DisRFM: Polar Riemannian Flow Matching for Structure-Preserving Graph Domain Adaptation

Yingxu Wang
The Chinese University of Hong Kong
yingxv.wang@gmail.com

Xinwang Liu
National University of Defense Technology
xinwangliu@nudt.edu.cn

Mengzhu Wang
Hebei University of Technology
dreamkily@gmail.com

Siyang Gao
The City University of Hong Kong
siyangao@cityu.edu.hk

Nan Yin*
The Education University of Hong Kong
yinnan8911@gmail.com

Abstract

Graph Domain Adaptation (GDA) aims to transfer graph classifiers across domains with both semantic and topological shifts. Existing Euclidean adversarial methods face two challenges: *Structural Degeneration*, where domain confusion entangles and suppresses label-relevant topology, and *Optimization Instability*, where minimax training induces oscillatory gradients under large structural shifts. We propose DisRFM, a geometry-aware GDA framework that addresses these challenges with Riemannian representation learning and flow-based transport. DisRFM embeds graph representations on a constant-curvature manifold and expresses them in geodesic polar coordinates. Polar endpoint regularization calibrates topology-sensitive radial scales via univariate Wasserstein alignment and preserves scale-normalized class semantics through confidence-filtered angular alignment, with radial magnitude modulating pseudo-label reliability. DisRFM introduces topology-conditioned polar flow matching, which couples class-compatible source and target samples by a normalized polar transport cost and learns a metric-corrected vector field along geodesic interpolants. Theoretical analysis characterizes the structural risk of unconditional domain confusion and relates polar discrepancies and flow error to target risk. Extensive experiments under diverse domain shifts demonstrate that DisRFM consistently outperforms state-of-the-art methods.

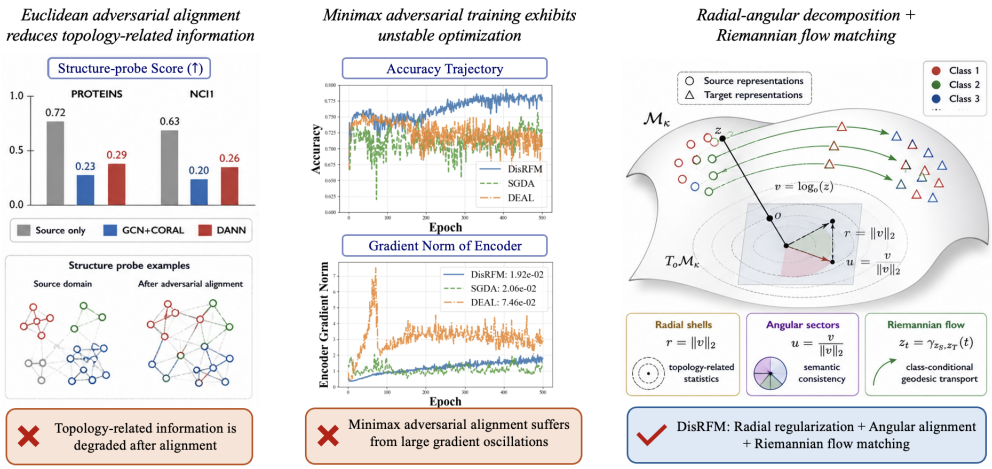
1 Introduction

Graph Domain Adaptation (GDA) aims to transfer a graph classifier trained on a labeled source domain to an unlabeled target domain under distribution shift [52]. Unlike standard feature-domain adaptation, graph shifts are often simultaneously semantic and topological: the decision boundary should remain transferable, while graph size, density, degree profiles, structural roles, motifs, or community patterns may vary across domains [20]. A reliable GDA model should therefore align task-relevant semantics without erasing structure-sensitive information that is useful for target prediction.

Existing GDA methods align Euclidean latent spaces by making source and target embeddings domain-indistinguishable through an adversarial objective [44, 41, 17, 15]. This paradigm is effective

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) Structural Degeneration (b) Optimization Instability (c) Riemannian Polar Flow (DisRFM)

Figure 1: Motivation of DisRFM. (a) Euclidean adversarial GDA erodes topology information, evidenced by lower post-alignment structure-probe scores. (b) Minimax alignment yields unstable training with gradient-norm oscillations. (c) DisRFM imposes a Riemannian polar bias: radial shells act as compact topology-related proxies, angular sectors preserve semantic consistency, and flow matching transports source embeddings toward targets along class-conditioned geodesics.

under moderate covariate shifts, but becomes fragile when domain variables are entangled with graph structure. First, Euclidean alignment can induce ***Structural Degeneration***. Since Euclidean coordinates conflate radial scale and direction, structural roles and class-discriminative cues may share coordinates; domain confusion can remove topology-sensitive variation alongside domain variation [5, 35, 16, 39, 6]. Figure 1(a) shows degraded structure-probe scores. Second, adversarial GDA optimizes a minimax game between the encoder and domain discriminator [19], causing ***Optimization Instability***. Under large structural shifts, it may yield oscillatory or poorly conditioned gradients [1, 33, 40], as shown in Figure 1(b). These limitations suggest that graph domain adaptation should avoid unconditional domain confusion. Instead, it should preserve label-relevant structure while transporting samples across semantically and topologically compatible regions.

We address this problem from a representation-geometry and transport viewpoint. Graph domain shifts often entangle label-relevant structural variation with domain-specific nuisance factors. As a result, enforcing Euclidean domain confusion may inadvertently suppress structural cues useful for target prediction. Curved manifolds offer a suitable inductive bias for hierarchy-rich and non-Euclidean graph structures beyond isotropic Euclidean spaces [35, 49, 6]. In a constant-curvature manifold, geodesic polar coordinates provide a natural view of embeddings through a scale-sensitive radial component and a scale-normalized angular component. This does not imply an identifiable separation between topology and semantics; rather, it suggests a useful regularization bias for graph transfer. The resulting challenge is to align domains without collapsing all domain-correlated structure or treating radial, angular, and transport effects as independent heuristics. This motivates topology-conditioned polar transport: source and target graphs should be matched according to class semantics and graph structure, and their alignment should follow the geometry of the representation space.

To this end, we propose DisRFM, a class-aware transport framework over constant-curvature manifolds. Polar geometry provides an operational coordinate bias, not an identifiable factorization: radius serves as a topology-related scale proxy, while angle offers a scale-normalized signal for semantic alignment. Radial structural calibration matches topology-sensitive metric scales, keeping comparable graphs at compatible radii without forcing structure into radius. Confidence-filtered angular alignment matches class prototypes on scale-normalized directions, preserving margins while limiting confirmation noise. Class-conditional Riemannian flow matching replaces discriminator-based alignment with velocity regression along pseudo-label coupled geodesics, transporting representations toward target support while preserving radius–angle coupling. Thus, DisRFM replaces domain-level indistinguishability with geometry-constrained, class-conditional manifold alignment.

Our main contributions are summarized as follows: (1) We identify a structural failure mode of domain-confusion GDA and formalize, through a simple information and total-variation argument,

why enforcing domain invariance can remove label-relevant graph structure when structure and domain are correlated. (2) We propose DisRFM, which integrates coordinate-wise radial/angular regularization in Riemannian polar coordinates with geometry-aware flow matching for stable graph domain adaptation. (3) Extensive experiments on benchmark datasets demonstrate that DisRFM outperforms state-of-the-art baselines and exhibits strong optimization stability.

2 Related Work

Domain Adaptation (DA). DA transfers knowledge from a labeled source domain to an unlabeled target domain by reducing distributional shifts while preserving task-relevant semantics [43, 55]. With the growing use of graph data, this paradigm has extended to Graph Domain Adaptation (GDA), where both node attributes and graph structures vary across domains [53, 59]. Most methods learn graph representations with GNNs and apply alignment techniques, such as adversarial learning, MMD minimization, or optimal transport, to obtain domain-invariant features [56, 57, 30]. However, they typically align graphs in Euclidean space and overlook non-Euclidean geometry, which often causes distorted representations and negative transfer under significant structural shifts [58, 13, 9]. To address this, we propose DisRFM for geometry-aware alignment in non-Euclidean space.

Manifold Learning on Graphs. Manifold learning assumes that high-dimensional data reside on low-dimensional manifolds, capturing geometric properties that Euclidean methods often distort [26, 60, 32]. Recent advances extend this paradigm to non-Euclidean spaces, integrating graph topology to better encode hierarchical and relational structures [47, 45, 14]. However, existing manifold-based methods mainly focus on single-domain tasks, overlooking domain adaptation [4, 46, 51]. Crucially, naive applications of general adaptation techniques to manifold spaces are suboptimal, as they fail to model cross-domain distributional shifts while respecting the underlying geometry [8, 18, 10]. To bridge this gap, we propose DisRFM, which formulates graph domain adaptation as a continuous, geometry-aware transformation on a non-Euclidean manifold via Riemannian flow matching.

3 Methodology

Overview. We propose DisRFM, a two-component GDA framework: (1) **Polar Endpoint Regularization** mitigates structural degeneration using radius as a topology-related scale proxy and angle as a scale-normalized class-alignment signal. (2) **Topology-Conditioned Polar Flow Matching** mitigates indiscriminate alignment by transporting source and target graphs through class- and structure-compatible regions, replacing adversarial domain confusion with geometry-aware transport supervision. Together, they preserve label-relevant structure, reduce over-alignment, and improve transfer under structural shifts without assuming globally identifiable topology–semantics separation.

3.1 Polar Endpoint Regularization

Existing GDA objectives usually align Euclidean embeddings with isotropic losses, which do not distinguish scale-sensitive and direction-sensitive variations and may therefore over-align graph representations under structural shift [35, 5]. We instead use polar coordinates induced by a fixed origin on a constant-curvature manifold. On a geodesic normal ball, the metric has the block form

$$ds^2 = dr^2 + S_c(r)^2 d\Omega^2, \qquad S_c(r) = \begin{cases} \sin(\sqrt{c}r)/\sqrt{c}, & c > 0, \\ r, & c = 0, \\ \sinh(\sqrt{-c}r)/\sqrt{-c}, & c < 0, \end{cases} \tag{1}$$

where $d\Omega^2$ is the standard spherical metric. This geometry motivates separate endpoint roles for radius and angle while preserving their metric coupling through the warping factor $S_c(r)^2$. Throughout the theoretical statements, r denotes geodesic radius. In implementation, r is computed from $\mathbf{v} = \mathrm{Log}_o^c(\mathbf{z})$ in the origin tangent coordinates; the constant conformal factor at the origin only rescales all radii and is absorbed by the radial loss weight.

Radial Structural Calibration. First, we obtain the graph-level representation $\mathbf{z} \in \mathcal{M}_c$ via [28]

$$\mathbf{m}_j^{(l)} = \mathbf{W}^{(l)} \otimes_c \mathbf{h}_j^{(l)} = \mathrm{Exp}_{\mathbf{0}}^c\left(\mathbf{W}^{(l)} \cdot \mathrm{Log}_{\mathbf{0}}^c(\mathbf{h}_j^{(l)})\right), \ \mathbf{h}_i^{(l+1)} = \mathrm{Exp}_{\mathbf{0}}^c\left(\sigma\Big(\sum_{j \in \mathcal{N}(i) \cup \{i\}} \alpha_{ij} \cdot \mathrm{Log}_{\mathbf{0}}^c(\mathbf{m}_j^{(l)})\Big)\right),$$

where the manifold curvature c controls the underlying geometry. We map a graph representation $z \in \mathcal{M}_c$ to the tangent space at the origin by $v = \mathrm{Log}_0^c(z) \in T_0\mathcal{M}_c$ and use the polar decomposition

$$r = \|\mathbf{v}\|_{g_o}, \quad \mathbf{u} = \mathbf{v}/\|\mathbf{v}\|_2,$$

where the radius r acts as a structure-related scalar proxy and $\mathbf{u}$ records normalized tangent direction. Rather than assuming all structure lies in r, we use radial magnitude as a compact topology-related summary and calibrate the target radial profile to the source. We formulate this as marginal matching and minimize the univariate Wasserstein distance [11, 36], computed by sorting. For batch size B, let $\mathcal{R}_S = \mathrm{sort}(\{r_S^{(i)}\}_{i=1}^B)$ and $\mathcal{R}_T = \mathrm{sort}(\{r_T^{(i)}\}_{i=1}^B)$ denote radial order statistics, and define:

$$\mathcal{L}_{rad} = \frac{1}{B}\sum_{k=1}^{B} |\mathcal{R}_S[k] - \mathcal{R}_T[k]| \,. \tag{2}$$

Minimizing $\mathcal{L}_{rad}$ aligns the radial statistics of the target with those of the source. This regularizer reduces the risk of structural collapse and preserves topology-related diversity during adaptation.

Angular Semantic Alignment. Let $\mathbf{W} \in \mathbb{R}^{K\times d}$ denote the source classifier weights, and let $\mathbf{w}_k$ be a prototype for class k. For a target graph with tangent representation $\mathbf{v}_T^{(i)}$, we define cosine similarities

$$s_{i,k} = \frac{\mathbf{v}_T^{(i)}}{\|\mathbf{v}_T^{(i)}\|_2} \cdot \left(\frac{\mathbf{w}_k}{\|\mathbf{w}_k\|_2}\right)^\top,$$

so semantic matching depends on direction rather than scale. Since target ground-truth labels are unavailable, we rely on the source classifier to generate pseudo-labels [25]. Recognizing that indiscriminate usage of noisy pseudo-labels risks distorting the manifold geometry [61, 7], we use pseudo-labels $\hat{y}_i = \arg\max_k p(k|v_T^{(i)})$ but filter uncertain targets with:

$$M_i = \mathbb{I}\left(\max_k p(k|\mathbf{v}_T^{(i)}) > \zeta\right),$$

where $p(\cdot \mid \mathbf{v}_T^{(i)}) = \mathrm{Softmax}(\mathbf{W}\mathbf{v}_T^{(i)} + \mathbf{b})$ denotes class probability, and $\mathbb{I}(\cdot)$ is the indicator function. This mask M_i retains samples with confidence exceeding ζ, filtering out uncertain predictions. The confidence score is computed from the full tangent representation but only used to screen pseudo-labels; angular similarity remains scale-normalized. Because large-radius target samples often lie farther from the shared support under structural shift, we use radius only as a soft reliability prior for pseudo-label supervision, not as a reason to discard complex graphs. Specifically, we set $\alpha_i = \exp(-\tau r_T^{(i)})$, with $\tau = 1$ after radial normalization. The angular loss is formulated as:

$$\mathcal{L}_{ang} = \frac{\sum_{i=1}^{B} M_i \cdot \alpha_i \cdot \ell_{ce}(\gamma \cdot \mathbf{s}_i, \hat{y}_i)}{\sum_{i=1}^{B} M_i \cdot \alpha_i + \epsilon}, \tag{3}$$

where ℓ_{ce} is the Cross-Entropy loss, $\mathbf{s}_i$ is the vector of cosine similarities, and γ is a temperature scaling factor to sharpen the predictions. By minimizing $\mathcal{L}_{ang}$, we encourage semantic consistency in the angular domain while allowing the radial magnitude to modulate the reliability of pseudo-label supervision. In our formulation, angle acts as the primary carrier of semantic alignment, whereas radius plays an auxiliary reliability-regulation role rather than a hard semantic exclusion role.

3.2 Topology-Conditioned Polar Flow Matching

The transport component uses one objective, not separate coupling and flow losses. Polar coupling determines admissible source–target pairs, while flow matching supervises transport geometry. It is not an additional regularizer; it is a stop-gradient sampling measure for the flow matching loss.

Stopped Polar Coupling. For each class y, define source and reliable target index sets $\mathcal{I}_y^S = \{i : y_i^S = y\}$, $\mathcal{I}_y^T = \{j : M_j = 1, \hat{y}_j = y\}$, and let $\rho_j = \max_k p(k|\mathbf{v}_T^{(j)})$ denote target confidence. For $i \in \mathcal{I}_y^S$ and $j \in \mathcal{I}_y^T$, we define the normalized polar transport cost

$$C_{ij}^{(y)} = \widetilde{d}_{\mathcal{M}_c}(z_i^S, z_j^T)^2 + |\widetilde{r}_i^S - \widetilde{r}_j^T|^2 + \widetilde{d}_{\mathbb{S}}(u_i^S, u_j^T)^2 - \log(\rho_j + \epsilon),$$

where $\widetilde{d}_{\mathcal{M}_c}$ and $\widetilde{d}_{\mathbb{S}}$ are z-score-normalized pairwise distances within the class minibatch, and $\widetilde{r}$ is the z-score-normalized radius over pooled source–target samples of that class. For raw component A, we use $\widetilde{A}_{ij} = \frac{A_{ij} - \mathrm{sg}(\mu_A)}{\mathrm{sg}(\sigma_A) + \epsilon}$, where μ_A, σ_A are minibatch statistics and $\mathrm{sg}(\cdot)$ stops gradients through them. The geometric, radial, angular, and confidence terms respectively enforce locality, match structure-sensitive scale, preserve class-compatible directions, and downweight unreliable pseudo-labels. All normalized components use unit weights, avoiding cost-balancing hyperparameters.

Let μ_y be uniform over $\mathcal{I}_y^S$ and let $\nu_y(j) = \rho_j / \sum_{l \in \mathcal{I}_y^T} \rho_l$. If either set is empty, class y is skipped for that minibatch. Otherwise, we compute an entropic transport plan

$$\pi_y^\star = \arg\min_{\pi \in \Pi(\mu_y, \nu_y)} \langle \pi, C^{(y)} \rangle + \varepsilon_{\mathrm{ot}} \sum_{i,j} \pi_{ij} \big(\log \pi_{ij} - 1 \big),$$

where $\Pi(\mu_y, \nu_y)$ is the set of couplings with marginals μ_y and ν_y. The minibatch coupling is $\pi_B = \mathrm{Normalize}(\sum_y \pi_y^\star)$. Here $\mathrm{sg}(\cdot)$ denotes stop-gradient: it is the identity in the forward pass and blocks backward gradients. We use $\mathrm{sg}(\pi_B)$ in the loss to prevent unstable Sinkhorn gradients.

Metric-Corrected Polar Flow Regression. For a coupled pair $(\mathbf{z}_i^S, \mathbf{z}_j^T) \sim \mathrm{sg}(\pi_B)$, define the constant-speed geodesic interpolant

$$\mathbf{z}_t = \gamma_{ij}(t) = \mathrm{Exp}^c_{\mathbf{z}_i^S} \Big(t \, \mathrm{Log}^c_{\mathbf{z}_i^S}(\mathbf{z}_j^T) \Big), \qquad t \sim \mathcal{U}[0,1].$$

Let ξ_t be its analytic velocity, $\xi_t = P^c_{\mathbf{z}_i^S \to \mathbf{z}_t} \Big(\mathrm{Log}^c_{\mathbf{z}_i^S}(\mathbf{z}_j^T) \Big)$, where $P^c_{x \to y}$ denotes parallel transport. Writing $\mathbf{z}_t$ in polar coordinates as (r_t, u_t) with $u_t = u(\mathbf{z}_t)$, the radial and angular velocities are:

$$\dot{r}_t = dr_{\mathbf{z}_t}[\xi_t], \qquad \dot{u}_t = du_{\mathbf{z}_t}[\xi_t] \in T_{u_t}\mathbb{S}^{d-1}.$$

We parameterize the learned vector field in polar form,

$$\mathbf{v}_\psi(z_t, t) = a_\psi(\mathbf{z}_t, t)\partial_r + b_\psi(\mathbf{z}_t, t), \qquad b_\psi(\mathbf{z}_t, t) \in T_{u_t}\mathbb{S}^{d-1},$$

where a_ψ predicts radial speed and b_ψ predicts angular speed. The unified topology-conditioned polar flow matching loss is

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{t \sim \mathcal{U}[0,1],\, (\mathbf{z}_i^S, \mathbf{z}_j^T) \sim \mathrm{sg}(\pi_B)} \Big[|a_\psi(\mathbf{z}_t, t) - \dot{r}_t|^2 + S_c(r_t)^2 \|b_\psi(\mathbf{z}_t, t) - \dot{u}_t\|_{\mathbb{S}}^2 \Big]. \tag{4}$$

The factor $S_c(r_t)^2$ is the angular block of the polar metric in Eq. (1). Therefore, DisRFM learns one manifold flow whose velocity error is decomposed and measured by the correct curvature-dependent geometry. This distinguishes our objective from generic Euclidean or Riemannian flow matching, while keeping the coupling and flow supervision in a single loss.

3.3 Learning Framework

DisRFM separates endpoint shaping from trajectory learning. The radial and angular losses regularize endpoint distributions in polar coordinates, while flow matching learns a single coupled manifold transport. This distinction is important: independent radial and angular flows would ignore the metric coupling and can produce distorted, non-geodesic trajectories.

Formally, DisRFM jointly optimizes the source task capability, geometric structural integrity, and the manifold transport trajectory. The objective function is formulated as:

$$\mathcal{L}_{total} = \mathcal{L}_{task} + \lambda_1 \mathcal{L}_{rad} + \lambda_2 \mathcal{L}_{ang} + \lambda_3 \mathcal{L}_{FM}, \tag{5}$$

where λ_1, λ_2, and λ_3 are hyperparameters balancing the contributions of radial regularization, angular regularization, and manifold transport, respectively. We note that $\mathcal{L}_{task}$ is computed from the full tangent representation, so the radial-angular decomposition functions as a regularization bias rather than a hard restriction on all prediction signals. Specifically, $\mathcal{L}_{task}$ is the standard cross-entropy loss computed on the labeled source domain $\mathcal{D}_S$ to ensure discriminative capability:

$$\mathcal{L}_{task} = \mathbb{E}_{(\mathcal{G}_S, y_S) \sim \mathcal{D}_S} \left[-\log p(y_S | \mathbf{v}_S) \right], \quad \mathbf{v}_S = \mathrm{Log}^c_o(f_\theta(\mathcal{G}_S)),$$

where $p(y|\mathbf{v}_S) = \mathrm{Softmax}(\mathbf{W}\mathbf{v}_S + \mathbf{b})$ denotes the predicted probability in the tangent space.

3.4 Theoretical Analysis

Our analysis is conditional: it identifies when topology-conditioned polar transport is beneficial, rather than claiming universal topology-semantics identifiability or global convergence of nonconvex training. Let $\mathbf{z} = f_\theta(G) \in \mathcal{M}_c$ be the Riemannian representation of G, and write $\mathbf{z} = (r, u)$ in polar coordinates around o. Let $a(G) \in \mathbb{R}^m$ denote topology-related graph statistics and y the class label.

Theorem 1 (A Structural Failure Mode of Domain Confusion) *Consider a binary structural variable $A \in \{0, 1\}$ with equal priors $\Pr(A = 0) = \Pr(A = 1) = 1/2$, label $Y = A$, and domain variable $D \in \{0, 1\}$ with $\Pr(D = A) = 1 - \eta$ for some $0 \le \eta < 1/2$. Assume a representation Z depends on the graph only through A in this toy model. If an adversarial objective enforces approximate domain invariance*

$$\mathrm{TV}\big(P_{Z|D=0}, P_{Z|D=1}\big) \le \delta,$$

then the total variation between the label-conditionals of Z is bounded by

$$\mathrm{TV}\big(P_{Z|Y=0}, P_{Z|Y=1}\big) \le \frac{\delta}{1 - 2\eta}.$$

Consequently, any classifier using Z has error at least

$$\inf_h \Pr(h(Z) \neq Y) \ge \frac{1}{2}\left(1 - \frac{\delta}{1 - 2\eta}\right).$$

Theorem 1 formalizes structural degeneration. If label-relevant structure correlates with the domain, forcing source and target representations to be indistinguishable can make them uninformative about the label. DisRFM avoids this failure mode by not imposing unconditional domain confusion. Instead, it aligns class-conditioned source and target distributions through topology-aware polar transport.

