 0.20, 0.50, 1.0, 2.0\}.$$

Second, we fix $\tau_z = 0.08$ and use $\tau_y \in \{1, 5, 10, 15, 20\}$. The heatmaps additionally include the six cross-settings formed by $\tau_y \in \{1, 5, 20\}$ and $\tau_z \in \{0.01, 0.50\}$; white cells denote combinations that were not evaluated. Agreement and entropies are reported as mean±SE over test samples.

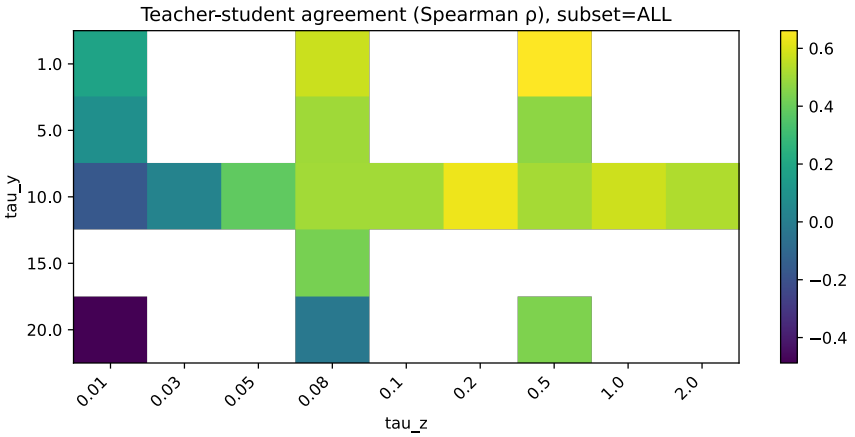
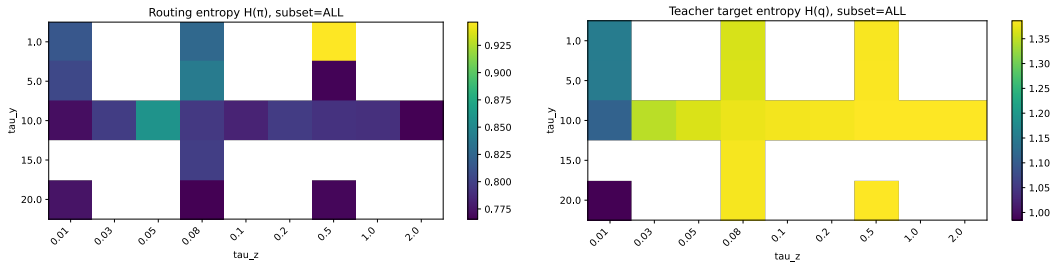


Figure A.8: Teacher–student agreement under the sparse temperature sweep (AgeDB-DIR). Teacher–student agreement $\rho(\pi, \pi^{\text{teach}})$ as a function of (τ_y, τ_z). Colored cells denote evaluated settings; white cells were not evaluated.

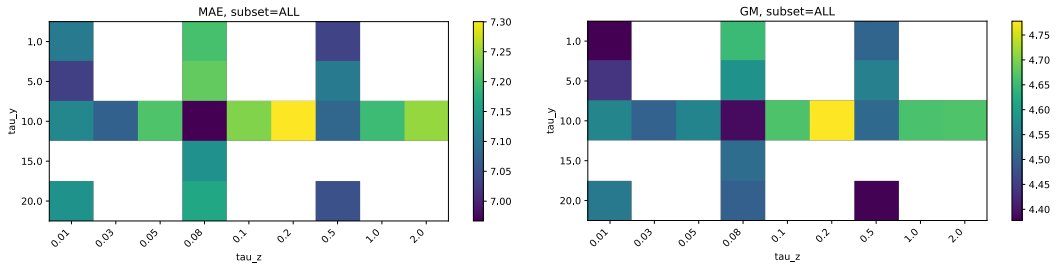
Figure A.8 and Table A.19 show that agreement is strongly temperature-dependent. Along the $\tau_y = 10$ slice, overall ρ rises from -0.173 at $\tau_z = 0.01$ and 0.026 at $\tau_z = 0.03$ to 0.504 at the default $\tau_z = 0.08$, reaching its largest value of 0.631 at $\tau_z = 0.20$.