Definition 1 (Operational Polar Representation) *A representation $\mathbf{z} = f_\theta(G)$ is $(\epsilon_s, \epsilon_y, \epsilon_g)$ operationally polar-disentangled for a graph domain adaptation task if the following conditions hold:*

- *(i) Structure sufficiency:* $\inf_{q_s:\mathbb{R}_+\to\mathbb{R}^m} \mathbb{E}\big[\|a(G) - q_s(r(G))\|\big] \le \epsilon_s$,
- *(ii) Semantic sufficiency:* $\inf_{q_y:\mathbb{S}^{d-1}\to\Delta^{K-1}} \mathbb{E}\big[\ell_{\mathrm{ce}}(q_y(u(G)), y)\big] \le \epsilon_y$,
- *(iii) Low first-order interference:* $\mathrm{GC}(\mathcal{L}_{\mathrm{rad}}, \mathcal{L}_{\mathrm{ang}}) := \frac{\left|\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{ang}} \rangle_g\right|}{\|\nabla_g \mathcal{L}_{\mathrm{rad}}\|_g \|\nabla_g \mathcal{L}_{\mathrm{ang}}\|_g + \delta} \le \epsilon_g$,

for a small numerical constant $\delta > 0$.

Definition 1 formalizes task-level polar disentanglement. Conditions *(i)* and *(ii)* assign complementary coordinate roles, while condition *(iii)* bounds normalized Riemannian first-order interaction between radial and angular endpoint objectives. The definition captures the coordinate-wise bias induced by our regularizers without assuming a globally identifiable topology–semantics factorization.

Theorem 2 (First-Order Orthogonality of Polar Endpoint Losses) *Let $\mathcal{M}_c$ be a d-dimensional constant-curvature manifold and let $o \in \mathcal{M}_c$ be the origin. Work on a geodesic normal ball $B_o(\rho)$, with $\rho < \pi/\sqrt{c}$ when $c > 0$, and exclude o. For $\mathbf{z} = (r, u) \in B_o(\rho) \setminus \{o\}$, the polar metric is Eq. (1). For a batch $\mathbf{z} = (\mathbf{z}_i)_{i=1}^B$, equip $\mathcal{M}_c^B$ with the product metric. Let $\mathcal{L}_{\mathrm{rad}}(\mathbf{z}) = \ell_r(r_{1:B})$ and $\mathcal{L}_{\mathrm{ang}}(\mathbf{z}) = \ell_u(u_{1:B})$ be almost-everywhere differentiable radial-only and angular-only endpoint losses. Then, wherever the gradients exist,*

$$\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{ang}} \rangle_g = 0.$$

If a differentiable radial reliability function weights the angular loss $\mathcal{L}_{\mathrm{ang}}^w = \sum_i \alpha(r_i)\ell_i(u_i)$, then

$$\left|\left\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{ang}}^w \right\rangle_g\right| \le \sum_{i=1}^B |\partial_{r_i}\ell_r|\,|\alpha'(r_i)|\,|\ell_i(u_i)|.$$

For the choice $\alpha(r) = e^{-r}$, the coupling term is bounded by $\sum_i |\partial_{r_i}\ell_r| e^{-r_i} |\ell_i(u_i)|$.

Theorem 2 justifies endpoint regularization: pure radial and pure angular endpoint losses are first-order orthogonal under the polar metric. The practical radius-weighted angular loss is not exactly orthogonal, but the reliability-weight derivative explicitly bounds its additional first-order interaction.

Theorem 3 (Radial Structural Proxy Bound) *Let P_S and P_T be source and target graph distributions. For a graph G, let $r(G) = \|\operatorname{Log}_o^c(f_\theta(G))\|_{g_o}$ and let $a(G) \in \mathbb{R}^m$ denote topology-related graph statistics. Denote by $r_\# P$ and $a_\# P$ the pushforward distributions induced by $r(G)$ and $a(G)$. Suppose there exists an L_a-Lipschitz map $h : \mathbb{R}_+ \to \mathbb{R}^m$ such that, for $D \in \{S, T\}$, $\mathbb{E}_{G\sim P_D}[\|a(G) - h(r(G))\|] \le \epsilon_D$. Then*

$$W_1(a_\# P_S, a_\# P_T) \le L_a W_1(r_\# P_S, r_\# P_T) + \epsilon_S + \epsilon_T.$$

For equal-size empirical distributions, $W_1(\widehat{P}_S^r, \widehat{P}_T^r) = \frac{1}{B}\sum_{k=1}^B |\mathcal{R}_S[k] - \mathcal{R}_T[k]|$, which is Eq. (2)*.*

Theorem 3 gives a conditional, testable guarantee: when radius predicts the chosen structural statistics with small error, radial calibration controls the corresponding structural discrepancy. It does not claim that radius automatically captures every graph-structural factor.

Theorem 4 (Angular Margin Preservation) *Let $\{\hat{w}_c\}_{c=1}^K \subset \mathbb{S}^{d-1}$ be normalized class prototypes, and define $h(u) = \arg\max_{c\in[K]} \hat{w}_c^\top u$. For a labeled sample $(\mathbf{z}, y)$, let $u = u(\mathbf{z})$ and let $\tilde{u}$ be the angular coordinate after an endpoint transport map. Define the angular margin*

$$m_y(u) = \hat{w}_y^\top u - \max_{c\neq y} \hat{w}_c^\top u.$$

If $m_y(u) \ge \mu$ and $d_{\mathbb{S}}(u, \tilde{u}) < \frac{\mu}{2}$ for some $\mu > 0$, then $h(\tilde{u}) = y$. For a labeled distribution P,

$$R_{\mathrm{post}}(h) \le R_{\mathrm{pre}}(h) + \rho_\mu + \frac{2}{\mu}\Delta_{\mathrm{ang}},$$

where $R_{\mathrm{pre}}(h) = \Pr_P[h(u) \neq y]$, $R_{\mathrm{post}}(h) = \Pr_P[h(\tilde{u}) \neq y]$, $\rho_\mu = \Pr_P[0 \le m_y(u) < \mu]$, and $\Delta_{\mathrm{ang}} = \mathbb{E}_P[d_{\mathbb{S}}(u, \tilde{u})]$.

Theorem 4 explains why class-conditional angular alignment is preferable to unconditional domain alignment: labels are preserved when angular drift is small relative to the source prototype margin.

Theorem 5 (Polar Flow Endpoint Stability) *Fix a coupling π and assume all analytic geodesics and learned flow trajectories stay inside a compact geodesic normal ball $B_o(\rho)$. Let $\gamma_t(z_S, z_T)$ be the analytic geodesic and let φ_t^ψ be the trajectory generated by the learned polar field in Eq.* (4) *with $\varphi_0^\psi = z_S$. Assume the reconstructed manifold vector field is L_v-Lipschitz in z on $B_o(\rho)$. Then there exists a constant C_{geo} depending only on the ball and the metric such that*

$$\mathbb{E}_{(z_S,z_T)\sim\pi}\left[d_{\mathcal{M}_c}(\varphi_1^\psi, z_T)\right] \le C_{\mathrm{geo}} e^{L_v}\sqrt{\mathcal{L}_{\mathrm{FM}}}.$$

This result follows from a Riemannian Gronwall argument: endpoint deviation is controlled by time-integrated metric-corrected velocity-regression error under a fixed population coupling. It does not imply global optimal transport or convergence of nonconvex training; minibatch Sinkhorn couplings add pseudo-label, coupling-estimation, finite-sample, and numerical-integration errors.

Theorem 6 (Target-Risk Bound with Polar Discrepancy) *Let P_S and P_T be source and target distributions over graph-label pairs. Assume $\mathbf{z} = f_\theta(G) \in B_o(\rho) \subset \mathcal{M}_c$ and the loss $\ell_\phi(\mathbf{z}, y) = \ell(g_\phi(\mathbf{z}), y) \in [0, 1]$ is L_ℓ-Lipschitz in z. Let Φ_ψ be the endpoint map induced by the learned polar flow and let Ψ be an ideal class-conditioned transport map. For each class y, let Γ_y be a lifted coupling over source–target graph pairs (G, G') such that the induced representation pair $\mathbf{z} = \Psi(f_\theta(G))$ and $\mathbf{z}' = f_\theta(G')$ has marginals $\Psi_\# P_S(\mathbf{z} \mid y)$ and $P_T(\mathbf{z} \mid y)$, respectively. Define*

$$D_{\mathrm{rad}}^\Gamma = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y}|r(\Psi(f_\theta(G))) - r(f_\theta(G'))|,\ D_{\mathrm{top}}^\Gamma = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y}\|q(G) - q(G')\|_2,$$

$$D_{\mathrm{ang}}^\Gamma = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y} d_{\mathbb{S}}(u(\Psi(f_\theta(G))), u(f_\theta(G'))),$$

Table 1: Graph classification results (in %) under node and edge density domain shifts on the Mutagenicity dataset, and feature domain shifts on DD, PROTEINS, BZR, BZR_MD, COX2, and COX2_MD. For convenience, PROTEINS, DD, COX2, COX2_MD, BZR, and BZR_MD are abbreviated as P, D, C, CM, B, and BM, respectively. **Bold** results indicate the best performance.

Methods	**Node Shift**			**Edge Shift**			**Feature Shift**					
	M0→M1	M0→M2	M0→M3	M0→M1	M0→M2	M0→M3	P→D	D→P	C→CM	CM→C	B→BM	BM→B
WL subtree	34.3	40.4	52.7	34.4	47.6	52.7	43.0	42.2	53.1	58.2	51.3	44.0
GCN	64.1±1.4	65.5±2.0	56.9±2.1	66.3±1.7	63.6±1.4	56.0±1.4	48.9±2.0	60.9±2.3	51.2±1.8	66.9±1.8	48.7±2.0	78.8±1.7
GIN	66.5±2.1	52.0±1.7	53.7±1.7	67.1±1.7	54.2±2.6	55.4±1.9	57.3±2.2	61.9±1.9	53.8±2.5	55.6±2.0	49.9±2.4	79.2±2.8
GMT	65.7±1.8	62.1±2.1	59.0±2.0	67.9±1.3	61.5±1.8	58.2±2.4	59.5±2.5	50.7±2.2	49.3±1.8	58.2±2.0	50.2±2.3	74.4±1.8
CIN	65.1±1.7	66.0±1.7	55.2±1.5	66.3±1.8	60.8±1.7	55.8±2.4	59.1±2.6	58.0±2.7	51.2±2.0	55.6±1.5	49.2±1.4	74.2±1.9
PathNN	70.2±1.5	67.1±2.0	58.0±1.9	68.9±1.9	62.9±1.7	58.1±1.6	57.9±1.8	53.8±3.3	49.8±1.7	66.9±2.5	50.3±2.3	75.3±2.2
dDGM	79.1±0.4	70.3±0.5	63.9±0.3	76.5±1.3	69.0±1.3	67.0±0.4	59.0±1.8	63.3±1.5	58.4±2.7	74.3±1.2	53.6±1.6	79.2±1.5
RieGrace	78.5±1.3	70.6±1.4	62.8±1.3	76.4±0.5	68.9±0.6	66.1±1.2	59.8±1.4	63.8±1.3	56.8±2.1	66.6±1.4	52.2±2.3	78.9±2.0
ProGDM	75.2±1.2	69.0±1.1	55.7±1.4	74.6±1.2	67.5±1.2	55.5±0.7	58.7±2.0	59.6±2.3	54.2±1.7	75.2±2.0	52.6±1.3	75.4±2.9
D-GCN	75.2±0.5	67.9±0.4	56.3±1.1	73.8±1.2	66.8±1.4	56.7±1.0	62.4±1.9	59.7±2.1	53.7±1.8	77.6±1.9	51.7±1.6	75.7±2.0
DEAL	77.1±0.9	70.9±0.9	60.3±1.1	76.6±1.6	70.6±1.2	60.2±2.1	61.7±2.0	60.0±1.5	52.7±2.7	69.4±2.9	52.4±2.9	78.6±1.4
SGDA	77.5±0.6	69.7±0.5	65.5±0.8	75.9±1.6	68.9±0.8	64.4±0.4	48.3±2.0	55.8±2.6	49.8±1.8	66.9±2.3	50.3±2.1	78.8±2.6
A2GNN	73.5±1.9	66.1±1.5	60.4±1.1	69.5±1.4	68.6±1.4	58.8±2.2	57.8±2.1	60.3±1.5	51.5±1.8	67.7±2.1	51.6±2.3	77.5±1.9
StruRW	78.3±1.3	69.7±1.3	62.6±0.7	76.1±1.5	69.0±1.3	62.1±1.0	59.1±2.3	58.8±2.8	51.2±2.0	54.8±2.9	49.2±1.4	74.7±2.1
PA-BOTH	69.8±1.5	63.8±1.9	55.3±1.1	74.7±1.1	65.3±1.3	52.2±1.5	54.2±3.2	56.7±2.6	52.9±2.8	61.8±2.0	47.5±3.0	78.8±1.9
GAA	79.3±1.2	71.2±0.7	65.6±1.3	77.5±1.2	70.0±1.2	66.5±1.3	62.4±0.6	64.1±0.8	59.4±1.8	78.4±1.2	57.2±3.3	78.5±3.0
TDSS	63.6±1.3	56.7±1.6	54.7±1.0	71.6±1.5	67.3±1.0	55.4±1.6	61.9±1.1	63.6±1.9	56.8±1.3	77.0±2.6	56.6±1.1	79.1±2.4
GOTDA	77.3±1.3	69.9±1.7	65.3±1.2	76.1±1.2	68.5±0.8	65.8±0.6	62.1±1.5	63.2±1.7	58.5±2.1	78.2±1.1	57.0±1.7	78.9±2.6
MASH	76.1±1.3	66.9±1.3	54.3±1.2	74.4±1.2	66.1±0.5	52.2±1.2	61.8±2.1	64.2±2.4	56.2±2.0	77.2±2.1	55.2±2.0	78.1±2.3
GeoAdapt	76.9±1.5	68.7±1.4	60.1±1.3	75.5±1.2	67.5±1.2	60.5±0.6	63.1±1.2	64.0±1.4	56.7±1.7	78.3±2.1	57.5±2.7	78.5±2.2
DisRFM-$\mathbb{E}^n$	79.4±0.7	71.0±1.0	65.2±0.2	77.6±0.4	70.2±1.2	66.1±0.3	61.9±1.0	63.0±2.0	56.5±1.9	78.2±1.0	56.8±2.4	78.9±1.3
DisRFM-$\mathbb{S}^n$	79.7±0.7	**72.0±1.0**	65.9±0.9	78.0±0.5	71.0±0.6	**68.2±1.0**	**63.8±1.2**	64.0±2.7	**59.7±1.4**	78.4±2.1	57.5±1.8	79.1±1.0
DisRFM-$\mathbb{H}^n$	**79.8±0.8**	71.7±0.7	**66.7±0.8**	**78.3±0.7**	**71.3±0.4**	67.7±0.9	63.3±1.3	**64.4±1.6**	58.3±2.2	**79.0±2.5**	**58.0±1.9**	**79.4±1.7**

where $\omega_y = P_T(Y = y)$. *Let* $C_\rho = \sup_{0 \le t \le \rho} S_c(t)$ *and let* $\lambda_{\mathrm{lab}} = \frac{1}{2} \sum_y |P_S(Y = y) - P_T(Y = y)|$. *If pseudo-labels used to construct the target class-conditionals have error* ϵ_{pl}, *then there exist constants* C_{top}, C_{FM}, *and* C_{pl} *such that*

$$R_T(g_\phi) \le R_S(g_\phi \circ \Phi_\psi) + L_\ell \Big(D^\Gamma_{\mathrm{rad}} + C_\rho D^\Gamma_{\mathrm{ang}} + C_{\mathrm{top}} D^\Gamma_{\mathrm{top}} + C_{\mathrm{FM}} \sqrt{\mathcal{L}_{\mathrm{FM}}} \Big) + C_{\mathrm{pl}} \epsilon_{\mathrm{pl}} + \lambda_{\mathrm{lab}}.$$

Theorem 6 shows that the target risk is controlled by the transported-source risk, radial discrepancy, angular class-conditional discrepancy, topology-conditioned transport discrepancy, flow endpoint error, pseudo-label noise, and label-prior mismatch. This is a conditional population bound: the empirical objective in Eq. (5) is a tractable surrogate, while minibatch coupling, pseudo-labeling, and finite-sample effects remain approximation terms rather than being hidden in the notation.

4 Experiments

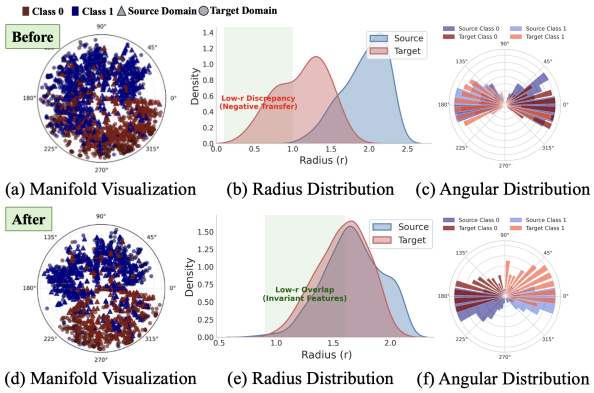
4.1 Experimental Settings

Dataset. We evaluate DisRFM on two types of domain shift scenarios: (1) structure-based domain shifts: we partition the PROTEINS, NCI1, Mutagenicity [12], and ogbg-molhiv [21] datasets based on node and edge densities, following the protocol described in [58]; (2) feature-based domain shifts: we evaluate DisRFM on DD, PROTEINS, BZR, BZR_MD, COX2, and COX2_MD [48] datasets. More details of experimental datasets are shown in Appendix H.

Baselines. We compare DisRFM with competitive baselines on the above datasets, including two graph kernel methods: WL subtree [42] and PathNN [34]; four general graph neural networks (GNNs): GCN [24], GIN [54], CIN [3] and GMT [2]; four manifold-based GNNs: dDGM [4], RieGrace [46], ProGDM [51], and D-GCN [45]; seven graph domain adaptation (GDA): DEAL [56], SGDA [37], StruRW [29], A2GNN [27], PA-BOTH [30], GAA [13], and TDSS [9]; and three manifold-based GDA: GOTDA [31], MASH [38], and GeoAdapt [18]. More settings are introduced in Appendix I.

4.2 Results Analysis

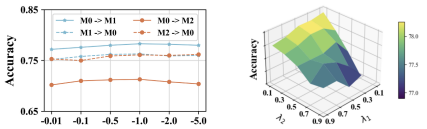
Performance Comparison. We evaluate DisRFM against all baselines under diverse domain shifts. Results are reported in Tables 1 and 17–24. We observe: First, manifold-based GNNs outperform Euclidean GNNs, suggesting that non-Euclidean geometry provides an inductive bias for graphs with hierarchical or relational structure. However, manifold representation learning alone is insufficient, as manifold-based adaptation baselines lag behind DisRFM in many settings. Second, DisRFM achieves



(a) Manifold Visualization (b) Radius Distribution (c) Angular Distribution

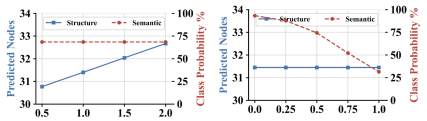
(d) Manifold Visualization (e) Radius Distribution (f) Angular Distribution

Figure 2: Visualization of manifold geometry before (top) and after (bottom) alignment.



(a) Manifold Curvature c (b) Coefficient (λ_1, λ_2)

Figure 3: Sensitivity to curvature c and coefficients (λ_1, λ_2).



(a) Radial (b) Angular

Figure 4: Counterfactual disentanglement via radial and angular.

the strongest performance across node-density, edge-density, and feature-shift scenarios. We attribute this to the two modules of our framework: polar endpoint regularization reduces structural-scale mismatch while preserving scale-normalized class semantics, and topology-conditioned polar flow matching replaces adversarial minimax alignment with stable geometry-aware transport supervision over compatible source–target regions. Third, among the three geometries, the hyperbolic variant DisRFM-$\mathbb{H}^n$ obtains the best mean performance in 8 out of the 12 scenarios, indicating its advantage under hierarchy-rich structural shifts. The spherical and Euclidean variants remain competitive, suggesting gains come not only from curvature choice but also from the proposed polar endpoint and transport objectives. More results and analysis are provided in Appendix K.1.

Ablation Study. To examine the contribution of each component in DisRFM, we conduct ablations on four variants: 1) DisRFM w/o FM, which removes the flow-based transport mechanism; 2) DisRFM w/o RA, which excludes the radial alignment loss; 3) DisRFM w/o AA, which excludes the angular alignment loss; and 4) DisRFM w/o PE, which disables the polar disentanglement strategy entirely. Results are reported in Table 2, and we draw three observations: (1) DisRFM outperforms DisRFM w/o FM, confirming that flow-based transport provides principled trajectory guidance to bridge domain discrepancies. (2) The performance drops in DisRFM w/o RA and DisRFM w/o AA validate the necessity of independent geometric constraints. (3) DisRFM w/o PE exhibits the sharpest degradation, underscoring that disentangling structural and semantic components is critical for robust graph adaptation. More results are reported in Appendix K.2.

Table 2: The results of ablation studies on the Mutagenicity dataset. **Bold** results indicate the best performance.

Methods	M0→M1	M1→M0	M0→M2	M2→M0	M0→M3	M3→M0
DisRFM w/o FM	76.2	73.6	69.6	74.0	65.3	66.2
DisRFM w/o RA	76.9	74.4	70.7	73.9	66.1	67.1
DisRFM w/o AA	76.4	74.0	70.2	73.5	65.7	66.0
DisRFM w/o PE	75.9	72.9	69.0	72.5	64.6	64.8
DisRFM	**78.3**	**76.3**	**71.3**	**76.1**	**67.7**	**69.2**

Sensitivity Analysis. We conduct a sensitivity analysis of manifold curvature c and balance coefficients (λ_1, λ_2), shown in Figure 3, to evaluate robustness of the chosen geometry and loss weighting under adaptation. As shown in Figure 3(a), the model performs best at $c = -1.0$. This reflects a trade-off: curvatures close to 0 fail to capture complex hierarchies due to limited Euclidean capacity, while excessively large curvature causes geometric distortion, hurting representation quality. We therefore choose $c = -1.0$ as the default curvature. We set $\lambda_3 = 0.1$ and test (λ_1, λ_2) varying between (0.1, 0.9). Figure 3(b) shows that DisRFM reaches its optimum when $\lambda_1 = 0.1$ and $\lambda_2 = 0.1$. Accuracy drops as these weights increase, suggesting that excessive geometric alignment can overwhelm the classification objective and hurt adaptation. More results appear in Appendix K.3.

Visualization of Manifold Geometry. We visualize polar geometry before and after alignment in Figure 2, using radius–angle plots and class-conditioned histograms. Before alignment, source and target embeddings exhibit radial mismatch: target samples concentrate at smaller radii, with limited overlap. Same-class samples are poorly mixed across domains, with only partial angular alignment. After alignment, radial profiles become closer, source–target overlap increases near the origin, and angular histograms show class-wise correspondence. Domains mix better within each class, without collapsing inter-class angular margins, while different classes remain directionally separated. These

patterns support our coordinate-wise design: radial regularization calibrates structural statistics, whereas angular alignment preserves class-discriminative semantics during transport.

Disentanglement Analysis. To assess whether learned polar coordinates support an operational structure–semantics separation, rather than global identifiability, we intervene on radial and angular factors. We scale r while fixing u, and interpolate u across classes while fixing r, then evaluate the perturbed representations with a structural probe and semantic classifier. As shown in Figure 4, radial scaling monotonically increases predicted node count while class probability remains nearly unchanged. Conversely, angular interpolation changes semantic probability but leaves the structural score stable. These results suggest that r mainly controls structure-related variation, whereas u carries class-discriminative semantics. More results appear in Appendix K.5.

5 Conclusion

We proposed DisRFM, a geometry-aware framework for graph domain adaptation under structural and semantic distribution shifts. DisRFM uses polar endpoint regularization to calibrate structure-sensitive radial scales and align scale-normalized angular semantics with confidence-filtered pseudo-labels. It replaces adversarial domain confusion with topology-conditioned Riemannian flow matching, enabling stable transport between class- and structure-compatible source–target representations. Theoretical analysis explains the structural risk of unconditional domain confusion and supports the radial-angular design and flow-based transport objective. Experiments under node-density, edge-density, and feature shifts show strong and stable performance. Future work will explore adaptive curvature learning and extensions to heterogeneous and generative graph tasks.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the International Conference on Machine Learning*, pages 214–223. PMLR, 2017.

[2] Jinheon Baek, Minki Kang, and Sung Ju Hwang. Accurate learning of graph representations with graph multiset pooling. *arXiv preprint arXiv:2102.11533*, 2021.

[3] Cristian Bodnar, Fabrizio Frasca, Nina Otter, Yuguang Wang, Pietro Lio, Guido F Montufar, and Michael Bronstein. Weisfeiler and lehman go cellular: Cw networks. *Proceedings of the Conference on Neural Information Processing Systems*, 34:2625–2640, 2021.

[4] Haitz Sáez de Ocáriz Borde, Anees Kazi, Federico Barbero, and Pietro Liò. Latent graph inference using product manifolds. *arXiv preprint arXiv:2211.16199*, 2022.

[5] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[6] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Proceedings of the Conference on Neural Information Processing Systems*, 32, 2019.

[7] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 627–636, 2019.

[8] Dingshuo Chen, Shuchen Xue, Liuji Chen, Yingheng Wang, Qiang Liu, Shu Wu, Zhi-Ming Ma, and Liang Wang. Graffe: Graph representation learning via diffusion probabilistic models. *arXiv preprint arXiv:2505.04956*, 2025.

[9] Wei Chen, Guo Ye, Yakun Wang, Zhao Zhang, Libang Zhang, Daixin Wang, Zhiqiang Zhang, and Fuzhen Zhuang. Smoothness really matters: A simple yet effective approach for unsupervised graph domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 15875–15883, 2025.

[10] Antoine Collas, Ce Ju, Nicolas Salvy, and Bertrand Thirion. Riemannian flow matching for brain connectivity matrices via pullback geometry. *arXiv preprint arXiv:2505.18193*, 2025.

[11] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

[12] Paul D Dobson and Andrew J Doig. Distinguishing enzyme structures from non-enzymes without alignments. *Journal of molecular biology*, 330(4):771–783, 2003.

[13] Ruiyi Fang, Bingheng Li, Zhao Kang, Qiuhao Zeng, Nima Hosseini Dashtbayaz, Ruizhi Pu, Boyu Wang, and Charles Ling. On the benefits of attribute-driven graph domain adaptation. *arXiv preprint arXiv:2502.06808*, 2025.

[14] Yanhong Fei, Yingjie Liu, Chentao Jia, Zhengyu Li, Xian Wei, and Mingsong Chen. A survey of geometric optimization for deep learning: from euclidean space to riemannian manifold. *ACM Computing Surveys*, 57(5):1–37, 2025.

[15] Wenzheng Feng, Jie Zhang, Yuxiao Dong, Yu Han, Huanbo Luan, Qian Xu, Qiang Yang, Evgeny Kharlamov, and Jie Tang. Graph random neural networks for semi-supervised learning on graphs. *Advances in neural information processing systems*, 33:22092–22103, 2020.

[16] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *Advances in neural information processing systems*, 31, 2018.

[17] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

[18] Shayan Gharib, Marcelo Hartmann, and Arto Klami. Geometric moment alignment for domain adaptation via siegel embeddings. *arXiv preprint arXiv:2510.14666*, 2025.

[19] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[20] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. Rolx: structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1231–1239, 2012.

[21] Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. Ogb-lsc: A large-scale challenge for machine learning on graphs. *arXiv preprint arXiv:2103.09430*, 2021.

[22] Hermann Karcher. Riemannian center of mass and mollifier smoothing. *Communications on pure and applied mathematics*, 30(5):509–541, 1977.

[23] Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of medicinal chemistry*, 48(1):312–320, 2005.

[24] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations*, 2017.

[25] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[26] Tong Lin and Hongbin Zha. Riemannian manifold learning. *IEEE transactions on pattern analysis and machine intelligence*, 30(5):796–809, 2008.

[27] Meihan Liu, Zeyu Fang, Zhen Zhang, Ming Gu, Sheng Zhou, Xin Wang, and Jiajun Bu. Rethinking propagation for unsupervised graph domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13963–13971, 2024.

[28] Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. *Advances in neural information processing systems*, 32, 2019.

[29] Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural re-weighting improves graph domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 21778–21793. PMLR, 2023.

[30] Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. *Proceedings of the International Conference on Machine Learning*, 2024.

[31] Tianhang Long, Yanfeng Sun, Junbin Gao, Yongli Hu, and Baocai Yin. Domain adaptation as optimal transport on grassmann manifolds. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10):7196–7209, 2022.

[32] Marina Meilă and Hanyu Zhang. Manifold learning: What, how, and why. *Annual Review of Statistics and Its Application*, 11(1):393–417, 2024.

[33] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018.

[34] Gaspard Michel, Giannis Nikolentzos, Johannes F Lutzeyer, and Michalis Vazirgiannis. Path neural networks: Expressive and accurate graph neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 24737–24755. PMLR, 2023.

[35] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017.

[36] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[37] Ziyue Qiao, Xiao Luo, Meng Xiao, Hao Dong, Yuanchun Zhou, and Hui Xiong. Semi-supervised domain adaptation in graph transfer learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2279–2287, 2023.

[38] Adam G Rustad et al. Graph integration for diffusion-based manifold alignment. In *2024 International Conference on Machine Learning and Applications (ICMLA)*, pages 1–8. IEEE, 2024.

[39] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International conference on machine learning*, pages 4460–4469. PMLR, 2018.

[40] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[41] Xiao Shen, Quanyu Dai, Fu-lai Chung, Wei Lu, and Kup-Sze Choi. Adversarial deep network embedding for cross-network node classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 2991–2999, 2020.

[42] Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-lehman graph kernels. *The Journal of Machine Learning Research.*, 12(9), 2011.

[43] Peeyush Singhal, Rahee Walambe, Sheela Ramanna, and Ketan Kotecha. Domain adaptation: challenges, methods, datasets, and applications. *IEEE access*, 11:6973–7020, 2023.

[44] Ke Sun, Zhanxing Zhu, and Zhouchen Lin. Adagcn: Adaboosting graph convolutional networks into deep models. In *International Conference on Learning Representations*, 2020.

[45] Li Sun, Zhenhao Huang, Zixi Wang, Feiyang Wang, Hao Peng, and Philip S Yu. Motif-aware riemannian graph neural network with generative-contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9044–9052, 2024.

[46] Li Sun, Junda Ye, Hao Peng, Feiyang Wang, and Philip S Yu. Self-supervised continual graph learning in adaptive riemannian spaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4633–4642, 2023.

[47] Li Sun, Suyang Zhou, Bowen Fang, Hechuan Zhang, Junda Ye, Yutong Ye, and Philip S Yu. Trace: Structural riemannian bridge matching for transferable source localization in information propagation. In *Proceedings of IJCAI*, 2025.

[48] Jeffrey J Sutherland, Lee A O'brien, and Donald F Weaver. Spline-fitting with a genetic algorithm: A method for developing classification structure- activity relationships. *Journal of chemical information and computer sciences*, 43(6):1906–1915, 2003.

[49] Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincar\'e glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*, 2018.

[50] Nikil Wale, Ian A Watson, and George Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 14:347–375, 2008.

[51] Yujie Wang, Shuo Zhang, Junda Ye, Hao Peng, and Li Sun. A mixed-curvature graph diffusion model. In *Proceedings of the International Conference on Information and Knowledge Management*, pages 2482–2492, 2024.

[52] Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. Unsupervised domain adaptive graph convolutional networks. In *Proceedings of the ACM Web Conference*, pages 1457–1467, 2020.

[53] Man Wu, Xin Zheng, Qin Zhang, Xiao Shen, Xiong Luo, Xingquan Zhu, and Shirui Pan. Graph learning under distribution shifts: A comprehensive survey on domain adaptation, out-of-distribution, and continual learning. *arXiv preprint arXiv:2402.16374*, 2024.

[54] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[55] Yuecong Xu, Haozhi Cao, Lihua Xie, Xiao-Li Li, Zhenghua Chen, and Jianfei Yang. Video unsupervised domain adaptation with deep learning: A comprehensive survey. *ACM Computing Surveys*, 56(12):1–36, 2024.

[56] Nan Yin, Li Shen, Baopu Li, Mengzhu Wang, Xiao Luo, Chong Chen, Zhigang Luo, and Xian-Sheng Hua. Deal: An unsupervised domain adaptive framework for graph-level classification. In *Proceedings of the ACM International Conference on Multimedia*, pages 3470–3479, 2022.

[57] Nan Yin, Li Shen, Mengzhu Wang, Long Lan, Zeyu Ma, Chong Chen, Xian-Sheng Hua, and Xiao Luo. Coco: A coupled contrastive framework for unsupervised domain adaptive graph classification. In *Proceedings of the International Conference on Machine Learning*, pages 40040–40053. PMLR, 2023.

[58] Nan Yin, Li Shen, Mengzhu Wang, Xinwang Liu, Chong Chen, and Xian-Sheng Hua. Dream: a dual variational framework for unsupervised graph domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[59] Kexin Zhang, Shuhan Liu, Song Wang, Weili Shi, Chen Chen, Pan Li, Sheng Li, Jundong Li, and Kaize Ding. A survey of deep graph learning under distribution shifts: from graph out-of-distribution generalization to adaptation. *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2024.

[60] Zhenyue Zhang, Jing Wang, and Hongyuan Zha. Adaptive manifold learning. *IEEE transactions on pattern analysis and machine intelligence*, 34(2):253–265, 2011.

[61] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European Conference on Computer Vision.*, pages 289–305, 2018.

A Riemannian Geometry and Stereographic Model

Problem Definition. We focus on the problem of GDA for graph classification. Let $\mathcal{D}_S = \{(\mathcal{G}_i^S, y_i^S)\}_{i=1}^{N_S}$ and $\mathcal{D}_T = \{\mathcal{G}_j^T\}_{j=1}^{N_T}$ denote the labeled source and unlabeled target domains. Both domains share a label space $\mathcal{Y}$ but under distribution shift $\mathbb{P}_S(\mathcal{D}_S) \neq \mathbb{P}_T(\mathcal{D}_T)$. Each graph is represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, X)$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, and $X \in \mathbb{R}^{|\mathcal{V}| \times F}$ denotes the node features. Our goal is to learn a geometry-aware encoder $f_\theta : \mathcal{G} \to \mathcal{M}_c$ and classifier $g_\phi : \mathcal{M}_c \to \mathcal{Y}$ to minimize classification error on $\mathcal{D}_T$, where $\mathcal{M}_c$ is a latent Riemannian manifold.

Let $\mathcal{M}_c$ denote the d-dimensional Riemannian manifold with constant curvature c, defined as the interior of a ball $\mathcal{M}_c = \{x \in \mathbb{R}^d : -c\|x\|^2 < 1\}$. The Riemannian metric tensor g_x^c is conformally equivalent to the Euclidean metric g^E, given by $g_x^c = (\lambda_x^c)^2 g^E$, where the conformal factor is $\lambda_x^c = 2/(1 + c\|x\|^2)$. The curvature c determines the geometry: $c < 0$ for the Hyperbolic space $\mathbb{H}^d$, $c > 0$ for the Spherical space $\mathbb{S}^d$, and $c \to 0$ for the Euclidean space $\mathbb{E}^d$.

Exponential and Logarithmic Maps. Transformations between the manifold $\mathcal{M}_c$ and the tangent space $T_x\mathcal{M}_c$ (isomorphic to $\mathbb{R}^d$) are mediated by the Exponential map $\mathrm{Exp}_x^c : T_x\mathcal{M}_c \to \mathcal{M}_c$ and the Logarithmic map $\mathrm{Log}_x^c : \mathcal{M}_c \to T_x\mathcal{M}_c$. For the origin 0, these maps are:

$$\mathrm{Exp}_0^c(v) = \tan_c(\|v\|)\frac{v}{\|v\|}, \quad \mathrm{Log}_0^c(y) = \tan_c^{-1}(\|y\|)\frac{y}{\|y\|},$$

where $\tan_c(\cdot)$ and $\tan_c^{-1}(\cdot)$ are curvature-dependent trigonometric functions (*e.g.*, $\tanh$ /arctanh for $c < 0$).

Geodesics and Distance. Geodesics generalize the concept of straight lines to Riemannian manifolds, representing the locally length-minimizing curves. The unique geodesic segment $\gamma : [0, 1] \to \mathcal{M}_c$ connecting a source point x to a target point y is parameterized as: $\gamma(t) = \mathrm{Exp}_x^c(t \cdot \mathrm{Log}_x^c(y))$. The Riemannian distance $d_{\mathcal{M}}(x, y)$ is the length of this geodesic, calculated as $d_{\mathcal{M}}(x, y) = \|\mathrm{Log}_x^c(y)\|_x$.

Parallel Transport. Comparing tangent vectors at distinct points requires Parallel Transport. The transport of a vector $v \in T_0\mathcal{M}_c$ to $T_x\mathcal{M}_c$ along the geodesic connecting 0 and x is given by $P_{0\to x}(v) = (\lambda_0^c/\lambda_x^c)v$. This operator ensures that the geometric properties of feature vectors (e.g., orientation) are preserved during transport.

Riemannian Polar Coordinates. Formally, any point $x \in \mathcal{M}_c \setminus \{0\}$ can be uniquely represented via the polar decomposition of its tangent vector in $T_0\mathcal{M}_c$. We define the radial coordinate r_x and angular coordinate u_x as:

$$r_x = \|\mathrm{Log}_0^c(x)\|_2, \quad u_x = \frac{\mathrm{Log}_0^c(x)}{\|\mathrm{Log}_0^c(x)\|_2}.$$

Mathematically, r_x corresponds to the Riemannian geodesic distance from the origin, *i.e.*, $r_x = d_{\mathcal{M}}(0, x)$, while $u_x \in \mathbb{S}^{d-1}$ represents the direction in the tangent space.

Riemannian Graph Convolution. Following the unified framework of Hyperbolic Graph Neural Networks (HGNN) [28], we generalize standard GCNs to a constant curvature Riemannian manifold $\mathcal{M}_c$. The sign of the curvature c adaptively dictates the underlying geometry, unifying Hyperbolic space $\mathbb{H}$ ($c < 0$), Euclidean space $\mathbb{E}$ ($c = 0$), and Spherical Space $\mathbb{S}$ ($c > 0$) space within a single formalism. Fundamental operations, such as linear transformation and aggregation, are executed by mapping features to the tangent space $T_{\mathbf{0}}\mathcal{M}_c$, performing Euclidean computations, and retracting the results back to the manifold. Specifically, the linear transformation of a node feature $\mathbf{h}_j^{(l)} \in \mathcal{M}_c$ via a weight matrix $\mathbf{W}^{(l)}$ is defined by Möbius matrix-vector multiplication:

$$\mathbf{m}_j^{(l)} = \mathbf{W}^{(l)} \otimes_c \mathbf{h}_j^{(l)} = \mathrm{Exp}_{\mathbf{0}}^c\left(\mathbf{W}^{(l)} \cdot \mathrm{Log}_{\mathbf{0}}^c(\mathbf{h}_j^{(l)})\right).$$

Then, neighborhood aggregation is grounded in the Fréchet mean [22], and approximated in the tangent space, resulting in the unified layer-wise update rule:

$$\mathbf{h}_i^{(l+1)} = \mathrm{Exp}_{\mathbf{0}}^c\left(\sigma\Big(\sum_{j \in \mathcal{N}(i) \cup \{i\}} \alpha_{ij} \cdot \mathrm{Log}_{\mathbf{0}}^c(\mathbf{m}_j^{(l)})\Big)\right). \tag{6}$$

This architecture ensures that representation learning faithfully respects the underlying manifold geometry.

B Proof of Theorem 1

Theorem 1 (A Structural Failure Mode of Domain Confusion) *Consider a binary structural variable $A \in \{0,1\}$ with equal priors $\Pr(A=0) = \Pr(A=1) = 1/2$, label $Y = A$, and domain variable $D \in \{0,1\}$ with $\Pr(D = A) = 1 - \eta$ for some $0 \le \eta < 1/2$. Assume a representation Z depends on the graph only through A in this toy model. If an adversarial objective enforces approximate domain invariance*

$$\mathrm{TV}\big(P_{Z|D=0}, P_{Z|D=1}\big) \le \delta,$$

then the total variation between the label-conditionals of Z is bounded by

$$\mathrm{TV}\big(P_{Z|Y=0}, P_{Z|Y=1}\big) \le \frac{\delta}{1-2\eta}.$$

Consequently, any classifier using Z has error at least

$$\inf_h \Pr(h(Z) \neq Y) \ge \frac{1}{2}\left(1 - \frac{\delta}{1-2\eta}\right).$$

Proof 1 *Let $\mathcal{Z}$ denote the measurable representation space of Z, and let $\mathcal{B}(\mathcal{Z})$ denote the collection of measurable subsets of $\mathcal{Z}$. For two probability measures P and Q on $\mathcal{Z}$, we use the convention*

$$\mathrm{TV}(P,Q) = \sup_{B \in \mathcal{B}(\mathcal{Z})} |P(B) - Q(B)|.$$

For $a, d \in \{0,1\}$, define

$$\mu_a := P_{Z|A=a}, \qquad \nu_d := P_{Z|D=d}.$$

Here, μ_a is the conditional distribution of the representation Z given the structural variable $A = a$, and ν_d is the conditional distribution of Z given the domain variable $D = d$. Since $Y = A$, we also have

$$P_{Z|Y=a} = P_{Z|A=a} = \mu_a.$$

The assumption that Z depends on the graph only through A means that, in this toy model, Z is conditionally independent of D given A. Equivalently,

$$P_{Z|A=a,D=d} = P_{Z|A=a} = \mu_a, \qquad a, d \in \{0,1\}.$$

Therefore, each domain-conditional distribution ν_d can be written as a mixture of the two structure-conditionals μ_0 and μ_1.

Assume equal class priors, i.e.,

$$\Pr(A=0) = \Pr(A=1) = \frac{1}{2}.$$

Let

$$e_0 := \Pr(D=1 \mid A=0), \qquad e_1 := \Pr(D=0 \mid A=1),$$

where e_0 and e_1 are the two possible mismatch rates between the structural variable A and the domain variable D. Since

$$\Pr(D = A) = 1 - \eta,$$

we have

$$\Pr(D \neq A) = \eta.$$

Under equal class priors,

$$\eta = \frac{1}{2}e_0 + \frac{1}{2}e_1,$$

and hence

$$e_0 + e_1 = 2\eta.$$

Now define the two mixture coefficients

$$\alpha_0 := \Pr(A=0 \mid D=0), \qquad \alpha_1 := \Pr(A=0 \mid D=1).$$

By Bayes' rule,

$$\alpha_0 = \frac{\Pr(D=0 \mid A=0)\Pr(A=0)}{\Pr(D=0)} = \frac{1-e_0}{1-e_0+e_1},$$

and

$$\alpha_1 = \frac{\Pr(D=1 \mid A=0)\Pr(A=0)}{\Pr(D=1)} = \frac{e_0}{e_0+1-e_1}.$$

Therefore,

$$\nu_0 = \alpha_0\mu_0 + (1-\alpha_0)\mu_1,$$

and

$$\nu_1 = \alpha_1\mu_0 + (1-\alpha_1)\mu_1.$$

Subtracting the two signed measures gives

$$\nu_0 - \nu_1 = (\alpha_0 - \alpha_1)(\mu_0 - \mu_1).$$

Taking total variation on both sides yields

$$\mathrm{TV}(\nu_0, \nu_1) = |\alpha_0 - \alpha_1|\,\mathrm{TV}(\mu_0, \mu_1).$$

It remains to lower bound $|\alpha_0 - \alpha_1|$*. Since* $\eta < 1/2$*, we have* $e_0 + e_1 < 1$*, so* $\alpha_0 > \alpha_1$*. Thus*

$$\begin{aligned}\alpha_0 - \alpha_1 &= \frac{1-e_0}{1-e_0+e_1} - \frac{e_0}{1+e_0-e_1} \\ &= \frac{1-e_0-e_1}{(1-e_0+e_1)(1+e_0-e_1)} \\ &= \frac{1-2\eta}{1-(e_0-e_1)^2}.\end{aligned}$$

Because $1-(e_0-e_1)^2 \le 1$ *and is positive, we obtain*

$$\alpha_0 - \alpha_1 \ge 1 - 2\eta.$$

Combining this with the previous identity gives

$$\mathrm{TV}(\nu_0, \nu_1) \ge (1-2\eta)\,\mathrm{TV}(\mu_0, \mu_1).$$

The adversarial domain-invariance condition assumes

$$\mathrm{TV}(P_{Z|D=0}, P_{Z|D=1}) = \mathrm{TV}(\nu_0, \nu_1) \le \delta.$$

Therefore,

$$(1-2\eta)\,\mathrm{TV}(\mu_0, \mu_1) \le \delta.$$

Since $\eta < 1/2$*, we have* $1-2\eta > 0$*, and hence*

$$\mathrm{TV}(\mu_0, \mu_1) \le \frac{\delta}{1-2\eta}.$$

Using $Y = A$*, this is equivalent to*

$$\mathrm{TV}\big(P_{Z|Y=0}, P_{Z|Y=1}\big) \le \frac{\delta}{1-2\eta}.$$

Finally, consider any deterministic classifier $h : \mathcal{Z} \to \{0,1\}$*. Let*

$$B_h := \{z \in \mathcal{Z} : h(z) = 1\}$$

be the measurable decision region where h *predicts class* 1*. Under equal class priors, the classification error of* h *is*

$$\begin{aligned}\Pr(h(Z) \ne Y) &= \frac{1}{2}\Pr(h(Z)=1 \mid Y=0) + \frac{1}{2}\Pr(h(Z)=0 \mid Y=1) \\ &= \frac{1}{2}\mu_0(B_h) + \frac{1}{2}\mu_1(B_h^c) \\ &= \frac{1}{2}\left[1 - \big(\mu_1(B_h) - \mu_0(B_h)\big)\right].\end{aligned}$$

By the definition of total variation,

$$\mu_1(B_h) - \mu_0(B_h) \le \mathrm{TV}(\mu_0, \mu_1).$$

Thus, for every classifier h,

$$\Pr(h(Z) \neq Y) \ge \frac{1}{2}\left(1 - \mathrm{TV}(\mu_0, \mu_1)\right).$$

Taking the infimum over all classifiers h and applying the bound above gives

$$\inf_h \Pr(h(Z) \neq Y) \ge \frac{1}{2}\left(1 - \frac{\delta}{1-2\eta}\right).$$

This completes the proof.

C Proof of Theorem 2

Theorem 2 (First-Order Orthogonality of Polar Endpoint Losses) *Let $\mathcal{M}_c$ be a d-dimensional constant-curvature manifold and let $o \in \mathcal{M}_c$ be the origin. Work on a geodesic normal ball $B_o(\rho)$, with $\rho < \pi/\sqrt{c}$ when $c > 0$, and exclude o. For $\mathbf{z} = (r, u) \in B_o(\rho) \setminus \{o\}$, the polar metric is Eq.* (1)*. For a batch $\mathbf{z} = (\mathbf{z}_i)_{i=1}^B$, equip $\mathcal{M}_c^B$ with the product metric. Let $\mathcal{L}_{\mathrm{rad}}(\mathbf{z}) = \ell_r(r_{1:B})$ and $\mathcal{L}_{\mathrm{ang}}(\mathbf{z}) = \ell_u(u_{1:B})$ be almost-everywhere differentiable radial-only and angular-only endpoint losses. Then, wherever the gradients exist,*

$$\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{ang}} \rangle_g = 0.$$

If a differentiable radial reliability function weights the angular loss $\mathcal{L}_{\mathrm{wang}} = \sum_i \alpha(r_i)\ell_i(u_i)$, then

$$\left| \langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{wang}} \rangle_g \right| \le \sum_{i=1}^{B} |\partial_{r_i} \ell_r| \, |\alpha'(r_i)| \, |\ell_i(u_i)|.$$

For the choice $\alpha(r) = e^{-r}$, the coupling term is bounded by $\sum_i |\partial_{r_i}\ell_r| e^{-r_i} |\ell_i(u_i)|$.