(Left) Student routing entropy $H(\pi)$. (Right) Teacher target entropy $H(\pi^{\text{teach}})$.

Figure A.9: Routing and teacher-target sharpness under the sparse temperature sweep (AgeDB-DIR). Smaller $H(\pi)$ indicates sharper routing. Larger $H(\pi^{\text{teach}})$ indicates more diffuse teacher targets.

Figure A.9 separates the effect on teacher targets from the learned student routing. Along the $\tau_y = 10$ slice, $H(\pi^{\text{teach}})$ increases from 1.111 at $\tau_z = 0.01$ to approximately $\log 4 = 1.386$ at $\tau_z = 2.0$, whereas $H(\pi)$ varies non-monotonically between 0.766 and 0.859. Thus, increasing τ_z consistently diffuses the teacher targets, but does not induce a monotonic change in student routing sharpness.



(Left) MAE (lower is better). (Right) GM (lower is better).

Figure A.10: Downstream performance under the sparse teacher-temperature sweep (AgeDB-DIR). Colored cells denote evaluated settings.

τ_y	τ_z	$\rho(\pi, \pi^{\text{teach}}) \uparrow$	$H(\pi)$	$H(\pi^{\text{teach}})$	MAE $\downarrow$	GM $\downarrow$
Fix $\tau_y = 10$, *sweep* τ_z						
10	0.01	-0.173±0.012	0.773±0.006	1.111±0.005	7.121	4.559
10	0.03	0.026±0.013	0.799±0.006	1.344±0.001	7.071	4.503
10	0.05	0.379±0.012	0.859±0.005	1.364±0.001	7.208	4.558
10†	0.08†	0.504±0.011	0.796±0.006	1.374±0.000	**6.967**	**4.389**
10	0.10	0.503±0.011	0.783±0.007	1.379±0.000	7.237	4.666
10	0.20	0.631±0.010	0.798±0.007	1.381±0.000	7.300	4.778
10	0.50	0.506±0.010	0.792±0.006	1.385±0.000	7.077	4.514
10	1.00	0.573±0.009	0.791±0.007	1.386±0.000	7.195	4.663
10	2.00	0.522±0.009	0.766±0.006	1.386±0.000	7.248	4.668
Fix $\tau_z = 0.08$, *sweep* τ_y						
1	0.08	0.571±0.010	0.827±0.006	1.361±0.001	7.203	4.650
5	0.08	0.499±0.010	0.841±0.006	1.365±0.001	7.220	4.581
10†	0.08†	0.504±0.011	0.796±0.006	1.374±0.000	**6.967**	**4.389**
15	0.08	0.427±0.012	0.799±0.006	1.381±0.000	7.135	4.520
20	0.08	-0.028±0.012	0.766±0.006	1.381±0.000	7.167	4.500

Table A.19: Temperature sensitivity on the overall test set. We report mean±SE for agreement and entropies, and downstream MAE/GM. † denotes the default setting; bold marks the best MAE and GM within each one-dimensional slice.

τ_y	τ_z	$\rho(\pi, \pi^{\text{teach}}) \uparrow$	$H(\pi)$	$H(\pi^{\text{teach}})$	MAE ↓	GM ↓
Fix $\tau_y = 10$, *sweep* τ_z						
10	0.01	-0.159±0.038	0.608±0.022	1.097±0.018	10.094	6.728
10	0.03	-0.004±0.040	0.615±0.020	1.340±0.005	10.047	7.299
10	0.05	0.303±0.044	0.694±0.018	1.363±0.003	10.465	7.059
10†	0.08†	0.487±0.036	0.573±0.018	1.374±0.001	9.541	**5.747**
10	0.10	0.435±0.035	0.620±0.019	1.377±0.001	10.647	7.395
10	0.20	0.604±0.033	0.534±0.021	1.380±0.000	11.083	7.912
10	0.50	0.399±0.036	0.567±0.020	1.385±0.000	9.985	6.523
10	1.00	0.474±0.035	0.514±0.021	1.386±0.000	**9.447**	6.333
10	2.00	0.474±0.035	0.586±0.020	1.386±0.000	10.326	7.015
Fix $\tau_z = 0.08$, *sweep* τ_y						
1	0.08	0.498±0.035	0.629±0.020	1.360±0.002	9.948	7.277
5	0.08	0.417±0.036	0.603±0.018	1.367±0.002	10.693	7.353
10†	0.08†	0.487±0.036	0.573±0.018	1.374±0.001	**9.541**	**5.747**
15	0.08	0.386±0.038	0.523±0.016	1.380±0.000	9.787	5.953
20	0.08	-0.085±0.038	0.592±0.017	1.381±0.001	10.239	7.051

Table A.20: Temperature sensitivity on the Few-shot test subset. † denotes the default setting; bold marks the best MAE and GM within each one-dimensional slice.

Figure A.10 and Table A.19 show that the default $(\tau_y, \tau_z) = (10, 0.08)$ gives the best overall MAE and GM in both primary slices, even though the largest agreement occurs at $(10, 0.20)$. Table A.20 provides a complementary tail view: the default gives the best Few-shot GM in both slices, while $\tau_z = 1.0$ gives the lowest Few-shot MAE in the τ_z slice. Each sweep point is an independently trained checkpoint, so the default sweep run is separate from the canonical full-model run. This accounts for small differences such as Few MAE 9.541 here versus 9.629 in the canonical result. Overall, the default provides the strongest aggregate regression performance without requiring the temperature that maximizes any single routing diagnostic.

A.10.3 Loss-Weight Sensitivity

Table A.21 reports sensitivity to λ_{aux} and λ_{gate} on AgeDB-DIR. We vary λ_{aux} with $\lambda_{gate} = 0.01$, and vary $\lambda_{gate} \in \{0.001, 0.003, 0.01, 0.1\}$ with $\lambda_{aux} = 0.1$; the table also includes the direct auxiliary-loss ablation $\lambda_{aux} = 0$ and one additional joint setting. The default values $\lambda_{aux} = 0.1$ and $\lambda_{gate} = 0.01$ were selected by validation-MAE grid search.

λ_{aux}	λ_{gate}	MAE ↓				GM ↓			
		All	Many	Med.	Few	All	Many	Med.	Few
Auxiliary-weight sweep ($\lambda_{gate} = 0.01$)									
0	0.01	7.4525	6.7647	8.5781	10.8357	4.8231	4.3448	6.0031	7.0611
0.01	0.01	7.1000	6.5424	7.9562	9.9979	4.4153	**4.0219**	5.1135	7.1038
0.03	0.01	7.2885	6.6444	8.3490	10.4386	4.5997	4.2000	5.4477	6.7994
0.1†	0.01†	**6.9669**	**6.3671**	8.0524	**9.6290**	**4.3918**	4.0968	**5.0715**	**5.6878**
0.3	0.01	7.1060	6.4492	8.1443	10.4377	4.5930	4.1941	5.3934	6.9478
1	0.01	7.3617	6.6600	8.4607	10.9499	4.6579	4.1600	5.7085	7.7211
3	0.01	7.3124	6.6713	8.1569	11.0321	4.6662	4.1921	5.6887	7.4207
Gate-weight sweep ($\lambda_{aux} = 0.1$; default row shown above)									
0.1	0.001	7.0356	6.5219	**7.7629**	9.8768	4.4797	4.1489	5.1433	6.3109
0.1	0.003	7.2717	6.6772	8.2062	10.3022	4.6186	4.2311	5.5901	6.2331
0.1	0.1	7.0951	6.5125	7.9913	10.1189	4.5772	4.1722	5.5000	6.6070
Additional joint setting									
0.01	0.1	7.3246	6.7108	8.3912	10.1725	4.6153	4.1869	5.6530	6.6109

Table A.21: Loss-weight sensitivity on AgeDB-DIR. † denotes the default setting. Bold marks the lowest value in each metric column.