Proof 2 *We prove the statement pointwise at any batch $\mathbf{z} = (z_1, \ldots, z_B) \in \mathcal{M}_c^B$ where all involved derivatives exist. For each sample $z_i \in B_o(\rho) \setminus \{o\}$, write its geodesic polar coordinates as*

$$z_i = (r_i, u_i),$$

where $r_i = d_{\mathcal{M}_c}(o, z_i) > 0$ is the geodesic radius and $u_i \in \mathbb{S}^{d-1}$ is the unit angular direction. The exclusion of o ensures that u_i is well-defined. When $c > 0$, the condition $\rho < \pi/\sqrt{c}$ keeps the normal ball away from the cut locus, so the polar coordinates are smooth on the considered region.

By the polar metric in Eq. (1)*, for a tangent vector*

$$\xi_i = (a_i, b_i) \in T_{z_i}\mathcal{M}_c,$$

where $a_i \in \mathbb{R}$ is the radial component and $b_i \in T_{u_i}\mathbb{S}^{d-1}$ is the angular component, the metric takes the form

$$g_{z_i}(\xi_i, \xi_i') = a_i a_i' + S_c(r_i)^2 \langle b_i, b_i' \rangle_{\mathbb{S}},$$

where $\xi_i' = (a_i', b_i')$, $\langle \cdot, \cdot \rangle_{\mathbb{S}}$ is the canonical metric on the unit sphere, and

$$S_c(r) = \begin{cases} \sin(\sqrt{c}\, r)/\sqrt{c}, & c > 0, \\ r, & c = 0, \\ \sinh(\sqrt{-c}\, r)/\sqrt{-c}, & c < 0. \end{cases}$$

Since $r_i > 0$ and the chart stays inside the normal ball, $S_c(r_i) > 0$.

The batch space $\mathcal{M}_c^B$ is equipped with the product metric. Hence, for two batch tangent vectors

$$\xi = (\xi_1, \ldots, \xi_B), \qquad \xi' = (\xi_1', \ldots, \xi_B'),$$

with $\xi_i = (a_i, b_i)$ and $\xi_i' = (a_i', b_i')$, we have

$$g_{\mathbf{z}}^{(B)}(\xi, \xi') = \sum_{i=1}^{B} \left[a_i a_i' + S_c(r_i)^2 \langle b_i, b_i' \rangle_{\mathbb{S}} \right].$$

Now let $F : \mathcal{M}_c^B \to \mathbb{R}$ be any differentiable scalar function. In polar coordinates, its differential along a batch tangent vector $\xi = (a_i, b_i)_{i=1}^B$ is

$$dF[\xi] = \sum_{i=1}^{B} \left[\partial_{r_i} F \cdot a_i + \langle \nabla_{\mathbb{S},i} F, b_i \rangle_{\mathbb{S}} \right],$$

where $\partial_{r_i} F$ is the partial derivative of F with respect to the radius r_i, and $\nabla_{\mathbb{S},i} F \in T_{u_i}\mathbb{S}^{d-1}$ is the spherical gradient of F with respect to the angular variable u_i.

By definition, the Riemannian gradient $\nabla_g F$ is the unique tangent vector satisfying

$$g_{\mathbf{z}}^{(B)}(\nabla_g F, \xi) = dF[\xi] \qquad \text{for all } \xi \in T_{\mathbf{z}}\mathcal{M}_c^B.$$

Comparing the radial and angular components gives

$$(\nabla_g F)_i = \left(\partial_{r_i} F,\ S_c(r_i)^{-2} \nabla_{\mathbb{S},i} F \right).$$

Therefore, for two differentiable scalar functions F and H,

$$\langle \nabla_g F, \nabla_g H \rangle_g = \sum_{i=1}^{B} \left[\partial_{r_i} F \, \partial_{r_i} H + S_c(r_i)^{-2} \langle \nabla_{\mathbb{S},i} F, \nabla_{\mathbb{S},i} H \rangle_{\mathbb{S}} \right].$$

We first apply this identity to

$$\mathcal{L}_{\mathrm{rad}}(\mathbf{z}) = \ell_r(r_{1:B}), \qquad \mathcal{L}_{\mathrm{ang}}(\mathbf{z}) = \ell_u(u_{1:B}),$$

where $r_{1:B} = (r_1, \ldots, r_B)$ and $u_{1:B} = (u_1, \ldots, u_B)$. Since $\mathcal{L}_{\mathrm{rad}}$ depends only on the radial variables, it has no angular derivative:

$$\nabla_{\mathbb{S},i} \mathcal{L}_{\mathrm{rad}} = 0, \qquad i = 1, \ldots, B.$$

Since $\mathcal{L}_{\mathrm{ang}}$ depends only on the angular variables, it has no radial derivative:

$$\partial_{r_i} \mathcal{L}_{\mathrm{ang}} = 0, \qquad i = 1, \ldots, B.$$

Substituting these two identities into the inner-product formula yields

$$\begin{aligned} \langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{ang}} \rangle_g &= \sum_{i=1}^{B} \left[\partial_{r_i} \mathcal{L}_{\mathrm{rad}} \cdot 0 + S_c(r_i)^{-2} \langle 0, \nabla_{\mathbb{S},i} \mathcal{L}_{\mathrm{ang}} \rangle_{\mathbb{S}} \right] \\ &= 0. \end{aligned}$$

This proves the first-order orthogonality between purely radial and purely angular endpoint losses.

We next consider the radially weighted angular loss

$$\mathcal{L}_{\mathrm{wang}}(\mathbf{z}) = \sum_{i=1}^{B} \alpha(r_i) \ell_i(u_i),$$

where $\alpha : \mathbb{R}_+ \to \mathbb{R}$ is differentiable and $\ell_i : \mathbb{S}^{d-1} \to \mathbb{R}$ is the angular loss for sample i. For this loss, the radial derivative is

$$\partial_{r_i} \mathcal{L}_{\mathrm{wang}} = \alpha'(r_i) \ell_i(u_i),$$

where $\alpha'(r_i)$ denotes the derivative of α evaluated at r_i. Its angular derivative is

$$\nabla_{\mathbb{S},i} \mathcal{L}_{\mathrm{wang}} = \alpha(r_i) \nabla_{\mathbb{S}} \ell_i(u_i).$$

Using the same inner-product formula and the fact that $\nabla_{\mathbb{S},i} \mathcal{L}_{\mathrm{rad}} = 0$, we obtain

$$\begin{aligned} \langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{wang}} \rangle_g &= \sum_{i=1}^{B} \partial_{r_i} \mathcal{L}_{\mathrm{rad}} \cdot \partial_{r_i} \mathcal{L}_{\mathrm{wang}} \\ &= \sum_{i=1}^{B} \partial_{r_i} \ell_r \cdot \alpha'(r_i) \ell_i(u_i), \end{aligned}$$

where $\partial_{r_i}\ell_r$ is shorthand for $\partial\ell_r(r_{1:B})/\partial r_i$.

Taking absolute values and applying the triangle inequality gives

$$\left|\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{wang}} \rangle_g\right| = \left|\sum_{i=1}^{B} \partial_{r_i}\ell_r \cdot \alpha'(r_i)\ell_i(u_i)\right| \leq \sum_{i=1}^{B} |\partial_{r_i}\ell_r|\,|\alpha'(r_i)|\,|\ell_i(u_i)|.$$

This proves the claimed bound.

Finally, for the reliability function used in the main text,

$$\alpha(r) = e^{-r},$$

we have

$$\alpha'(r) = -e^{-r}, \qquad |\alpha'(r)| = e^{-r}.$$

Substituting this into the previous inequality gives

$$\left|\langle \nabla_g \mathcal{L}_{\mathrm{rad}}, \nabla_g \mathcal{L}_{\mathrm{wang}} \rangle_g\right| \leq \sum_{i=1}^{B} |\partial_{r_i}\ell_r|\, e^{-r_i}\, |\ell_i(u_i)|.$$

This completes the proof.

D Proof of Theorem 3

Theorem 3 (Radial Structural Proxy Bound) *Let P_S and P_T be source and target graph distributions. For a graph G, let $r(G) = \|\operatorname{Log}_o^c(f_\theta(G))\|_{g_o}$ and let $a(G) \in \mathbb{R}^m$ denote topology-related graph statistics. Denote by $r_\# P$ and $a_\# P$ the pushforward distributions induced by $r(G)$ and $a(G)$. Suppose there exists an L_a-Lipschitz map $h : \mathbb{R}_+ \to \mathbb{R}^m$ such that, for $D \in \{S, T\}$, $\mathbb{E}_{G\sim P_D}[\|a(G) - h(r(G))\|] \leq \epsilon_D$. Then*

$$W_1(a_\# P_S, a_\# P_T) \leq L_a W_1(r_\# P_S, r_\# P_T) + \epsilon_S + \epsilon_T.$$

For equal-size empirical distributions, $W_1(\widehat{P}_S^r, \widehat{P}_T^r) = \frac{1}{B}\sum_{k=1}^{B} |\mathcal{R}_S[k] - \mathcal{R}_T[k]|$, which is Eq. (2)*.*

Proof 3 *Let $\mathcal{G}$ denote the graph space. For a graph distribution P on $\mathcal{G}$, the pushforward distribution induced by the radial map $r : \mathcal{G} \to \mathbb{R}_+$ is denoted by $r_\# P$. That is, for any measurable set $B \subseteq \mathbb{R}_+$,*

$$r_\# P(B) = P\big(\{G \in \mathcal{G} : r(G) \in B\}\big).$$

Similarly, the pushforward distribution induced by the topology-statistic map $a : \mathcal{G} \to \mathbb{R}^m$ is denoted by $a_\# P$, meaning that for any measurable set $C \subseteq \mathbb{R}^m$,

$$a_\# P(C) = P\big(\{G \in \mathcal{G} : a(G) \in C\}\big).$$

For $D \in \{S, T\}$, let

$$G_D \sim P_D, \qquad R_D := r(G_D), \qquad A_D := a(G_D), \qquad H_D := h(r(G_D)) = h(R_D).$$

Here, G_D is a graph sampled from domain D, R_D is its radial coordinate, A_D is its topology-statistic vector, and H_D is the topology proxy predicted from the radius by the map h. By definition,

$$R_D \sim r_\# P_D, \qquad A_D \sim a_\# P_D, \qquad H_D \sim h_\#(r_\# P_D),$$

where $h_\#(r_\# P_D)$ denotes the pushforward of the radial distribution through the map h.

Recall that for two probability measures μ and ν on a metric space $(\mathcal{X}, d)$, the Wasserstein-1 distance is

$$W_1(\mu, \nu) = \inf_{\gamma \in \Pi(\mu,\nu)} \mathbb{E}_{(X,Y)\sim\gamma}\big[d(X, Y)\big],$$

where $\Pi(\mu, \nu)$ denotes the set of all couplings whose first marginal is μ and whose second marginal is ν.

We first decompose the discrepancy between the source and target structural statistics by inserting the radius-induced proxy distributions:

$$\begin{aligned} W_1(a_\# P_S, a_\# P_T) \le\ & W_1(a_\# P_S, h_\#(r_\# P_S)) \\ & + W_1(h_\#(r_\# P_S), h_\#(r_\# P_T)) \\ & + W_1(h_\#(r_\# P_T), a_\# P_T). \end{aligned}$$

This follows from the triangle inequality of W_1.

We now bound the first term. Consider the coupling between $a_\# P_S$ and $h_\#(r_\# P_S)$ induced by drawing the same source graph $G_S \sim P_S$ and pairing

$$A_S = a(G_S), \qquad H_S = h(r(G_S)).$$

This is a valid coupling because its two marginals are exactly $a_\# P_S$ and $h_\#(r_\# P_S)$. Therefore, by the definition of W_1,

$$\begin{aligned} W_1(a_\# P_S, h_\#(r_\# P_S)) &\le \mathbb{E}_{G_S \sim P_S}\big[\|a(G_S) - h(r(G_S))\|\big] \\ &\le \epsilon_S. \end{aligned}$$

The last inequality is exactly the assumed source-domain proxy error.

The same argument applies to the target domain. Using the coupling induced by the same target graph $G_T \sim P_T$, we obtain

$$\begin{aligned} W_1(h_\#(r_\# P_T), a_\# P_T) &\le \mathbb{E}_{G_T \sim P_T}\big[\|h(r(G_T)) - a(G_T)\|\big] \\ &\le \epsilon_T. \end{aligned}$$

It remains to bound the middle term

$$W_1(h_\#(r_\# P_S), h_\#(r_\# P_T)).$$

Let $\gamma_r \in \Pi(r_\# P_S, r_\# P_T)$ be any coupling between the source and target radial distributions. If $(R_S, R_T) \sim \gamma_r$, then

$$(h(R_S), h(R_T))$$

is a valid coupling between $h_\#(r_\# P_S)$ and $h_\#(r_\# P_T)$. Since $h : \mathbb{R}_+ \to \mathbb{R}^m$ is L_a-Lipschitz, we have

$$\|h(R_S) - h(R_T)\| \le L_a |R_S - R_T|.$$

Thus,

$$\begin{aligned} W_1(h_\#(r_\# P_S), h_\#(r_\# P_T)) &\le \mathbb{E}_{(R_S, R_T) \sim \gamma_r}\big[\|h(R_S) - h(R_T)\|\big] \\ &\le L_a \mathbb{E}_{(R_S, R_T) \sim \gamma_r}\big[|R_S - R_T|\big]. \end{aligned}$$

Taking the infimum over all couplings $\gamma_r \in \Pi(r_\# P_S, r_\# P_T)$ gives

$$W_1(h_\#(r_\# P_S), h_\#(r_\# P_T)) \le L_a W_1(r_\# P_S, r_\# P_T).$$

Combining the three bounds yields

$$\begin{aligned} W_1(a_\# P_S, a_\# P_T) &\le \epsilon_S + L_a W_1(r_\# P_S, r_\# P_T) + \epsilon_T \\ &= L_a W_1(r_\# P_S, r_\# P_T) + \epsilon_S + \epsilon_T. \end{aligned}$$

This proves the population-level radial structural proxy bound.

We next prove the empirical formula. Suppose the source and target minibatches have the same size B. Let their empirical radial distributions be

$$\widehat{P}_S^r = \frac{1}{B} \sum_{i=1}^{B} \delta_{r_i^S}, \qquad \widehat{P}_T^r = \frac{1}{B} \sum_{i=1}^{B} \delta_{r_i^T},$$

where $r_i^S = r(G_i^S)$ and $r_i^T = r(G_i^T)$ are source and target radial coordinates, and δ_x denotes the Dirac measure at point x.

Let

$$\mathcal{R}_S[1] \le \cdots \le \mathcal{R}_S[B], \qquad \mathcal{R}_T[1] \le \cdots \le \mathcal{R}_T[B]$$

be the sorted source and target radii. In one dimension with cost $|r - r'|$, the optimal transport coupling is the monotone coupling, equivalently the coupling induced by matching equal quantiles. Therefore,

$$W_1(\widehat{P}_S^r, \widehat{P}_T^r) = \int_0^1 \left|F_S^{-1}(q) - F_T^{-1}(q)\right| dq,$$

where F_S^{-1} and F_T^{-1} are the quantile functions of $\widehat{P}_S^r$ and $\widehat{P}_T^r$. For equal-size empirical distributions, these quantile functions are constant on each interval $\big((k-1)/B, k/B\big]$:

$$F_S^{-1}(q) = \mathcal{R}_S[k], \qquad F_T^{-1}(q) = \mathcal{R}_T[k], \qquad q \in \left(\frac{k-1}{B}, \frac{k}{B}\right].$$

Hence,

$$\begin{aligned} W_1(\widehat{P}_S^r, \widehat{P}_T^r) &= \sum_{k=1}^{B} \int_{(k-1)/B}^{k/B} |\mathcal{R}_S[k] - \mathcal{R}_T[k]|\, dq \\ &= \frac{1}{B} \sum_{k=1}^{B} |\mathcal{R}_S[k] - \mathcal{R}_T[k]|\,. \end{aligned}$$

This is exactly the radial loss in Eq. (2)*. The proof is complete.*

E Proof of Theorem 4

Theorem 4 (Angular Margin Preservation) *Let $\{\hat{w}_c\}_{c=1}^K \subset \mathbb{S}^{d-1}$ be normalized class prototypes, and define $h(u) = \arg\max_{c\in[K]} \hat{w}_c^\top u$. For a labeled sample $(\mathbf{z}, y)$, let $u = u(\mathbf{z})$ and let $\tilde{u}$ be the angular coordinate after an endpoint transport map. Define the angular margin*

$$m_y(u) = \hat{w}_y^\top u - \max_{c\neq y} \hat{w}_c^\top u.$$

If $m_y(u) \geq \mu$ and $d_{\mathbb{S}}(u, \tilde{u}) < \frac{\mu}{2}$ for some $\mu > 0$, then $h(\tilde{u}) = y$. For a labeled distribution P,

$$R_{\mathrm{post}}(h) \leq R_{\mathrm{pre}}(h) + \rho_\mu + \frac{2}{\mu}\Delta_{\mathrm{ang}},$$

where $R_{\mathrm{pre}}(h) = \Pr_P[h(u) \neq y]$, $R_{\mathrm{post}}(h) = \Pr_P[h(\tilde{u}) \neq y]$, $\rho_\mu = \Pr_P[0 \leq m_y(u) < \mu]$, and $\Delta_{\mathrm{ang}} = \mathbb{E}_P[d_{\mathbb{S}}(u, \tilde{u})]$.

Proof 4 *We first clarify the notation. Let $[K] = \{1, \ldots, K\}$ be the set of class indices. For each class $c \in [K]$, $\hat{w}_c \in \mathbb{S}^{d-1}$ is a normalized prototype, so*

$$\|\hat{w}_c\|_2 = 1.$$

For a representation $\mathbf{z}$, let $u = u(\mathbf{z}) \in \mathbb{S}^{d-1}$ denote its angular coordinate. Let $\tilde{u} \in \mathbb{S}^{d-1}$ denote the angular coordinate after the endpoint transport map. The classifier is

$$h(u) = \arg\max_{c\in[K]} \hat{w}_c^\top u.$$

We assume a fixed deterministic tie-breaking rule for $\arg\max$*, so that h is well-defined.*

For a labeled sample with label $y \in [K]$, the angular margin is

$$m_y(u) = \hat{w}_y^\top u - \max_{c\neq y} \hat{w}_c^\top u.$$

Equivalently, $m_y(u)$ measures how much larger the logit of the true class y is than the largest competing class logit. In particular, if $m_y(u) > 0$, then y is the unique maximizer and hence $h(u) = y$.

We first prove the pointwise margin-preservation claim. For any competing class $c \neq y$, define the pairwise margin

$$m_{y,c}(u) = \hat{w}_y^\top u - \hat{w}_c^\top u.$$

By definition of $m_y(u)$*,*

$$m_{y,c}(u) \geq m_y(u), \qquad \forall c \neq y.$$

Now compare the same pairwise margin after transport:

$$\begin{aligned} m_{y,c}(\tilde{u}) &= \hat{w}_y^\top \tilde{u} - \hat{w}_c^\top \tilde{u} \\ &= \hat{w}_y^\top u - \hat{w}_c^\top u + \hat{w}_y^\top (\tilde{u} - u) - \hat{w}_c^\top (\tilde{u} - u) \\ &= m_{y,c}(u) + \hat{w}_y^\top (\tilde{u} - u) - \hat{w}_c^\top (\tilde{u} - u). \end{aligned}$$

Using Cauchy's inequality and $\|\hat{w}_y\|_2 = \|\hat{w}_c\|_2 = 1$*, we have*

$$\left|\hat{w}_y^\top (\tilde{u} - u)\right| \leq \|\tilde{u} - u\|_2,$$

and

$$\left|\hat{w}_c^\top (\tilde{u} - u)\right| \leq \|\tilde{u} - u\|_2.$$

Therefore,

$$m_{y,c}(\tilde{u}) \geq m_{y,c}(u) - 2\|\tilde{u} - u\|_2.$$

Since $u, \tilde{u} \in \mathbb{S}^{d-1}$*, their Euclidean chordal distance is bounded by their spherical geodesic distance:*

$$\|\tilde{u} - u\|_2 = 2\sin\left(\frac{d_{\mathbb{S}}(u, \tilde{u})}{2}\right) \leq d_{\mathbb{S}}(u, \tilde{u}).$$

Thus,

$$m_{y,c}(\tilde{u}) \geq m_{y,c}(u) - 2d_{\mathbb{S}}(u, \tilde{u}).$$

Using $m_{y,c}(u) \geq m_y(u)$ *gives*

$$m_{y,c}(\tilde{u}) \geq m_y(u) - 2d_{\mathbb{S}}(u, \tilde{u}).$$

If

$$m_y(u) \geq \mu \qquad \textit{and} \qquad d_{\mathbb{S}}(u, \tilde{u}) < \frac{\mu}{2},$$

then for every $c \neq y$*,*

$$m_{y,c}(\tilde{u}) > \mu - 2 \cdot \frac{\mu}{2} = 0.$$

Hence,

$$\hat{w}_y^\top \tilde{u} > \hat{w}_c^\top \tilde{u}, \qquad \forall c \neq y.$$

Therefore, y *remains the unique maximizer after transport, and*

$$h(\tilde{u}) = y.$$

This proves the pointwise statement.

We now prove the distribution-level risk bound. Let $(\mathbf{z}, y) \sim P$ *be a random labeled sample. Define*

$$U := u(\mathbf{z}), \qquad \tilde{U} := \tilde{u},$$

where U *is the pre-transport angular coordinate and* $\tilde{U}$ *is the post-transport angular coordinate. Define the pre-transport error event*

$$E_{\text{pre}} := \{h(U) \neq y\},$$

and the post-transport error event

$$E_{\text{post}} := \{h(\tilde{U}) \neq y\}.$$

By definition,

$$R_{\text{pre}}(h) = \Pr_P(E_{\text{pre}}), \qquad R_{\text{post}}(h) = \Pr_P(E_{\text{post}}).$$

Next define the small-margin event

$$E_{\text{mar}} := \{0 \leq m_y(U) < \mu\},$$

so that

$$\rho_\mu = \Pr_P(E_{\text{mar}}).$$

Also define the large-angular-drift event

$$E_{\text{drift}} := \left\{ d_{\mathbb{S}}(U, \tilde{U}) \geq \frac{\mu}{2} \right\}.$$

We claim that

$$E_{\text{post}} \subseteq E_{\text{pre}} \cup E_{\text{mar}} \cup E_{\text{drift}}.$$

To prove this inclusion, consider any sample outside the right-hand side. Then

$$h(U) = y,$$

so the true class is predicted correctly before transport. This implies

$$m_y(U) \geq 0,$$

because the true class logit is at least as large as every competing class logit. Since the sample is also outside E_{mar}, it cannot satisfy $0 \leq m_y(U) < \mu$. Therefore,

$$m_y(U) \geq \mu.$$

Moreover, being outside E_{drift} gives

$$d_{\mathbb{S}}(U, \tilde{U}) < \frac{\mu}{2}.$$

By the pointwise result proved above, these two inequalities imply

$$h(\tilde{U}) = y.$$

Thus the sample is not in E_{post}. This proves the event inclusion.

Taking probabilities and applying the union bound gives

$$\begin{aligned} R_{\text{post}}(h) &= \Pr_P(E_{\text{post}}) \\ &\leq \Pr_P(E_{\text{pre}}) + \Pr_P(E_{\text{mar}}) + \Pr_P(E_{\text{drift}}) \\ &= R_{\text{pre}}(h) + \rho_\mu + \Pr_P\left[d_{\mathbb{S}}(U, \tilde{U}) \geq \frac{\mu}{2} \right]. \end{aligned}$$

The random variable

$$X := d_{\mathbb{S}}(U, \tilde{U})$$

is nonnegative. By Markov's inequality,

$$\Pr_P\left[X \geq \frac{\mu}{2} \right] \leq \frac{\mathbb{E}_P[X]}{\mu/2} = \frac{2}{\mu} \mathbb{E}_P[d_{\mathbb{S}}(U, \tilde{U})].$$

By definition,

$$\Delta_{\text{ang}} = \mathbb{E}_P[d_{\mathbb{S}}(U, \tilde{U})].$$

Therefore,

$$\Pr_P\left[d_{\mathbb{S}}(U, \tilde{U}) \geq \frac{\mu}{2} \right] \leq \frac{2}{\mu} \Delta_{\text{ang}}.$$

Substituting this into the previous inequality yields

$$R_{\text{post}}(h) \leq R_{\text{pre}}(h) + \rho_\mu + \frac{2}{\mu} \Delta_{\text{ang}}.$$

This completes the proof.

F Proof of Theorem 5

Theorem 5 (Polar Flow Endpoint Stability) *Fix a coupling π and assume all analytic geodesics and learned flow trajectories stay inside a compact geodesic normal ball $B_o(\rho)$. Let $\gamma_t(z_S, z_T)$ be the analytic geodesic and let φ_t^ψ be the trajectory generated by the learned polar field in Eq.* (4) *with $\varphi_0^\psi = z_S$. Assume the reconstructed manifold vector field is L_v-Lipschitz in z on $B_o(\rho)$. Then there exists a constant C_{geo} depending only on the ball and the metric such that*

$$\mathbb{E}_{(z_S, z_T) \sim \pi}\left[d_{\mathcal{M}_c}(\varphi_1^\psi, z_T) \right] \leq C_{\text{geo}} e^{L_v} \sqrt{\mathcal{L}_{\text{FM}}}.$$

Proof 5 *Let*

$$K := \overline{B_o(\rho)} \subset \mathcal{M}_c$$

denote the compact geodesic normal ball in which all analytic geodesics and learned flow trajectories stay. We write

$$d(\cdot,\cdot) := d_{\mathcal{M}_c}(\cdot,\cdot)$$

for the geodesic distance on $\mathcal{M}_c$. For a point $z \in \mathcal{M}_c$, let g_z be the Riemannian metric at z, and for a tangent vector $\xi \in T_z\mathcal{M}_c$, define

$$\|\xi\|_{g_z} := \sqrt{g_z(\xi,\xi)}.$$

For a coupled source–target pair $(z_S, z_T) \sim \pi$, let

$$\gamma_t = \gamma_t(z_S, z_T), \qquad t \in [0,1],$$

be the analytic constant-speed geodesic from z_S to z_T. Thus,

$$\gamma_0 = z_S, \qquad \gamma_1 = z_T,$$

and its velocity is denoted by

$$\dot\gamma_t \in T_{\gamma_t}\mathcal{M}_c.$$

Let $V_\psi(z,t) \in T_z\mathcal{M}_c$ be the reconstructed manifold vector field induced by the learned polar field in Eq. (4)*. In polar coordinates, if $z = (r,u)$, this field has the form*

$$V_\psi(z,t) = a_\psi(z,t)\partial_r + b_\psi(z,t),$$

where $a_\psi(z,t)$ is the radial velocity component and $b_\psi(z,t) \in T_u\mathbb{S}^{d-1}$ is the angular velocity component. The learned trajectory φ_t^ψ is the solution of the initial value problem

$$\frac{d}{dt}\varphi_t^\psi = V_\psi(\varphi_t^\psi, t), \qquad \varphi_0^\psi = z_S.$$

For each pair (z_S, z_T) and time t, define the pointwise velocity-regression error

$$\varepsilon(t; z_S, z_T) := \|V_\psi(\gamma_t, t) - \dot\gamma_t\|_{g_{\gamma_t}}.$$

This quantity measures, at the analytic geodesic point γ_t, the Riemannian norm of the difference between the learned vector field and the true geodesic velocity.

If $\gamma_t = (r_t, u_t)$ in polar coordinates and

$$\dot\gamma_t = \dot r_t \partial_r + \dot u_t, \qquad \dot u_t \in T_{u_t}\mathbb{S}^{d-1},$$

then the polar metric

$$ds^2 = dr^2 + S_c(r)^2 d\Omega^2$$

gives

$$\varepsilon(t; z_S, z_T)^2 = |a_\psi(\gamma_t,t) - \dot r_t|^2 + S_c(r_t)^2 \|b_\psi(\gamma_t,t) - \dot u_t\|_{\mathbb{S}}^2.$$

Therefore, the polar flow matching loss can be written as

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{(z_S,z_T)\sim\pi,\ t\sim U[0,1]}\left[\varepsilon(t; z_S, z_T)^2\right],$$

where $U[0,1]$ denotes the uniform distribution over $[0,1]$.

We now compare the learned trajectory φ_t^ψ with the analytic geodesic γ_t. Define

$$e(t) := d(\varphi_t^\psi, \gamma_t).$$

Since both curves start from the same source point,

$$e(0) = d(z_S, z_S) = 0.$$

Also, since $\gamma_1 = z_T$,

$$e(1) = d(\varphi_1^\psi, z_T).$$

Let $\mathsf{P}_{\gamma_t \to \varphi_t^\psi}$ denote parallel transport from $T_{\gamma_t}\mathcal{M}_c$ to $T_{\varphi_t^\psi}\mathcal{M}_c$ along the minimizing geodesic inside the normal ball. Parallel transport is an isometry, so for any $\xi \in T_{\gamma_t}\mathcal{M}_c$,

$$\left\| \mathsf{P}_{\gamma_t \to \varphi_t^\psi} \xi \right\|_{g_{\varphi_t^\psi}} = \|\xi\|_{g_{\gamma_t}}.$$

By the first variation inequality for the Riemannian distance, the upper Dini derivative of $e(t)$ satisfies

$$D^+ e(t) \le \left\| V_\psi(\varphi_t^\psi, t) - \mathsf{P}_{\gamma_t \to \varphi_t^\psi} \dot{\gamma}_t \right\|_{g_{\varphi_t^\psi}},$$

where

$$D^+ e(t) := \limsup_{h \downarrow 0} \frac{e(t+h) - e(t)}{h}.$$

Using the triangle inequality, we obtain

$$\begin{aligned} D^+ e(t) \le & \left\| V_\psi(\varphi_t^\psi, t) - \mathsf{P}_{\gamma_t \to \varphi_t^\psi} V_\psi(\gamma_t, t) \right\|_{g_{\varphi_t^\psi}} \\ & + \left\| \mathsf{P}_{\gamma_t \to \varphi_t^\psi} \left(V_\psi(\gamma_t, t) - \dot{\gamma}_t \right) \right\|_{g_{\varphi_t^\psi}}. \end{aligned}$$

The reconstructed vector field is assumed to be L_v-Lipschitz in z on K. In intrinsic Riemannian form, this means

$$\| V_\psi(x,t) - \mathsf{P}_{y \to x} V_\psi(y,t) \|_{g_x} \le L_v d(x,y), \qquad x, y \in K,\ t \in [0,1].$$

Applying this with $x = \varphi_t^\psi$ and $y = \gamma_t$ gives

$$\left\| V_\psi(\varphi_t^\psi, t) - \mathsf{P}_{\gamma_t \to \varphi_t^\psi} V_\psi(\gamma_t, t) \right\|_{g_{\varphi_t^\psi}} \le L_v e(t).$$

For the second term, the isometry of parallel transport gives

$$\begin{aligned} \left\| \mathsf{P}_{\gamma_t \to \varphi_t^\psi} \left(V_\psi(\gamma_t, t) - \dot{\gamma}_t \right) \right\|_{g_{\varphi_t^\psi}} &= \| V_\psi(\gamma_t, t) - \dot{\gamma}_t \|_{g_{\gamma_t}} \\ &= \varepsilon(t; z_S, z_T). \end{aligned}$$

Therefore,

$$D^+ e(t) \le L_v e(t) + \varepsilon(t; z_S, z_T).$$

By Gronwall's inequality and $e(0) = 0$,

$$\begin{aligned} e(1) &\le \int_0^1 e^{L_v(1-t)} \varepsilon(t; z_S, z_T)\, dt \\ &\le e^{L_v} \int_0^1 \varepsilon(t; z_S, z_T)\, dt. \end{aligned}$$

Since $e(1) = d(\varphi_1^\psi, z_T)$, we have the pointwise endpoint bound

$$d(\varphi_1^\psi, z_T) \le e^{L_v} \int_0^1 \varepsilon(t; z_S, z_T)\, dt.$$

Taking expectation over the coupling π yields

$$\begin{aligned} \mathbb{E}_{(z_S, z_T) \sim \pi} \left[d(\varphi_1^\psi, z_T) \right] &\le e^{L_v} \mathbb{E}_{(z_S, z_T) \sim \pi} \left[\int_0^1 \varepsilon(t; z_S, z_T)\, dt \right] \\ &= e^{L_v} \mathbb{E}_{(z_S, z_T) \sim \pi,\ t \sim U[0,1]} \left[\varepsilon(t; z_S, z_T) \right]. \end{aligned}$$

By Jensen's inequality, or equivalently Cauchy–Schwarz,

$$\begin{aligned} \mathbb{E}_{(z_S, z_T) \sim \pi,\ t \sim U[0,1]} \left[\varepsilon(t; z_S, z_T) \right] &\le \left(\mathbb{E}_{(z_S, z_T) \sim \pi,\ t \sim U[0,1]} \left[\varepsilon(t; z_S, z_T)^2 \right] \right)^{1/2} \\ &= \sqrt{\mathcal{L}_{\mathrm{FM}}}. \end{aligned}$$

Combining the previous two inequalities gives

$$\mathbb{E}_{(z_S,z_T)\sim\pi}\left[d_{\mathcal{M}_c}(\varphi_1^{\psi}, z_T)\right] \le e^{L_v}\sqrt{\mathcal{L}_{\mathrm{FM}}}.$$

The above derivation is written in intrinsic Riemannian form, in which the geometric constant can be taken as 1*. If the learned vector field is expressed and analyzed through polar coordinates, the compactness of K and smoothness of the metric on the normal ball imply uniform equivalence between coordinate norms, polar metric norms, and intrinsic Riemannian norms. These equivalence constants depend only on $K = B_o(\rho)$ and the metric g, not on ψ, π, or the sampled pairs. Collecting them into a single constant $C_{\mathrm{geo}} > 0$, we obtain*

$$\mathbb{E}_{(z_S,z_T)\sim\pi}\left[d_{\mathcal{M}_c}(\varphi_1^{\psi}, z_T)\right] \le C_{\mathrm{geo}} e^{L_v}\sqrt{\mathcal{L}_{\mathrm{FM}}}.$$

This proves the theorem.

G Proof of Theorem 6

Theorem 6 (Target-Risk Bound with Polar Discrepancy) *Let P_S and P_T be source and target distributions over graph-label pairs. Assume $\mathbf{z} = f_\theta(G) \in B_o(\rho) \subset \mathcal{M}_c$ and the loss $\ell_\phi(\mathbf{z}, y) = \ell(g_\phi(\mathbf{z}), y) \in [0,1]$ is L_ℓ-Lipschitz in z. Let Φ_ψ be the endpoint map induced by the learned polar flow and let Ψ be an ideal class-conditioned transport map. For each class y, let Γ_y be a lifted coupling over source–target graph pairs (G, G') such that the induced representation pair $\mathbf{z} = \Psi(f_\theta(G))$ and $\mathbf{z}' = f_\theta(G')$ has marginals $\Psi_{\#}P_S(\mathbf{z} \mid y)$ and $P_T(\mathbf{z} \mid y)$, respectively. Define*

$$D_{\mathrm{rad}}^{\Gamma} = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y} |r(\Psi(f_\theta(G))) - r(f_\theta(G'))|,\ D_{\mathrm{top}}^{\Gamma} = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y} \|q(G) - q(G')\|_2,$$

$$D_{\mathrm{ang}}^{\Gamma} = \sum_y \omega_y \mathbb{E}_{(G,G')\sim\Gamma_y} d_{\mathbb{S}}(u(\Psi(f_\theta(G))), u(f_\theta(G'))),$$

where $\omega_y = P_T(Y = y)$. Let $C_\rho = \sup_{0\le t\le\rho} S_c(t)$ and let $\lambda_{\mathrm{lab}} = \frac{1}{2}\sum_y |P_S(Y = y) - P_T(Y = y)|$. If pseudo-labels used to construct the target class-conditionals have error ϵ_{pl}, then there exist constants C_{top}, C_{FM}, and C_{pl} such that

$$R_T(g_\phi) \le R_S(g_\phi \circ \Phi_\psi) + L_\ell\Big(D_{\mathrm{rad}}^{\Gamma} + C_\rho D_{\mathrm{ang}}^{\Gamma} + C_{\mathrm{top}} D_{\mathrm{top}}^{\Gamma} + C_{\mathrm{FM}}\sqrt{\mathcal{L}_{\mathrm{FM}}}\Big) + C_{\mathrm{pl}}\epsilon_{\mathrm{pl}} + \lambda_{\mathrm{lab}}.$$

Proof 6 *We interpret Γ_y as a lifted coupling over graph pairs whose induced representation marginals are $\Psi_{\#}P_S(\mathbf{z} \mid y)$ and $P_T(\mathbf{z} \mid y)$. Let $\mathcal{Y} = [K]$ be the label space. For a graph-label pair $(G, Y) \sim P_D$, $D \in \{S, T\}$, define*

$$Z = f_\theta(G) \in B_o(\rho) \subset \mathcal{M}_c.$$

For each domain $D \in \{S, T\}$ and class $y \in \mathcal{Y}$, write

$$p_D(y) := P_D(Y = y), \qquad P_D^y := P_D(Z \mid Y = y).$$

Thus $p_D(y)$ is the class prior in domain D, and P_D^y is the class-conditional distribution of the representation Z.

The target risk of g_ϕ is

$$R_T(g_\phi) = \mathbb{E}_{(G,Y)\sim P_T}\big[\ell_\phi(f_\theta(G), Y)\big],$$

where

$$\ell_\phi(\mathbf{z}, y) = \ell(g_\phi(\mathbf{z}), y) \in [0,1].$$

The transported source risk induced by the learned endpoint map Φ_ψ is

$$R_S(g_\phi \circ \Phi_\psi) = \mathbb{E}_{(G,Y)\sim P_S}\big[\ell_\phi(\Phi_\psi(f_\theta(G)), Y)\big].$$

By assumption, for every label y and every $\mathbf{z}, \mathbf{z}' \in B_o(\rho)$,

$$|\ell_\phi(\mathbf{z}, y) - \ell_\phi(\mathbf{z}', y)| \le L_\ell d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}').$$

For each class y, let Γ_y be a lifted coupling between source and target graph-representation pairs such that the first representation marginal is $\Psi_\# P_S^y$ and the second representation marginal is P_T^y. Equivalently, under $(G, G') \sim \Gamma_y$, the representation pair is

$$\mathbf{z} = \Psi(f_\theta(G)), \qquad \mathbf{z}' = f_\theta(G'),$$

where $\mathbf{z}$ follows the ideal transported source class-conditional distribution and $\mathbf{z}'$ follows the target class-conditional distribution. The lifted form is used only so that topology statistics $q(G)$ and $q(G')$ are well-defined.

We first decompose the target risk by class:

$$\begin{aligned} R_T(g_\phi) &= \sum_{y \in \mathcal{Y}} P_T(Y = y) \mathbb{E}_{\mathbf{z}' \sim P_T^y} \big[\ell_\phi(\mathbf{z}', y)\big] \\ &= \sum_{y \in \mathcal{Y}} \omega_y \mathbb{E}_{(\mathbf{z}, \mathbf{z}') \sim \Gamma_y} \big[\ell_\phi(\mathbf{z}', y)\big], \end{aligned}$$

where

$$\omega_y = P_T(Y = y).$$

Using the Lipschitz continuity of ℓ_ϕ in its representation argument, we obtain

$$\ell_\phi(\mathbf{z}', y) \le \ell_\phi(\mathbf{z}, y) + L_\ell d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}').$$

Therefore,

$$\begin{aligned} R_T(g_\phi) \le &\sum_y \omega_y \mathbb{E}_{(\mathbf{z}, \mathbf{z}') \sim \Gamma_y} \big[\ell_\phi(\mathbf{z}, y)\big] \\ &+ L_\ell \sum_y \omega_y \mathbb{E}_{(\mathbf{z}, \mathbf{z}') \sim \Gamma_y} \big[d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}')\big]. \end{aligned}$$

Define the target-prior-weighted ideal transported source risk as

$$\bar{R}_S^\Psi(g_\phi) := \sum_y \omega_y \mathbb{E}_{\mathbf{z} \sim \Psi_\# P_S^y} \big[\ell_\phi(\mathbf{z}, y)\big].$$

Since the first marginal of Γ_y is $\Psi_\# P_S^y$, the first term above is exactly $\bar{R}_S^\Psi(g_\phi)$. Hence,

$$R_T(g_\phi) \le \bar{R}_S^\Psi(g_\phi) + L_\ell \sum_y \omega_y \mathbb{E}_{\Gamma_y} \big[d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}')\big].$$

We now upper bound the Riemannian distance by polar discrepancies. Write

$$\mathbf{z} = (r(\mathbf{z}), u(\mathbf{z})), \qquad \mathbf{z}' = (r(\mathbf{z}'), u(\mathbf{z}'))$$

in geodesic polar coordinates around the origin o. Since both points lie in $B_o(\rho)$, we have

$$0 \le r(\mathbf{z}), r(\mathbf{z}') \le \rho.$$

Let

$$C_\rho = \sup_{0 \le t \le \rho} S_c(t).$$

Because $B_o(\rho)$ is compact and S_c is continuous, $C_\rho < \infty$.

For any two non-origin points $\mathbf{z}, \mathbf{z}' \in B_o(\rho)$, construct a piecewise smooth path from $\mathbf{z}$ to $\mathbf{z}'$ as follows. First move radially from radius $r(\mathbf{z})$ to radius $r(\mathbf{z}')$ while keeping the angular direction fixed at $u(\mathbf{z})$. This segment has length

$$|r(\mathbf{z}) - r(\mathbf{z}')|.$$

Then move angularly on the geodesic sphere of radius $r(\mathbf{z}')$ from $u(\mathbf{z})$ to $u(\mathbf{z}')$. Under the polar metric

$$ds^2 = dr^2 + S_c(r)^2 d\Omega^2,$$

this angular segment has length

$$S_c(r(\mathbf{z}')) d_{\mathbb{S}}(u(\mathbf{z}), u(\mathbf{z}')) \le C_\rho d_{\mathbb{S}}(u(\mathbf{z}), u(\mathbf{z}')).$$

The geodesic distance is no larger than the length of any admissible path, so

$$d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}') \le |r(\mathbf{z}) - r(\mathbf{z}')| + C_\rho d_{\mathbb{S}}(u(\mathbf{z}), u(\mathbf{z}')).$$

If one of the two points is the origin, the same inequality holds by continuity; the angular coordinate at the origin can be chosen arbitrarily because $S_c(0) = 0$.

Since

$$\|q(G) - q(G')\|_2 \ge 0,$$

for any finite constant $C_{\mathrm{top}} \ge 0$ we also have the looser topology-augmented bound

$$\begin{aligned} d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}') \le{}& |r(\mathbf{z}) - r(\mathbf{z}')| + C_\rho d_{\mathbb{S}}(u(\mathbf{z}), u(\mathbf{z}')) \\ &+ C_{\mathrm{top}} \|q(G) - q(G')\|_2. \end{aligned}$$

Here $q(G)$ denotes the chosen topology descriptor of graph G, and C_{top} is a finite constant attached to this topology discrepancy. In the purely polar metric bound one may take $C_{\mathrm{top}} = 0$; retaining a positive C_{top} gives the stated topology-conditioned form.

Taking expectation under each Γ_y and summing with weights ω_y gives

$$\begin{aligned} \sum_y \omega_y \mathbb{E}_{\Gamma_y}\left[d_{\mathcal{M}_c}(\mathbf{z}, \mathbf{z}')\right] \le{}& \sum_y \omega_y \mathbb{E}_{\Gamma_y}\left[|r(\mathbf{z}) - r(\mathbf{z}')|\right] \\ &+ C_\rho \sum_y \omega_y \mathbb{E}_{\Gamma_y}\left[d_{\mathbb{S}}(u(\mathbf{z}), u(\mathbf{z}'))\right] \\ &+ C_{\mathrm{top}} \sum_y \omega_y \mathbb{E}_{\Gamma_y}\left[\|q(G) - q(G')\|_2\right]. \end{aligned}$$

By the definitions in the theorem, this is

$$D^{\Gamma}_{\mathrm{rad}} + C_\rho D^{\Gamma}_{\mathrm{ang}} + C_{\mathrm{top}} D^{\Gamma}_{\mathrm{top}}.$$

Thus,

$$R_T(g_\phi) \le \bar{R}^{\Psi}_S(g_\phi) + L_\ell \left(D^{\Gamma}_{\mathrm{rad}} + C_\rho D^{\Gamma}_{\mathrm{ang}} + C_{\mathrm{top}} D^{\Gamma}_{\mathrm{top}}\right).$$

It remains to compare the target-prior-weighted ideal transported source risk $\bar{R}^{\Psi}_S(g_\phi)$ with the actual source risk after the learned endpoint map Φ_ψ.

First, compare $\bar{R}^{\Psi}_S(g_\phi)$ with the source-prior-weighted ideal transported source risk

$$R_S(g_\phi \circ \Psi) = \mathbb{E}_{(G,Y)\sim P_S}\left[\ell_\phi(\Psi(f_\theta(G)), Y)\right].$$

For each class y, define

$$a_y := \mathbb{E}_{\mathbf{z}\sim P^y_S}\left[\ell_\phi(\Psi(\mathbf{z}), y)\right].$$

Since $\ell_\phi \in [0, 1]$, we have

$$0 \le a_y \le 1.$$

Then

$$\bar{R}^{\Psi}_S(g_\phi) = \sum_y \omega_y a_y,$$

whereas

$$R_S(g_\phi \circ \Psi) = \sum_y P_S(Y = y) a_y.$$

Therefore,

$$\bar{R}^{\Psi}_S(g_\phi) - R_S(g_\phi \circ \Psi) = \sum_y \left(P_T(Y = y) - P_S(Y = y)\right) a_y.$$

Let

$$\Delta_y = P_T(Y = y) - P_S(Y = y).$$

Since both $P_T(Y = \cdot)$ and $P_S(Y = \cdot)$ are probability distributions,

$$\sum_y \Delta_y = 0.$$

Using $0 \le a_y \le 1$, we have

$$\sum_y \Delta_y a_y \le \sum_{y:\Delta_y>0} \Delta_y = \frac{1}{2}\sum_y |\Delta_y| = \lambda_{\mathrm{lab}}.$$

Hence,

$$\bar{R}_S^{\Psi}(g_\phi) \le R_S(g_\phi \circ \Psi) + \lambda_{\mathrm{lab}}.$$

Next, compare the ideal endpoint map Ψ with the learned endpoint map Φ_ψ. By Lipschitz continuity of the loss,

$$\begin{aligned} R_S(g_\phi \circ \Psi) &= \mathbb{E}_{(G,Y)\sim P_S}\big[\ell_\phi(\Psi(f_\theta(G)), Y)\big] \\ &\le \mathbb{E}_{(G,Y)\sim P_S}\big[\ell_\phi(\Phi_\psi(f_\theta(G)), Y)\big] \\ &\quad + L_\ell \mathbb{E}_{(G,Y)\sim P_S}\big[d_{\mathcal{M}_c}(\Psi(f_\theta(G)), \Phi_\psi(f_\theta(G)))\big] \\ &= R_S(g_\phi \circ \Phi_\psi) + L_\ell \mathbb{E}_{(G,Y)\sim P_S}\big[d_{\mathcal{M}_c}(\Psi(Z), \Phi_\psi(Z))\big]. \end{aligned}$$

Let π_Ψ denote the population coupling that pairs each source representation Z with its ideal transported endpoint $\Psi(Z)$. Applying the polar flow endpoint stability result to this coupling gives

$$\mathbb{E}_{Z\sim P_S}\big[d_{\mathcal{M}_c}(\Phi_\psi(Z), \Psi(Z))\big] \le C_{\mathrm{FM}}\sqrt{\mathcal{L}_{\mathrm{FM}}},$$

where C_{FM} is a finite constant depending on the compact normal ball, the metric, and the Lipschitz constant of the reconstructed vector field. In particular, under Theorem 5, one may take

$$C_{\mathrm{FM}} = C_{\mathrm{geo}} e^{L_v}.$$

Therefore,

$$R_S(g_\phi \circ \Psi) \le R_S(g_\phi \circ \Phi_\psi) + L_\ell C_{\mathrm{FM}}\sqrt{\mathcal{L}_{\mathrm{FM}}}.$$

Combining the previous inequalities yields the bound under true target class-conditionals:

$$R_T(g_\phi) \le R_S(g_\phi \circ \Phi_\psi) + L_\ell\left(D_{\mathrm{rad}}^{\Gamma} + C_\rho D_{\mathrm{ang}}^{\Gamma} + C_{\mathrm{top}} D_{\mathrm{top}}^{\Gamma} + C_{\mathrm{FM}}\sqrt{\mathcal{L}_{\mathrm{FM}}}\right) + \lambda_{\mathrm{lab}}.$$

Finally, we account for pseudo-label noise. Let $\widehat{Y}$ denote the pseudo-label used to construct target class-conditionals, and let $\widehat{P}_T$ denote the pseudo-labeled target distribution over $(G, \widehat{Y})$. Suppose the pseudo-label error is

$$\epsilon_{\mathrm{pl}} = \Pr_{(G,Y)\sim P_T}(\widehat{Y} \ne Y).$$

Under the canonical coupling that pairs (G, Y) with $(G, \widehat{Y})$, the two joint distributions differ only when $\widehat{Y} \ne Y$. Hence their total variation distance is bounded by

$$\mathrm{TV}\big(P_T(G,Y), \widehat{P}_T(G,\widehat{Y})\big) \le \epsilon_{\mathrm{pl}}.$$

Equivalently, for the class-conditional construction, assume the aggregate pseudo-label perturbation satisfies

$$\sum_y \omega_y \mathrm{TV}\big(P_T(Z \mid Y=y), \widehat{P}_T(Z \mid \widehat{Y}=y)\big) \le C_{\mathrm{cond}}\epsilon_{\mathrm{pl}},$$

where C_{cond} is finite when retained target classes have non-vanishing class mass. This is the standard conditioning cost caused by replacing true target labels with pseudo-labels.