The default setting achieves the lowest overall MAE (6.9669) and GM (4.3918) among all reported settings. Removing the auxiliary loss increases All and Few MAE by 0.4856 and 1.2067, respectively. Figure A.11 further shows that the MAE response to λ_{aux} is non-monotonic, with the default also producing the smallest Few-shot–overall MAE gap in this slice.

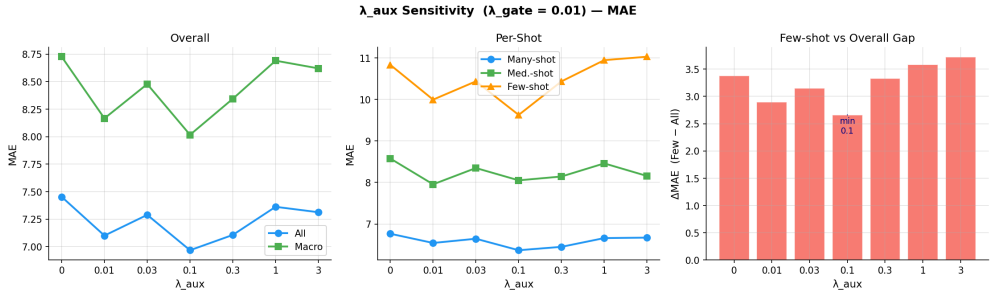


Figure A.11: Auxiliary-loss-weight sensitivity on AgeDB-DIR with $\lambda_{gate} = 0.01$. The panels report overall and macro MAE, shot-group MAE, and the Few-shot–overall MAE gap.

Across the valid gate-weight sweep, All MAE ranges from 6.9669 to 7.2717, and the default $\lambda_{gate} = 0.01$ yields the best All and Few MAE. The lowest Med. MAE occurs at $\lambda_{gate} = 0.001$, confirming that individual shot-group metrics do not vary monotonically with the loss weights. Figure A.12 visualizes these complementary trends.

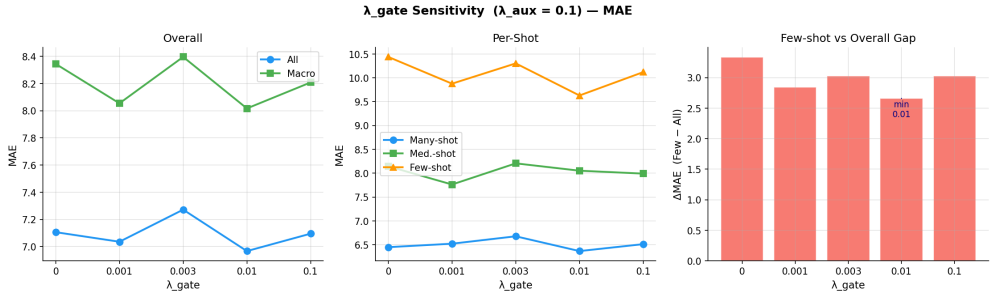


Figure A.12: Gate-distillation-weight sensitivity on AgeDB-DIR with $\lambda_{aux} = 0.1$. The panels report overall and macro MAE, shot-group MAE, and the Few-shot–overall MAE gap.

REFERENCES

Zijian Dong, Yilei Wu, Chongyao Chen, Yingtian Zou, Yichi Zhang, and Juan Helen Zhou. Improve representation for imbalanced regression through geometric constraints. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5082–5091, 2025.

Yu Gong, Greg Mori, and Frederick Tung. Ranksim: Ranking similarity regularization for deep imbalanced regression. *arXiv preprint arXiv:2205.15236*, 2022.

Yuzhe Yang, Kaiwen Zha, Yingcong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In *International conference on machine learning*, pp. 11842–11851. PMLR, 2021.