Because $B_o(\rho)$ is compact, the manifold distance is bounded on the representation support. Moreover, on the considered graph support, the topology descriptor discrepancy is assumed to have finite bound. Therefore, every loss/discrepancy term appearing in the preceding coupling argument is bounded by a finite constant. Thus, by the variational characterization of total variation, replacing true target class-conditionals by pseudo-labeled class-conditionals changes the bound by at most

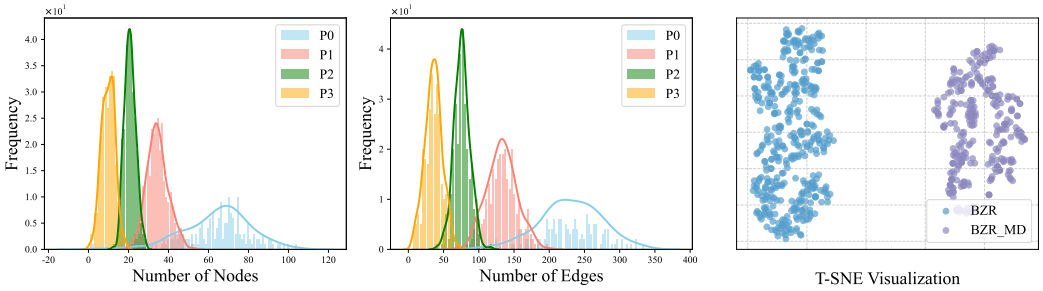
$$C_{\mathrm{pl}}\epsilon_{\mathrm{pl}},$$

for some finite constant C_{pl} depending on the conditioning constant, the compact ball, the loss bound, and the bounded topology statistics.

Adding this perturbation term gives

$$\begin{aligned} R_T(g_\phi) \le R_S(g_\phi \circ \Phi_\psi) &+ L_\ell\left(D_{\mathrm{rad}}^{\Gamma} + C_\rho D_{\mathrm{ang}}^{\Gamma} + C_{\mathrm{top}} D_{\mathrm{top}}^{\Gamma} + C_{\mathrm{FM}}\sqrt{\mathcal{L}_{\mathrm{FM}}}\right) \\ &+ C_{\mathrm{pl}}\epsilon_{\mathrm{pl}} + \lambda_{\mathrm{lab}}. \end{aligned}$$

This is the desired result.



(a) Node Distribution (b) Edge Distribution (c) Feature Distribution

Figure 5: Visualization of domain shifts across different types. (a) Node distribution shift between sub-datasets of PROTEINS. (b) Edge distribution shift between sub-datasets of PROTEINS. (c) Feature distribution shift between BZR and BZR_MD datasets.

H Dataset

H.1 Dataset Description

Table 3: Statistics of the experimental datasets.

Datasets	Graphs	Avg. Nodes	Avg. Edges	Classes
NCI1	4,110	29.87	32.30	2
MUTAGENICITY	4,337	30.32	30.77	2
PROTEINS	1,113	39.1	72.8	2
ogbg-molhiv	41,127	25.5	27.5	2
DD	1,178	284.32	715.66	2
COX2	467	41.22	43.45	2
COX2_MD	303	26.28	335.12	2
BZR	405	35.75	38.36	2
BZR_MD	306	21.30	225.06	2

We conduct extensive experiments on a variety of datasets. The statistics of the datasets are summarized in Table 3. More detailed descriptions of each dataset are provided as follows:

- For structure-based domain shifts:
  - **PROTEINS.** The PROTEINS dataset [12] consists of 1,113 protein graphs, each annotated with a binary label indicating whether the protein is an enzyme. In each graph, nodes correspond to amino acids, and edges connect amino acids that are within 6 Å of each other along the sequence. We further partition the dataset into four subsets, denoted as P0, P1, P2, and P3, according to edge density and node density.
  - **NCI1.** The NCI1 dataset [50] consists of 4,110 molecular graphs, where nodes represent atoms and edges correspond to chemical bonds. Each graph is annotated with a binary label indicating whether the molecule inhibits cancer cell growth. We further partition the dataset into four subsets, denoted as N0, N1, N2, and N3, according to edge density and node density.
  - **Mutagenicity.** The Mutagenicity dataset [23] consists of 4,337 molecular graphs, where nodes represent atoms and edges correspond to chemical bonds. Each graph is annotated with a binary label indicating whether the compound is mutagenic. Following the PROTEINS dataset, we further partition the dataset into four subsets, denoted as M0, M1, M2, and M3, according to edge density and node density.
  - **ogbg-molhiv.** The ogbg-molhiv dataset [21] consists of 41,127 molecular graphs, where nodes represent atoms and edges correspond to chemical bonds. Each graph is annotated with a binary label indicating whether the molecule exhibits HIV inhibitory activity. Following the PROTEINS dataset, we further partition the dataset into four subsets, denoted as H0, H1, H2, and H3, according to edge density and node density.

- For feature-based domain shifts:
  - **DD.** The DD dataset [12] consists of 1,178 protein structure graphs, where nodes correspond to amino acids and edges encode spatial or chemical proximity between them. Compared to the PROTEINS dataset, DD graphs are substantially larger and denser, thereby introducing pronounced structural variability while preserving similar semantic labels.
  - **COX2.** The COX2 dataset [48] consists of 467 molecular graphs, while COX2_MD contains 303 modified molecular graphs. In both datasets, nodes correspond to atoms and edges represent chemical bonds. COX2_MD introduces controlled structural variations to COX2 while preserving semantic labels.
  - **BZR.** The BZR dataset [48] consists of 405 molecular graphs, while BZR_MD contains 306 structurally modified graphs derived from BZR. In both datasets, nodes correspond to atoms and edges represent chemical bonds. BZR_MD introduces controlled structural variations to simulate domain shifts while preserving consistent label semantics.

H.2 Data Processing

For datasets from the TUDataset [2] (e.g., PROTEINS and NCI1), we adopt the standard preprocessing and normalization procedures provided by PyTorch Geometric [3] . For datasets from the Open Graph Benchmark (OGB) [4] , such as ogbg-molhiv, we follow the official OGB preprocessing and normalization protocols.

I Baselines

In this section, we introduce the details of the compared baselines as follows:

- **Graph kernel methods.** We compare DisRFM with two graph kernel methods:
  - **WL subtree**: WL subtree [42] is a graph kernel method that computes graph similarity via a kernel function, encoding local neighborhood structures into subtree patterns and efficiently capturing topological information within graphs.
  - **PathNN**: PathNN [34] is an expressive graph neural network that models graphs by aggregating information along simple paths, enabling accurate capture of long-range dependencies and higher-order structural patterns beyond local neighborhoods.
- **General Graph Neural Networks.** We compare DisRFM with four general graph neural networks:
  - **GCN**: GCN [24] is a message-passing graph neural network that updates node representations by aggregating and normalizing features from immediate neighbors, effectively capturing local structural and attribute information in graphs.
  - **GIN**: GIN [54] is a graph neural network with an injective aggregation function that sums neighbor features, achieving strong expressive power equivalent to the Weisfeiler–Lehman test for distinguishing graph structures.
  - **CIN**: CIN [3] is a higher-order graph neural network inspired by the Weisfeiler–Lehman framework, which operates on cellular complexes to capture rich topological structures beyond pairwise node interactions.
  - **GMT**: GMT [2] is a graph neural network architecture that learns graph-level representations via hierarchical multiset pooling, enabling adaptive aggregation of node features and improving expressiveness for capturing complex graph structures.
- **Manifold-based Graph Neural Networks**: We compare DisRFM with four manifold-based general graph neural networks:
  - **dDGM**: dDGM [4] is a latent graph inference framework that models graph structure in continuous product manifolds, enabling the learning of expressive graph representations by jointly inferring latent relational structure and node embeddings.

[2] https://chrsmrrs.github.io/datasets/
[3] https://pyg.org/
[4] https://ogb.stanford.edu/

  - **RieGrace**: RieGrace [46] is a self-supervised continual graph learning approach that adapts Riemannian representation spaces over time, enabling robust knowledge accumulation and transfer across evolving graph distributions.
  - **ProGDM**: ProGDM [51] is a graph diffusion model that integrates mixed-curvature geometric spaces, enabling effective information propagation and representation learning across graphs with heterogeneous structural properties.
  - **D-GCN**: D-GCN [45] is a motif-aware Riemannian graph neural network that leverages generative–contrastive learning to capture higher-order structural patterns and geometric relationships in graph representations.
- **Graph Domain Adaptation methods.** We compare DisRFM with seven graph domain adaptation methods:
  - **DEAL**: DEAL [56] addresses unsupervised domain adaptation for graph-level classification by learning domain-invariant graph representations through adversarial training, where a shared graph encoder is optimized to confuse a domain discriminator while preserving discriminative power on source labels.
  - **SGDA**: SGDA [37] performs semi-supervised domain adaptation for graph transfer learning by jointly aligning feature representations and structural distributions across source and target domains, enabling effective knowledge transfer under limited target supervision.
  - **StruRW**: StruRW [29] improves graph domain adaptation by re-weighting structural components to mitigate distributional shifts and enhance cross-domain representation alignment.
  - **A2GNN**: A2GNN [27] rethinks information propagation for unsupervised graph domain adaptation by redesigning message-passing mechanisms to better align cross-domain graph representations.
  - **PA-BOTH**: PA-BOTH [30] enhances graph domain adaptation through pairwise alignment of node representations and structural patterns across domains, effectively reducing distribution shifts and facilitating robust knowledge transfer between source and target graphs.
  - **GAA**: GAA [13] investigates attribute-driven graph domain adaptation by leveraging node attributes to guide representation alignment across domains, demonstrating that attribute information plays a critical role in mitigating domain shifts and improving transfer performance.
  - **TDSS**: TDSS [9] emphasizes graph smoothness as a key inductive bias for unsupervised graph domain adaptation, enforcing consistent representations among neighboring nodes to effectively reduce domain discrepancy and improve cross-domain generalization.
- **Manifold-based Domain Adaptation methods.** We compare DisRFM with three manifold-based domain adaptation methods:
  - **GOTDA**: GOTDA [31] formulates domain adaptation as an optimal transport problem on Grassmann manifolds, aligning subspace representations across domains to reduce distributional discrepancy and enable effective knowledge transfer.
  - **MASH**: MASH [38] performs graph integration for diffusion-based manifold alignment by leveraging diffusion processes to align manifold structures across domains, facilitating coherent representation learning and cross-domain knowledge transfer.
  - **GeoAdapt**: GeoAdapt [18] performs domain adaptation by aligning geometric moments via Siegel embeddings, enabling principled matching of higher-order distributional statistics across domains in a structured geometric space.

I.1 Implementation Details

Implementation Details. We implement DisRFM and all baselines in PyTorch and conduct all experiments on NVIDIA A100 GPUs. For the encoder, DisRFM adopts HGCN [28] as the backbone, enabling adaptive representation learning across hyperbolic, Euclidean, and spherical spaces by modulating the curvature parameter c. All experiments use identical hyperparameter settings, with Adam, a learning rate of 1×10^{-4}, weight decay of 1×10^{-12}, hidden dimension 128, and three

GNN layers. We set the manifold curvature to $c = -1.0$, the confidence threshold to $\zeta = 0.7$, and $\lambda_1 = \lambda_2 = \lambda_3 = 0.1$. Following prior settings [52, 56], DisRFM is trained on labeled source data and evaluated on unlabeled target data. We report accuracy on TUDataset benchmarks (e.g., PROTEINS) and AUC on OGB datasets (e.g., ogbg-molhiv), averaged over five independent runs.

J Complexity Analysis

Here we analyze the computational complexity of the proposed DisRFM. The computational complexity is dominated by the Riemannian graph convolution and manifold flow matching. For a given batch of graphs, $|\mathcal{E}|$ denotes the total number of edges. d is the embedding dimension. L denotes the number of the HGCN. $|\mathcal{V}|$ is the total number of nodes. B represents the batch size. The HGCN takes $\mathcal{O}\left(L \cdot \left(|\mathcal{E}| \cdot d + |\mathcal{V}| \cdot d^2\right)\right)$ time complexity, while the flow matching and geometric constraints introduce an additional term of $\mathcal{O}\left(B^2 \cdot d\right)$ for coupling and alignment. As a result, the overall computational complexity of DisRFM is $\mathcal{O}\left(L \cdot \left(|\mathcal{E}| \cdot d + |\mathcal{V}| \cdot d^2\right) + B^2 \cdot d\right)$.

K More experimental results

K.1 More performance comparison

In this section, we present additional experimental results comparing the proposed DisRFM with all baseline models across various datasets, as reported in Tables 17–24. We further observe that DisRFM consistently outperforms baselines in most cases, further demonstrating its effectiveness and robustness.

Additionally, we conduct experiments to evaluate the flexibility of DisRFM. Specifically, under the HGCN geometric framework [28], we replace the tangent space aggregation module with different GNN architectures (i.e., GCN, SAGE, GAT, and GIN), and report the results in Figure 7. The results show that GIN consistently outperforms other GNN architectures across most settings, indicating its stronger representational capacity. This observation further supports our choice of GIN as the default message passing mechanism to enhance the overall performance of DisRFM.

K.2 More Ablation Study

To validate the effectiveness of each component in DisRFM, we further conduct ablation studies on the PROTEINS, NCI1, and ogbg-molhiv datasets. Specifically, we evaluate four variants of DisRFM, including DisRFM w/o FM, DisRFM w/o RA, DisRFM w/o AA, and DisRFM w/o PE. The experimental results are reported in Tables 14, 15, and 16. From these results, we observe trends consistent with those summarized in Section 4.2.

K.3 More Sensitivity Analysis

In this section, we first investigate the sensitivity of the proposed DisRFM to the angular confidence threshold ζ on the Mutagenicity, PROTEINS, NCI1, and ogbg-molhiv datasets. The threshold ζ governs the reliability of target pseudo-labels by excluding ambiguous samples during angular alignment. Figure 8 reports the performance of DisRFM as ζ varies over $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. As shown in Figure 8, the performance consistently improves as ζ increases from 0.5 to 0.7, followed by a mild degradation at higher values. This behavior suggests that a low threshold allows noisy target samples to participate in angular alignment, leading to negative transfer, whereas an overly strict threshold excludes informative samples and weakens semantic guidance. Based on this trade-off, we set $\zeta = 0.7$ as the default value. We further analyze the sensitivity of DisRFM to the curvature parameter c on the PROTEINS, NCI1, and ogbg-molhiv datasets. The results, summarized in Figure 9, exhibit trends consistent with the analysis presented in Section 4.2.

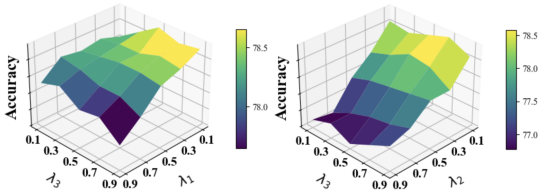


Figure 6: The model performance w.r.t. different combinations of λ_1, λ_2 and λ_3.

Table 4: Training time per epoch of different graph domain adaptation methods (in seconds).

Methods	PROTEINS	NCI1	Mutagenicity	ogbg-molhiv
DEAL	0.176	0.528	0.598	1.347
SGDA	0.072	0.247	0.260	1.073
GAA	0.061	0.194	0.200	1.101
TDSS	0.057	0.311	0.169	1.014
DisRFM	0.091	0.477	0.511	1.737

Table 5: GPU memory consumption of different graph domain adaptation methods during training (in GB).

	PROTEINS	NCI1	Mutagenicity	ogbg-molhiv
DEAL	1.6	1.3	1.4	2.3
SGDA	1.5	1.5	1.3	1.9
GAA	1.8	1.4	1.7	2.7
TDSS	1.4	1.2	1.4	1.8
DisRFM	1.6	1.3	1.5	2.0

Additionally, we provide sensitivity analyses of the proposed DisRFM with respect to balance coefficients (λ_1, λ_3) and (λ_2, λ_3) on the Mutagenicity dataset. The results are illustrated in Figure 6, from which we observe trends consistent with those discussed in Section 4.2.

K.4 Training Time and Memory Comparison

Tables 4 and 5 present a comprehensive comparison of training time per epoch and GPU memory consumption among DisRFM and existing graph domain adaptation methods under identical settings. The results indicate that DisRFM incurs moderate computational cost and memory overhead across multiple datasets.

K.5 More Operational Disentanglement Analysis

A key empirical question in DisRFM is whether the learned polar coordinates provide an operational separation between structure-related and label-predictive signals. Specifically, we examine whether the radial coordinate r serves as a compact proxy for graph-size statistics, while the angular coordinate u carries most class-discriminative information under the trained representation. We emphasize that this analysis does not assume a globally identifiable topology-semantics factorization. Instead, it evaluates whether the intended radial-angular bias is reflected in probe performance, permutation ablations, and latent-space interventions. We conduct three diagnostic analyses to assess this behavior.

K.5.1 Operational Evidence for Graph-Structure Signals in Radial Coordinates

Structural Correlation Analysis. We examine whether the radial coordinate r is associated with simple graph-size statistics, including the number of nodes and edges. As shown in Table 6, r exhibits substantial positive correlations with both node count and edge count, suggesting that the learned radius carries information about graph complexity under our representation geometry. This pattern is consistent with the intended radial inductive bias, where radial distance provides a scale-sensitive coordinate for representing structural variation. To reduce the concern that this correlation is explained only by class labels, we further compute partial correlations after controlling for the ground-truth label y. The coefficients remain close to the unconditional correlations, suggesting that r retains graph-size information beyond what is explained by the label under this linear control.

Structural Prediction Probe. To test whether structural information is more linearly accessible from r than from u, we freeze the encoder and train a linear Ridge regression probe to predict the number of edges. As a baseline, we use tangent-space representations, where standard Euclidean vectors are obtained via the logarithmic map without explicit polar factorization. As shown in Table 7, using only the scalar r achieves a moderate R^2, indicating that the radius captures a nontrivial amount of edge-count information. In contrast, the angular component u alone yields a much lower R^2, suggesting that this particular structural statistic is less linearly accessible from the angular direction.

Table 6: Correlation between the radial coordinate r and graph complexity. Partial Corr indicates the Pearson correlation after controlling for the class label y.

Structure $a(G)$	**Spearman**	**Pearson**	**Partial Corr**
Number of Nodes ($\|V\|$)	0.5783	0.5366	0.5227
Number of Edges ($\|E\|$)	0.5773	0.6076	0.5949

Table 7: Experimental results of structural prediction probe.

Input Features	R^2 (↑)	**MAE** (↓)
r only (1-dim)	0.4454	8.7791
u only (128-dim)	0.0812	12.3421
$r + u$ (Combined)	**0.7072**	**6.8961**
Tangent Space	0.5113	7.7992

Combining r and u further improves prediction performance and outperforms the tangent-space baseline in this probe. These results support an operational separation in which r provides a compact structure-related signal, while u contributes complementary information without concentrating the edge-count signal.

K.5.2 Operational Evidence for Class Semantics in Angular Coordinates

Semantic Probe Analysis. To assess whether label-predictive information is primarily accessible from the angular component, we evaluate the standalone classification performance of each decoupled factor. As shown in Table 8, using only the radial coordinate (r) yields accuracy only modestly above random guessing for this binary task, indicating that r contains limited label-predictive information compared with the angular component. In contrast, the angular component (u) alone achieves performance close to that of the full representation ($r + u$). This suggests that class-discriminative information is predominantly encoded in u, while incorporating r provides only a small additional benefit for classification in this setting.

Feature Permutation Ablation. To further evaluate the role of each factor, we conduct a feature permutation experiment on the test set using a classifier trained on the full ($r + u$) representation, as shown in Table 8. When the radial values are randomly shuffled across samples while preserving their original angular directions (Shuffled r + Orig u), performance remains close to the original result, suggesting that the classifier is not strongly sensitive to sample-specific radial values. In contrast, when the angular directions are shuffled while keeping the true radial values (Orig r + Shuffled u), accuracy drops sharply to near-random chance. This contrast indicates that semantic predictions in this experiment depend much more strongly on u than on r, supporting the intended operational separation between radial and angular factors.

K.5.3 Latent-Space Intervention Analysis

Radial Intervention. We perform a latent-space intervention by scaling the radial coordinate of a target sample ($r' = \text{scale} \cdot r$) while fixing its angular direction (u). The intervened tangent representation $v'_t = r' \cdot u$ is then evaluated using both the structural probe and the semantic classifier. As shown in Table 9, increasing the scaling factor from 0.5 to 2.0 induces a monotonic increase in the predicted structural score. In contrast, the semantic classification probabilities remain nearly unchanged within the reported precision. This result suggests that, for the selected samples and probes, radial changes mainly affect the structure-related predictor while having limited influence on the semantic classifier.

Angular Intervention. We perform the inverse intervention by fixing the radial coordinate r while smoothly interpolating the angular component u toward a sample from a different class. Specifically, we construct counterfactual tangent representations $v' = r_A \cdot u_{\text{interp}}$, where u_{interp} transitions from a Class 0 sample to a Class 1 sample. As shown in Table 10, interpolating u induces a substantial semantic shift and eventually flips the classifier's prediction from Class 0 to Class 1. Meanwhile, the predicted structural score remains stable within the reported precision. Together with the radial

Table 8: Experimental results of semantic probe analysis and feature permutation.

Input Combination	Source Acc	Target Acc
r only	59.41%	63.19%
u only	91.79%	78.41%
$r + u$	**92.39%**	**78.71%**
Shuffled r + Orig u	90.77%	77.77%
Orig r + Shuffled u	50.83%	53.04%

Table 9: Counterfactual radial intervention.

Sample	Scale	New Radius (r')	Pred Nodes (Structure)	Class Prob
Graph #0	0.5	0.7019	30.77	Class 1: 68.52%
Graph #0	1.0	1.4039	31.40	Class 1: 68.52% (Original)
Graph #0	1.5	2.1058	32.04	Class 1: 68.52%
Graph #0	2.0	2.8078	32.67	Class 1: 68.52%
Graph #1	0.5	0.7304	30.80	Class 0: 93.34%
Graph #1	1.0	1.4609	31.45	Class 0: 93.34% (Original)
Graph #1	1.5	2.1913	32.11	Class 0: 93.34%
Graph #1	2.0	2.9217	32.77	Class 0: 93.34%

intervention, this provides additional empirical evidence that the radial magnitude is associated with structure-related variation, while the angular direction carries most of the label-predictive information in this experiment.

K.6 Other Analysis

K.6.1 Evaluation of Class-Consistent Flow Coupling

A key challenge in DisRFM is to construct semantically reliable source–target trajectories while reducing arbitrary cross-domain mixing. Since target labels are unavailable during training, we evaluate different coupling strategies using ground-truth labels only as an oracle metric. As shown in Table 11, coupling precision improves monotonically across the evaluated strategies. Random coupling produces substantial semantic mismatch, while global nearest-neighbor coupling and standard pseudo-label coupling improve alignment but may still introduce mismatch noise and potentially contribute to negative transfer. In contrast, our proposed Confidence-Gated Class-Conditioned Coupling achieves the highest precision among the compared strategies. By filtering uncertain boundary samples, the method reduces likely semantic mismatches and provides cleaner source–target pairs for flow construction. Consequently, the learned vector field is supervised on trajectories that are more likely to connect semantically consistent source and target representations, supporting more semantic-aware transport.

K.6.2 Trajectory Diagnostics of Riemannian Flow Matching

A key empirical question is whether the learned Riemannian flow trajectories respect the intended radial–angular separation during transport. To examine this, we analyze integrated trajectory metrics along the ODE path. As shown in Table 12, the results suggest the following observations:

- **Radial Discrepancy Reduction.** The flow substantially reduces the radial Wasserstein-1 (W_1) discrepancy between source and target domains in the evaluated trajectories, suggesting improved alignment of structure-related radial statistics.
- **Small Angular Drift (AD_{ang}).** Despite radial alignment, the accumulated angular drift remains small. This indicates that the transport introduces limited perturbation to the angular direction, which is consistent with preserving most label-predictive information under the trained classifier.
- **No Observed Boundary Crossings.** Along the evaluated trajectories, we observe no class-boundary crossings under the trained decision function. This suggests that the learned

Table 10: Counterfactual angular intervention.

Alpha	Angular Source	Pred Nodes (Struct)	Prob Class 0	Prob Class 1
0.00	100% u_A (Class 0)	31.45	93.34%	6.66%
0.25	Mixed (0.25)	31.45	87.80%	12.20%
0.50	Mixed (0.50)	31.45	74.45%	25.55%
0.75	Mixed (0.75)	31.45	52.12%	47.88%
1.00	100% u_B (Class 1)	31.45	31.48%	68.52%

Table 11: Evaluation of source-target coupling precision.

Coupling Strategy	Precision
Random coupling	50.55%
Global nearest neighbor	77.31%
Pseudo-label class-conditioned NN	79.15%
Confidence-gated class-cond NN (Ours)	**81.33%**

transport does not visibly change the predicted semantic region for these samples, although it should be interpreted as an empirical diagnostic rather than a formal invariance guarantee.

- **Lower Path Energy ($E(\gamma)$).** The Riemannian trajectory has lower path energy than the Euclidean transport baseline in this diagnostic. This is consistent with the geometry-aware flow following more efficient manifold-adapted paths, but does not imply global optimality of the learned transport.

Overall, these trajectory-level metrics provide empirical evidence that the learned Riemannian flow mainly adjusts radial discrepancies while introducing limited angular perturbation in the reported setting. This supports the intended structure-aware and class-consistent behavior of DisRFM, without assuming exact radial–angular independence throughout the entire latent space.

L Existing Assets, Licenses, and Terms of Use

We use only existing public benchmark datasets and open-source software packages. Table 25 summarizes the external assets used in this work, including their original sources, access routes, licenses or terms of use, and our compliance actions. We do not redistribute raw datasets, modified datasets, pretrained models, or third-party code. All datasets are accessed through their official sources or standard loaders, including TUDataset/PyTorch Geometric and OGB. For datasets whose upstream providers do not specify a named standard license, we report the license as "not explicitly specified by the provider", follow the stated citation and usage requirements, cite both the dataset collection and the original dataset papers, and use the data only for academic benchmarking. Baseline methods are credited through citations to their original papers; unless otherwise stated, we use our own implementations and do not redistribute official baseline code. If code is released, we will include a third-party notice file preserving the copyright and license information of all software dependencies.

M Limitations

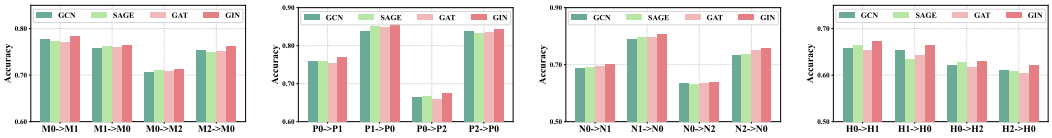
The scope and limitations of DisRFM are discussed as follows. The proposed framework relies on a geometry-aware inductive bias induced by polar coordinates, which may not fully capture the relationship between structural and semantic factors, particularly in noisy graph domains. In addition, the use of Riemannian operations and flow-based transport introduces additional computational overhead, which may become a bottleneck for high-dimensional embeddings. The theoretical analysis is explicitly conditional and does not claim universal topology–semantics identifiability or global convergence for nonconvex training. These observations suggest directions for future work, including improving scalability, reducing hyperparameter sensitivity, and extending the framework to more general settings, such as heterogeneous and generative graph tasks.

Table 12: Trajectory analysis of Riemannian Flow Matching.

Trajectory Metric	**Evaluation Result**
Radial Discrepancy (W_1 Dist)	$0.4176 \rightarrow \mathbf{0.0000}$
Angular Drift (AD_{ang})	0.1533 radians
Class Boundary Crossings	$\mathbf{0.00}\%$
Path Energy ($E(\gamma)$)	**0.0033** (Hyperbolic FM) vs. 0.0246 (Euclidean)

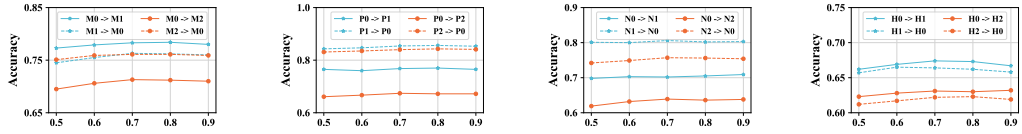
N Broader Impacts

This work studies a foundational graph domain adaptation problem and does not introduce new human-subject data, scraped personal data, or high-risk deployed systems. The proposed DisRFM framework may have positive impacts in scientific and industrial graph learning scenarios, such as molecular property prediction, protein or biochemical graph analysis, and other relational domains where labeled target data are scarce and distribution shifts are common. By preserving topology-related information and avoiding indiscriminate domain confusion, the method can improve robustness under structural and semantic shifts. However, graph adaptation methods may also create risks when applied to high-stakes or sensitive relational data, since biased graph construction, noisy pseudo-labels, or domain mismatch may lead to negative transfer, unfair predictions, or unreliable downstream decisions. Therefore, practical deployment should include target-domain validation, calibration checks, subgroup or structural-regime evaluation, privacy protection when relational data involve people, and human expert oversight in scientific or safety-critical applications.



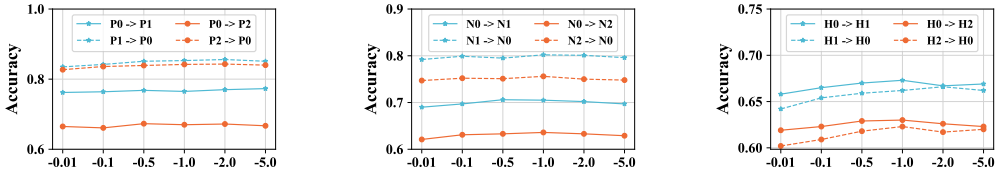
(a) Mutagenicity (b) PROTEINS (c) NCI1 (d) ogbg-molhiv

Figure 7: The performance with different tangent space aggregation modules on different datasets.



(a) Mutagenicity (b) PROTEINS (c) NCI1 (d) ogbg-molhiv

Figure 8: Hyperparameter sensitivity analysis of confidence threshold ζ on the Mutagenicity, PROTEINS, NCI1 and ogbg-molhiv datasets.



(a) PROTEINS (b) NCI1 (c) ogbg-molhiv

Figure 9: Hyperparameter sensitivity analysis of curvature parameter c on the PROTEINS, NCI1 and ogbg-molhiv datasets.

Table 13: The results of ablation studies on the Mutagenicity dataset (source → target). **Bold** results indicate the best performance.

Methods	M0→M1	M1→M0	M0→M2	M2→M0	M0→M3	M3→M0	M1→M2	M2→M1	M1→M3	M3→M1	M2→M3	M3→M2
DisRFM w/o FM	76.2	73.6	69.6	74.0	65.3	66.2	77.2	81.1	66.3	69.3	71.2	73.6
DisRFM w/o RA	76.9	74.4	70.7	73.9	66.1	67.1	77.8	82.0	66.8	70.6	71.9	74.7
DisRFM w/o AA	76.4	74.0	70.2	73.5	65.7	66.0	77.0	81.3	65.9	69.5	71.3	73.8
DisRFM w/o PE	75.9	72.9	69.0	72.5	64.6	64.8	76.2	80.0	64.1	67.9	70.6	72.5
DisRFM	**78.3**	**76.3**	**71.3**	**76.1**	**67.7**	**69.2**	**79.0**	**83.3**	**68.3**	**73.0**	**73.3**	**76.0**

Table 14: The results of ablation studies on the PROTEINS dataset (source → target). **Bold** results indicate the best performance.

Methods	P0→P1	P1→P0	P0→P2	P2→P0	P0→P3	P3→P0	P1→P2	P2→P1	P1→P3	P3→P1	P2→P3	P3→P2
DisRFM w/o FM	74.6	82.6	64.9	81.7	72.0	80.9	65.3	72.3	72.7	71.9	72.9	59.9
DisRFM w/o RA	75.3	82.9	66.0	82.1	73.4	81.7	67.2	73.3	73.0	73.4	72.1	60.9
DisRFM w/o AA	75.1	83.0	65.7	81.5	72.9	81.5	66.1	72.7	71.6	72.7	73.0	60.3
DisRFM w/o PE	74.2	81.9	64.7	80.8	71.9	80.6	65.3	71.9	70.8	72.0	71.9	59.6
DisRFM	**77.0**	**85.6**	**67.4**	**84.3**	**75.9**	**83.8**	**67.9**	**74.9**	**75.9**	**74.4**	**74.5**	**62.9**

Table 15: The results of ablation studies on the NCI1 dataset (source → target). **Bold** results indicate the best performance.

Methods	N0→N1	N1→N0	N0→N2	N2→N0	N0→N3	N3→N0	N1→N2	N2→N1	N1→N3	N3→N1	N2→N3	N3→N2
DisRFM w/o FM	68.4	78.4	62.3	73.4	62.2	70.0	68.6	70.1	66.2	56.4	72.1	64.0
DisRFM w/o RA	68.9	78.2	62.6	72.9	63.2	71.3	69.2	70.7	66.4	57.2	72.6	64.5
DisRFM w/o AA	67.8	78.7	62.3	74.0	62.3	71.0	68.7	69.8	65.1	56.7	72.7	63.9
DisRFM w/o PE	67.0	77.6	62.0	71.7	62.0	70.2	67.9	69.3	64.0	56.4	71.9	63.2
DisRFM	**70.2**	**80.6**	**63.9**	**75.7**	**64.6**	**72.6**	**70.7**	**71.9**	**68.0**	**58.2**	**74.4**	**65.6**

Table 16: The results of ablation studies on the ogbg-molhiv dataset (source → target). **Bold** results indicate the best performance.

Methods	H0→H1	H1→H0	H0→H2	H2→H0	H0→H3	H3→H0	H1→H2	H2→H1	H1→H3	H3→H1	H2→H3	H3→H2
DisRFM w/o FM	65.1	64.7	60.8	60.0	64.3	58.3	67.2	66.1	70.8	65.9	74.8	68.2
DisRFM w/o RA	66.1	65.6	61.7	59.4	64.0	59.7	67.7	67.0	70.1	65.7	75.7	68.8
DisRFM w/o AA	64.9	64.3	59.8	58.8	63.8	58.0	66.5	66.3	69.9	64.6	74.3	67.5
DisRFM w/o PE	64.5	63.8	59.4	58.6	62.6	58.2	66.0	66.1	68.4	63.9	74.0	67.1
DisRFM	**67.4**	**66.4**	**63.1**	**62.2**	**66.2**	**60.9**	**68.7**	**68.4**	**72.1**	**67.3**	**77.2**	**70.5**

Table 17: The graph classification results (in %) on PROTEINS under node density domain shift (source → target). P0, P1, P2, and P3 denote the sub-datasets partitioned with node density. **Bold** results indicate the best performance.

Methods	P0→P1	P1→P0	P0→P2	P2→P0	P0→P3	P3→P0	P1→P2	P2→P1	P1→P3	P3→P1	P2→P3	P3→P2
WL subtree	69.1	59.7	61.2	75.9	41.6	83.5	61.5	72.7	24.7	72.7	63.1	62.9
GCN	73.7±0.3	82.7±0.4	57.6±0.2	84.0±1.3	24.4±0.4	17.3±0.2	57.6±0.1	70.9±0.7	24.4±0.5	26.3±0.1	37.5±0.2	42.5±0.8
GIN	71.8±2.7	70.2±4.7	58.5±4.3	56.9±4.9	74.2±1.7	78.2±3.3	63.3±2.7	67.1±3.8	35.9±4.2	61.0±2.4	71.9±2.1	65.1±1.0
GMT	73.7±0.2	82.7±0.1	57.6±0.3	83.1±0.5	75.6±1.4	17.3±0.6	57.6±1.5	73.7±0.6	75.6±0.4	26.3±1.2	75.6±0.7	42.4±0.5
CIN	74.1±0.6	83.8±1.0	60.1±2.1	78.6±3.1	75.6±0.2	74.8±3.7	63.9±2.7	74.1±0.6	57.0±4.3	58.9±3.3	75.6±0.7	63.6±1.0
PathNN	71.5±1.6	67.1±1.4	63.5±1.1	54.2±1.9	44.6±1.3	45.6±1.9	63.1±2.3	63.5±1.6	28.0±2.2	49.1±1.9	53.2±2.6	58.9±1.9
dDGM	72.5±1.3	81.2±1.5	64.6±1.2	81.5±1.4	59.8±1.0	68.4±1.5	66.2±1.6	71.8±0.8	50.8±1.2	65.1±1.8	67.5±1.1	62.5±1.5
RieGrace	75.0±1.2	81.7±1.1	67.6±0.6	80.8±0.8	61.6±1.7	60.0±1.2	65.9±0.7	71.0±1.3	59.8±1.7	60.2±1.3	65.7±1.2	60.6±1.5
ProGDM	74.4±0.8	83.6±0.7	66.8±0.6	83.1±0.9	52.6±1.0	17.3±1.0	63.7±1.3	74.2±1.2	33.0±0.5	26.3±1.0	65.9±1.4	42.4±1.0
D-GCN	73.7±1.0	82.3±1.2	57.6±1.0	83.5±1.3	42.4±1.0	56.8±1.3	58.7±1.0	74.0±1.1	48.2±1.2	55.8±2.1	42.2±1.2	51.9±1.8
DEAL	75.2±1.2	78.8±3.3	67.5±2.1	74.5±6.1	65.2±2.2	76.2±2.0	66.6±0.9	73.7±1.4	71.1±1.8	69.9±2.7	57.6±1.5	65.7±0.4
SGDA	64.2±0.5	61.0±0.7	66.9±1.2	61.9±0.9	65.4±1.6	66.5±1.0	64.6±1.1	60.1±0.5	66.3±1.3	59.3±0.8	66.0±1.6	66.2±1.3
StruRW	71.9±2.3	82.6±1.9	66.7±1.8	74.5±2.8	52.8±1.9	57.3±2.0	62.2±2.4	63.3±2.1	59.5±1.6	56.3±2.0	66.6±2.3	52.4±2.0
A2GNN	65.7±0.6	65.9±0.8	66.3±0.9	65.6±1.1	65.2±1.4	65.6±1.3	65.9±1.7	65.8±1.6	65.0±1.5	66.1±1.2	65.2±1.9	65.9±1.8
PA-BOTH	61.0±0.8	61.2±1.3	60.3±0.6	66.7±2.1	63.7±1.5	61.9±2.0	66.2±1.4	69.9±2.3	68.0±0.7	69.4±1.8	61.5±0.4	**67.6±1.0**
GAA	74.9±1.3	83.5±1.4	65.9±1.3	82.3±1.9	64.4±1.7	55.2±1.8	67.0±1.7	74.3±1.3	65.4±1.9	58.7±1.7	70.5±2.1	57.5±1.5
TDSS	73.9±1.2	82.9±1.3	57.8±1.3	83.1±1.2	60.4±1.7	61.6±2.0	58.1±1.4	74.4±1.5	62.7±1.9	65.8±1.0	58.2±1.7	65.5±1.0
GOTDA	74.5±1.4	83.3±1.3	64.5±1.4	83.7±0.7	52.3±1.2	70.4±0.7	65.0±1.3	74.6±1.5	38.6±1.6	68.6±1.0	64.3±1.4	61.9±1.3
MASH	74.7±1.2	83.6±1.3	62.6±1.7	84.1±0.8	47.1±1.3	71.1±0.9	60.9±1.6	74.5±1.2	49.3±0.9	65.9±1.3	64.7±1.4	66.0±2.0
GeoAdapt	74.1±1.2	83.0±1.4	66.4±1.0	83.3±1.2	68.5±1.8	73.9±2.2	65.9±1.4	74.0±1.4	67.8±2.0	64.5±1.9	68.7±1.6	59.9±2.3
DisRFM-$\mathbb{E}^n$	75.8±0.7	83.3±1.2	68.0±1.1	83.2±1.5	74.8±2.0	82.1±1.7	65.8±0.5	73.9±1.2	72.5±1.4	72.0±1.5	72.8±1.9	62.2±2.1
DisRFM-$\mathbb{S}^n$	76.1±1.4	83.7±1.5	68.3±0.6	83.8±1.6	75.6±1.3	82.7±1.8	67.0±1.1	75.1±0.6	74.1±1.7	73.7±1.4	73.3±1.4	64.6±1.1
DisRFM-$\mathbb{H}^n$	**76.4±1.7**	**84.1±2.1**	**69.0±1.4**	**84.3±1.5**	**75.9±1.1**	**83.9±1.9**	**67.2±1.8**	**75.5±1.3**	**75.7±2.0**	**74.4±1.6**	**75.7±1.7**	66.1±1.9

Table 18: The graph classification results (in %) on PROTEINS under edge density domain shift (source → target). P0, P1, P2, and P3 denote the sub-datasets partitioned with edge density. **Bold** results indicate the best performance.

Methods	P0→P1	P1→P0	P0→P2	P2→P0	P0→P3	P3→P0	P1→P2	P2→P1	P1→P3	P3→P1	P2→P3	P3→P2
WL subtree	68.7	82.3	50.7	82.3	58.1	83.8	64.0	74.1	43.7	70.5	71.3	60.1
GCN	73.4±0.2	83.5±0.3	57.6±0.2	84.2±1.8	24.0±0.1	16.6±0.4	57.6±0.2	73.7±0.4	24.0±0.1	26.6±0.2	39.9±0.9	42.5±0.1
GIN	62.5±4.7	74.9±3.7	53.0±4.6	59.6±4.2	73.7±0.8	64.7±3.4	60.6±2.7	69.8±0.6	31.1±2.8	63.1±3.4	72.3±2.7	64.6±1.4
GMT	73.4±0.3	83.5±0.2	57.6±0.1	83.5±0.3	24.0±0.1	83.5±0.1	57.4±0.2	73.4±0.2	24.1±0.1	73.4±0.3	24.0±0.1	57.6±0.2
CIN	74.5±0.2	84.1±0.5	57.8±0.2	82.7±0.9	75.6±0.6	79.2±2.2	61.5±2.7	74.0±1.0	75.5±0.8	72.5±2.1	**76.0±0.3**	60.9±1.2
PathNN	72.7±1.3	80.8±1.6	63.5±2.5	68.1±1.1	43.2±1.6	25.3±1.5	60.2±1.7	66.0±1.3	33.1±2.9	38.5±1.1	49.8±1.4	54.3±1.1
dDGM	73.0±1.4	82.1±1.4	65.9±1.4	81.4±1.3	58.6±1.7	56.5±1.6	65.0±1.3	72.3±1.2	50.0±1.0	56.0±1.1	68.0±1.0	59.6±1.4
RieGrace	74.9±1.4	82.6±1.3	65.4±1.3	82.1±1.0	61.9±1.8	66.7±1.9	64.1±0.8	72.7±1.5	59.6±1.2	63.3±1.6	70.7±1.1	61.0±1.0
ProGDM	74.5±1.0	84.7±1.2	65.3±0.8	83.3±1.1	48.0±1.2	16.5±1.0	62.7±1.4	74.6±1.2	34.8±0.9	26.6±1.0	62.7±1.6	42.4±1.0
D-GCN	73.4±1.0	83.1±1.3	57.6±1.5	82.2±1.3	44.0±0.9	56.9±1.7	59.3±1.1	74.2±1.3	51.8±1.2	55.4±1.4	42.8±1.0	51.8±0.7
DEAL	74.1±0.3	78.8±2.3	65.1±1.8	79.1±3.4	63.7±2.7	67.3±5.3	69.0±0.9	74.2±1.6	60.8±2.4	70.6±2.8	71.3±5.7	65.9±0.9
SGDA	63.8±0.6	65.2±1.3	66.7±1.0	59.1±1.5	60.1±0.8	64.4±1.2	65.2±0.7	63.9±0.9	64.5±0.6	61.1±1.3	58.9±1.4	64.9±1.2
StruRW	72.6±2.2	84.5±1.7	66.2±2.2	72.5±2.4	48.9±2.0	56.5±2.3	63.1±1.8	64.4±2.4	55.8±2.0	56.6±2.4	67.0±2.6	42.4±2.0
A2GNN	65.4±1.3	66.3±1.1	65.2±1.4	66.3±1.2	65.4±0.7	65.9±0.9	66.9±1.3	65.4±1.2	65.6±0.9	65.5±1.2	66.1±2.0	**66.0±1.8**
PA-BOTH	63.1±0.7	67.2±1.1	64.3±0.5	72.1±1.8	66.3±0.7	64.1±1.2	**69.7±2.1**	67.5±1.8	61.2±1.4	67.7±2.3	61.2±1.6	65.5±0.6
GAA	75.4±1.3	82.8±1.4	66.9±1.8	83.5±1.5	64.2±1.9	56.0±2.2	67.5±1.0	74.5±1.3	60.1±1.6	60.3±1.5	72.4±1.5	58.1±1.4
TDSS	73.6±1.2	83.7±1.2	57.8±1.3	83.7±1.5	60.9±1.3	61.5±2.2	67.8±1.7	73.5±1.3	64.1±1.0	63.3±1.6	66.3±1.1	59.9±2.0
GOTDA	75.0±1.2	83.6±1.3	65.7±1.3	83.8±1.5	51.5±1.6	70.4±1.1	64.7±1.0	74.6±0.9	36.7±1.4	64.2±1.8	61.4±1.2	59.7±1.0
MASH	74.4±1.5	84.6±1.3	61.4±1.3	82.7±1.6	47.7±1.5	63.5±0.9	61.9±1.7	73.8±1.1	47.8±0.9	66.6±1.4	59.0±1.2	64.7±1.0
GeoAdapt	74.5±1.3	83.8±1.4	65.3±1.3	84.0±1.6	66.9±1.5	73.3±1.8	65.0±1.5	73.9±2.2	71.9±1.7	64.9±2.4	61.1±2.0	63.1±1.9
DisRFM-$\mathbb{E}^n$	76.0±0.7	82.9±1.3	66.7±0.3	82.0±1.3	71.1±1.6	81.2±1.8	67.0±1.0	73.2±0.8	72.9±1.5	71.1±2.1	71.6±1.1	61.7±1.7
DisRFM-$\mathbb{S}^n$	74.9±0.7	84.4±1.3	66.8±0.3	83.4±1.1	72.6±1.7	81.3±1.6	67.5±0.9	74.0±0.5	73.7±1.9	72.6±1.7	72.3±1.7	61.3±1.8
DisRFM-$\mathbb{H}^n$	**77.0±1.3**	**85.6±1.5**	**67.4±1.4**	**84.3±2.1**	**75.9±2.0**	**83.8±1.7**	67.9±1.6	**74.9±1.3**	**75.9±1.5**	**74.4±1.9**	74.5±1.4	62.9±1.7

Table 19: The classification results (in %) on the NCI1 dataset under node density domain shift (source → target). N0, N1, N2, and N3 denote the sub-datasets partitioned with node density. **Bold** results indicate the best performance.

Methods	N0→N1	N1→N0	N0→N2	N2→N0	N0→N3	N3→N0	N1→N2	N2→N1	N1→N3	N3→N1	N2→N3	N3→N2
WL subtree	54.9	60.0	51.4	51.4	44.4	63.1	51.9	53.4	58.4	57.4	61.2	50.9
GCN	54.0±2.0	66.5±1.9	47.3±2.2	37.4±1.8	38.4±2.3	27.6±1.5	59.0±1.7	56.9±2.0	56.5±1.2	51.0±1.7	60.5±1.9	57.4±1.6
GIN	57.0±1.9	38.7±2.3	45.7±1.8	34.4±1.9	43.6±2.1	27.0±2.0	54.7±1.7	52.8±2.1	59.0±1.8	49.1±2.5	59.4±2.0	57.4±1.3
GMT	53.0±1.9	63.7±1.6	48.9±2.2	32.8±2.0	41.5±1.7	27.8±1.6	56.4±1.9	57.2±1.7	54.1±1.9	49.8±1.7	61.6±2.1	58.5±1.1
CIN	54.5±1.9	37.6±1.9	53.2±1.8	36.2±1.4	42.1±1.8	27.5±2.0	57.2±1.6	51.6±1.9	59.3±1.5	50.0±1.4	62.0±1.4	57.4±2.3
PathNN	56.8±2.5	62.8±2.1	52.7±2.2	64.5±1.2	50.6±1.8	37.4±2.1	59.2±1.4	58.6±1.5	60.2±1.8	53.1±2.1	60.9±1.2	57.6±1.5
dDGM	69.1±0.5	76.7±0.6	60.9±0.7	75.5±1.1	60.5±1.9	72.8±0.4	70.5±0.5	68.0±0.6	66.4±0.9	56.4±1.7	68.6±1.3	65.3±1.1
RieGrace	68.4±0.5	75.5±0.9	61.0±0.8	73.8±0.8	58.8±1.4	63.4±1.3	69.9±1.3	66.7±1.2	63.9±0.5	57.4±2.1	69.2±0.4	65.9±0.7
ProGDM	53.6±1.2	73.6±1.2	46.4±0.9	27.0±1.0	44.5±2.2	27.0±1.1	57.5±1.5	49.1±1.0	66.5±1.3	49.2±1.6	66.5±1.0	57.4±1.7
D-GCN	56.7±0.7	77.3±0.6	51.2±0.9	69.8±1.9	40.4±0.9	27.4±0.1	63.0±1.1	65.2±0.3	62.2±1.6	49.9±0.5	66.5±1.0	58.7±0.8
DEAL	70.6±0.6	77.4±0.8	62.9±1.2	74.5±0.5	62.6±4.4	70.3±1.6	69.4±0.6	70.0±0.7	66.3±0.6	**62.4±1.4**	74.0±0.6	66.6±1.0
SGDA	69.2±1.2	76.5±1.1	63.2±0.9	73.4±0.8	53.2±1.1	60.2±1.5	70.5±1.3	69.0±0.5	63.0±0.9	57.4±1.5	68.3±1.3	66.5±0.8
A2GNN	53.0±1.3	73.7±0.5	52.9±1.9	73.6±1.0	58.9±1.7	70.5±1.8	59.8±1.8	59.4±0.7	63.6±1.8	59.3±0.8	66.7±0.4	61.2±1.7
StruRW	67.3±0.7	77.0±2.0	62.1±1.0	73.9±0.6	53.6±2.5	62.6±1.7	69.7±0.5	68.3±0.3	62.7±2.1	62.1±2.0	68.3±1.2	67.1±0.5
PA-BOTH	51.2±0.5	75.4±1.0	42.8±0.1	64.2±1.9	35.8±1.1	55.3±2.1	61.6±2.1	53.4±1.7	64.4±1.0	50.9±1.1	66.7±0.4	57.5±0.2
GAA	70.6±0.9	78.8±0.6	63.7±0.5	74.5±0.9	62.9±0.7	72.7±0.7	70.6±0.3	69.0±0.8	**66.7±0.8**	62.0±1.8	69.4±1.1	66.9±0.5
TDSS	50.9±0.9	74.3±0.6	42.8±1.2	60.4±1.0	33.6±0.1	36.4±1.3	59.4±1.5	52.6±2.3	66.3±0.4	49.5±0.8	66.6±1.1	57.3±1.0
GOTDA	67.6±0.7	75.2±1.3	63.4±1.2	73.5±0.5	52.5±0.8	41.2±1.1	70.4±0.2	67.4±0.4	64.3±0.9	54.2±0.5	67.8±0.6	63.4±0.7
MASH	52.1±0.7	75.4±0.8	43.1±1.0	68.1±1.3	37.1±0.8	36.7±1.2	65.4±1.3	60.3±1.6	63.4±2.3	49.8±0.8	66.6±0.8	57.7±1.2
GeoAdapt	64.6±1.0	76.1±1.0	59.9±1.9	71.2±4.5	49.2±0.6	47.7±1.3	69.3±0.4	66.1±0.4	64.3±1.4	53.5±1.1	67.3±0.7	61.1±0.6
DisRFM-$\mathbb{E}^n$	72.6±0.7	77.6±0.8	63.1±0.5	75.1±1.3	62.9±1.9	70.4±1.9	71.1±0.5	73.9±0.6	63.9±1.7	57.8±1.3	72.2±0.7	66.4±0.7
DisRFM-$\mathbb{S}^n$	73.8±0.4	78.1±0.7	63.3±0.7	75.0±0.5	63.7±1.5	72.3±1.1	70.7±0.3	74.7±0.4	64.7±1.5	58.5±0.8	**75.2±0.5**	**67.8±0.7**
DisRFM-$\mathbb{H}^n$	**74.9±1.9**	**79.2±1.1**	**64.0±1.2**	**75.7±1.4**	**63.8±1.5**	**73.5±1.7**	**71.6±1.6**	**75.6±1.5**	65.9±1.7	59.7±2.0	74.3±1.7	67.7±1.4

Table 20: The classification results (in %) on the NCI1 dataset under edge density domain shift (source → target). N0, N1, N2, and N3 denote the sub-datasets partitioned with edge density. **Bold** results indicate the best performance.

Methods	N0→N1	N1→N0	N0→N2	N2→N0	N0→N3	N3→N0	N1→N2	N2→N1	N1→N3	N3→N1	N2→N3	N3→N2
WL subtree	50.1	59.9	50.9	51.0	51.8	54.5	56.7	57.0	52.5	57.6	54.4	46.9
GCN	53.7±2.1	68.2±1.9	50.6±1.9	45.8±1.7	35.5±2.4	28.2±2.2	55.9±1.7	56.8±1.9	55.7±1.5	53.2±2.1	60.4±1.8	54.8±0.9
GIN	51.4±1.7	47.5±2.1	50.4±1.6	32.7±1.6	47.0±2.1	29.4±2.3	52.4±1.9	55.7±2.0	64.3±1.8	51.1±1.2	60.3±1.3	53.5±1.6
GMT	53.0±1.9	58.3±1.7	49.0±1.9	34.3±1.6	39.7±1.5	27.5±2.4	55.8±1.6	60.4±2.1	51.3±1.0	51.5±1.6	60.0±2.6	54.5±1.4
CIN	54.5±2.1	36.8±1.9	53.0±1.3	32.7±2.1	44.9±1.4	27.4±1.1	58.4±1.6	54.2±2.3	63.9±1.6	51.2±1.4	64.7±1.1	53.7±2.1
PathNN	56.4±1.4	65.5±1.4	54.6±1.5	60.9±1.8	46.7±2.1	35.5±1.9	57.6±1.8	60.6±1.2	61.9±1.8	53.9±2.0	65.1±1.8	52.5±1.5
dDGM	69.1±0.4	77.3±1.1	62.7±0.8	74.7±0.9	60.9±1.2	63.4±1.4	68.5±0.6	68.4±0.3	68.3±1.1	56.7±2.6	68.9±0.2	60.8±0.6
RieGrace	68.5±1.2	76.7±0.5	62.1±0.7	74.4±1.6	56.9±1.4	55.2±1.0	68.2±0.5	67.7±0.5	67.8±1.1	57.2±1.5	69.3±0.1	63.5±1.5
ProGDM	54.0±1.2	73.0±1.1	47.4±1.2	57.6±1.6	41.8±0.9	27.5±1.1	53.9±1.3	52.1±1.4	68.2±1.0	51.7±1.4	68.2±1.0	53.4±1.2
D-GCN	56.2±1.3	78.2±0.5	54.8±0.5	72.3±1.7	41.2±0.7	27.5±1.1	61.1±0.4	65.0±0.1	64.8±1.1	51.4±0.3	68.2±1.0	53.8±0.7
DEAL	70.3±0.4	80.2±0.6	62.9±1.7	74.7±3.9	61.9±2.6	70.7±3.7	67.5±0.7	69.7±1.0	67.6±0.4	**63.4±1.3**	73.2±0.3	64.2±0.9
SGDA	68.3±1.4	78.5±0.5	62.4±0.8	74.4±0.6	51.6±1.3	46.8±1.2	69.3±0.6	68.9±0.5	64.0±1.0	56.1±1.0	68.2±0.7	62.1±1.0
A2GNN	53.5±1.3	74.8±0.9	53.1±0.8	73.7±1.4	52.1±1.4	71.0±1.6	59.6±2.4	59.5±1.4	67.8±0.6	58.9±1.4	68.4±0.7	59.6±1.4
StruRW	66.4±1.0	76.7±1.9	63.2±0.5	75.2±1.2	53.1±1.3	53.5±1.4	68.5±0.7	67.9±1.3	65.5±2.1	59.7±1.6	66.7±0.9	62.6±1.4
PA-BOTH	49.4±0.8	75.1±1.5	48.1±2.5	65.5±1.6	40.3±2.0	55.4±1.2	60.1±1.1	55.8±2.1	68.2±2.0	51.5±0.5	68.4±0.3	54.0±1.4
GAA	70.0±1.1	80.4±1.0	63.5±0.6	75.4±1.3	63.8±0.6	71.4±0.5	69.7±0.4	68.3±0.5	**69.6±0.3**	61.4±1.2	69.6±0.7	63.2±0.5
TDSS	49.0±1.0	74.7±0.7	46.8±1.0	72.8±0.9	31.9±0.8	36.5±1.1	56.3±2.7	54.1±1.9	68.2±2.0	51.0±1.4	68.3±1.3	53.2±1.0
GOTDA	66.4±0.6	74.8±0.7	63.7±1.0	73.9±0.7	51.3±1.1	39.4±1.8	68.2±0.4	68.7±0.4	65.5±0.6	54.5±1.1	67.7±0.6	58.6±0.7
MASH	51.5±1.6	76.4±1.4	48.0±0.9	72.1±2.7	36.1±1.2	37.5±1.7	63.3±0.6	60.3±2.0	66.0±1.7	51.9±1.8	67.3±1.2	54.3±2.1
GeoAdapt	62.7±0.8	75.6±0.9	60.4±0.7	72.9±1.2	46.6±1.3	37.9±1.4	67.3±1.3	67.6±0.4	64.9±2.0	54.3±1.3	66.0±1.1	57.6±1.3
DisRFM-$\mathbb{E}^n$	69.5±0.6	80.3±0.6	63.2±1.2	73.4±1.5	63.2±1.5	70.6±1.2	69.7±0.6	70.5±0.4	64.9±1.6	57.3±0.5	73.4±0.4	64.4±0.4
DisRFM-$\mathbb{S}^n$	**70.8±1.3**	80.4±0.3	63.3±1.1	74.8±1.5	63.9±1.6	71.1±1.0	70.4±0.4	71.0±0.3	65.7±1.8	58.1±0.6	73.6±0.7	64.8±0.7
DisRFM-$\mathbb{H}^n$	70.2±1.3	**80.6±1.5**	**63.9±1.0**	**75.7±1.1**	**64.6±1.2**	**72.6±1.7**	**70.7±1.4**	**71.9±0.7**	68.0±2.4	58.2±0.7	**74.4±0.4**	**65.6±1.2**

Table 21: The classification results (in %) on the Mutagenicity dataset under node density domain shift (source → target). M0, M1, M2, and M3 denote the sub-datasets partitioned with node density. **Bold** results indicate the best performance.

Methods	M0→M1	M1→M0	M0→M2	M2→M0	M0→M3	M3→M0	M1→M2	M2→M1	M1→M3	M3→M1	M2→M3	M3→M2
WL subtree	34.3	46.0	40.4	50.4	52.7	52.1	51.6	48.3	45.6	40.8	53.5	52.4
GCN	64.1±1.4	62.6±2.3	65.5±2.0	62.3±1.1	56.9±2.1	53.7±1.8	66.1±1.6	70.8±2.1	47.7±1.9	49.2±2.1	52.1±1.7	54.4±2.1
GIN	66.5±2.1	53.2±2.3	52.0±1.7	62.6±2.3	53.7±1.7	48.6±2.3	71.5±2.5	70.0±2.1	52.2±2.3	64.3±2.1	50.9±1.4	52.4±1.5
GMT	65.7±1.8	67.1±1.9	62.1±2.1	66.4±1.2	59.0±2.0	58.0±2.7	69.9±1.4	68.4±2.3	60.4±1.3	48.2±2.0	59.1±1.3	60.6±1.7
CIN	65.1±1.7	52.9±2.3	66.0±1.7	69.8±2.0	55.2±1.5	55.1±1.3	66.0±1.6	69.9±2.1	54.9±1.8	57.8±1.5	46.7±2.0	51.8±1.1
PathNN	70.2±1.5	68.9±1.5	67.1±2.0	70.2±1.3	58.0±1.9	60.4±2.2	68.9±1.6	70.3±1.9	59.8±1.7	57.7±1.5	59.9±1.7	61.8±0.9
dDGM	79.1±0.4	71.1±1.2	70.3±0.5	73.3±0.3	63.9±0.3	67.1±1.3	78.3±1.2	82.7±1.2	66.2±0.4	72.8±1.0	72.1±1.0	76.8±1.3
RieGrace	78.5±1.3	70.1±1.2	70.6±1.4	72.4±0.6	62.8±1.3	68.3±1.5	77.2±1.3	82.0±0.3	65.0±1.2	73.3±1.9	71.1±0.4	76.2±0.6
ProGDM	75.2±1.2	71.5±1.6	69.0±1.1	70.3±1.1	55.7±1.4	56.8±1.7	70.4±1.1	76.0±1.0	56.6±1.3	48.9±2.1	57.8±1.1	59.9±1.3
D-GCN	75.2±0.5	69.2±1.3	67.9±0.4	69.6±1.1	56.3±1.1	58.4±1.6	70.6±0.3	76.8±0.5	56.1±1.2	55.1±1.1	58.1±0.5	62.5±1.4
DEAL	77.1±0.9	70.9±0.4	70.9±0.9	69.2±1.5	60.3±1.1	57.1±9.3	77.6±0.5	82.5±1.3	62.9±1.6	74.1±1.0	65.8±1.2	76.4±1.1
SGDA	77.5±0.6	71.4±0.6	69.7±0.5	72.2±1.4	65.5±0.8	60.0±0.7	78.8±1.4	82.8±1.3	67.2±0.6	65.8±1.6	70.9±1.4	73.2±0.9
A2GNN	73.5±1.9	70.4±1.0	66.1±1.5	69.8±0.9	60.4±1.1	66.4±1.8	69.0±1.7	72.3±1.5	56.3±1.6	71.1±2.2	57.9±1.4	66.0±1.7
StruRW	78.3±1.3	71.0±1.4	69.7±1.3	71.3±1.5	62.6±0.7	62.6±1.6	78.8±0.4	82.2±1.2	65.8±0.8	71.5±1.3	70.0±1.4	74.5±1.0
PA-BOTH	69.8±1.5	47.5±2.0	63.8±1.9	48.8±1.6	55.3±1.1	53.9±1.7	59.9±1.8	67.3±2.0	49.1±1.3	68.6±1.4	48.6±1.4	62.7±1.7
GAA	79.3±1.2	73.9±1.3	71.2±0.7	72.6±1.3	65.6±1.3	63.7±2.3	77.1±1.4	82.1±1.4	67.7±1.3	75.8±1.0	70.7±1.6	75.2±0.9
TDSS	63.6±1.3	50.3±1.1	56.7±1.6	51.1±1.4	54.7±1.0	58.5±1.7	60.1±1.3	70.1±2.3	48.6±1.5	68.5±1.6	49.5±1.3	59.3±2.0
GOTDA	77.3±1.3	71.7±1.5	69.9±1.7	71.3±1.7	65.3±1.2	67.0±2.3	77.8±1.3	81.6±1.2	65.7±1.6	64.7±1.4	71.0±1.1	71.1±1.6
MASH	76.1±1.3	67.0±1.6	66.9±1.3	69.9±1.8	54.3±1.2	55.1±2.6	75.1±1.6	80.0±1.5	55.1±1.7	63.7±1.5	58.4±1.2	64.9±1.6
GeoAdapt	76.9±1.5	71.4±1.2	68.7±1.4	71.4±1.5	60.1±1.3	65.0±2.4	77.2±1.3	81.1±1.5	63.4±1.4	64.2±2.1	69.3±1.3	70.6±1.6
DisRFM-$\mathbb{E}^n$	79.4±0.7	72.1±0.7	71.0±1.0	72.9±0.6	65.2±0.2	70.4±3.0	79.3±0.1	82.9±0.5	66.4±0.7	73.8±1.8	71.9±0.5	77.6±0.4
DisRFM-$\mathbb{S}^n$	79.7±0.7	**74.6±0.6**	**72.0±1.0**	73.2±0.8	65.9±0.9	70.7±1.7	**82.7±0.1**	83.8±0.5	67.2±0.4	74.5±1.0	72.6±0.3	**80.1±0.4**
DisRFM-$\mathbb{H}^n$	**79.8±0.8**	72.3±0.7	71.7±0.7	**74.0±0.4**	**66.7±0.8**	**70.9±1.1**	81.1±0.3	**84.4±0.6**	**68.5±1.0**	**74.9±1.2**	**73.3±0.8**	79.1±1.0

Table 22: The classification results (in %) on the Mutagenicity dataset under edge density domain shift (source → target). M0, M1, M2, and M3 denote the sub-datasets partitioned with edge density. **Bold** results indicate the best performance.

Methods	M0→M1	M1→M0	M0→M2	M2→M0	M0→M3	M3→M0	M1→M2	M2→M1	M1→M3	M3→M1	M2→M3	M3→M2
WL subtree	34.4	39.6	47.6	43.6	52.7	46.4	59.8	40.9	53.0	65.1	52.6	42.4
GCN	66.3±1.7	65.1±2.2	63.6±1.4	67.6±1.6	56.0±1.4	49.6±1.9	66.1±2.0	71.8±1.8	48.0±2.3	47.8±1.7	51.6±1.8	56.6±2.2
GIN	67.1±1.7	56.1±1.5	54.2±2.6	65.6±1.6	55.4±1.9	50.8±2.1	67.9±2.1	69.4±2.0	53.4±1.6	66.7±2.0	50.4±1.1	55.4±1.2
GMT	67.9±1.3	63.9±1.8	61.5±1.8	68.5±1.5	58.2±2.4	54.3±1.8	66.7±1.9	76.1±1.7	58.8±1.5	44.5±1.5	59.0±1.4	63.7±1.7
CIN	66.3±1.8	56.7±2.5	60.8±1.7	67.4±1.7	55.8±2.4	52.3±1.8	63.4±2.3	69.3±1.2	54.3±1.2	60.7±1.4	49.0±1.3	54.4±1.7
PathNN	68.9±1.9	65.9±1.8	62.9±1.7	69.2±1.8	58.1±1.6	59.9±1.4	67.7±1.6	70.0±2.1	56.3±1.8	62.8±1.9	59.2±1.6	63.7±1.2
dDGM	76.5±1.3	75.5±0.6	69.0±1.3	75.7±1.3	67.0±0.4	63.2±0.9	77.7±1.2	81.1±0.3	65.6±0.3	67.3±1.1	71.0±0.5	75.0±0.9
RieGrace	76.4±0.5	76.0±0.4	68.9±0.6	74.9±1.2	66.1±1.2	67.3±1.5	77.3±1.1	80.7±1.2	67.5±0.7	70.4±1.2	71.7±0.4	75.4±0.8
ProGDM	74.6±1.2	72.8±1.4	67.5±1.2	71.9±1.2	55.5±0.7	52.6±1.3	69.4±1.1	74.5±1.2	56.7±0.9	48.7±1.6	58.2±1.1	62.8±1.1
D-GCN	73.8±1.2	70.8±1.2	66.8±1.4	71.4±1.2	56.7±1.0	56.9±1.2	69.6±1.5	75.1±1.3	56.6±1.3	55.4±0.7	59.4±1.2	63.9±0.7
DEAL	76.6±1.6	72.7±0.6	70.6±1.2	70.9±0.9	60.2±2.1	55.6±1.5	77.8±1.0	80.7±0.5	63.4±0.9	67.6±1.1	66.3±0.7	74.8±1.1
SGDA	75.9±1.6	74.3±1.3	68.9±0.8	74.3±0.4	64.4±0.4	56.2±0.7	78.5±0.3	81.8±1.1	68.1±0.3	65.2±2.2	71.7±0.4	73.4±1.2
A2GNN	69.5±1.4	72.0±1.8	68.6±1.4	71.0±2.0	58.8±2.2	69.5±1.1	69.0±1.2	71.5±1.3	57.1±2.7	69.3±1.0	59.7±1.3	67.4±1.2
StruRW	76.1±1.5	74.1±1.6	69.0±1.3	73.9±1.5	62.1±1.0	58.9±2.0	78.0±1.3	80.8±1.2	66.4±0.7	67.8±0.8	71.9±0.3	74.9±1.0
PA-BOTH	74.7±1.1	51.4±1.0	65.3±1.3	51.4±1.0	52.2±1.5	56.8±1.1	57.1±1.1	66.7±1.0	48.9±2.4	66.9±1.3	48.1±2.5	60.7±1.7
GAA	77.5±1.2	75.7±1.2	70.0±1.2	74.5±0.7	66.5±1.3	66.7±1.1	76.7±1.3	80.9±0.7	67.9±1.4	72.7±0.8	71.7±1.0	75.2±1.4
TDSS	71.6±1.5	53.6±1.2	67.3±1.0	56.2±1.3	55.4±1.6	62.2±1.0	57.3±1.3	71.1±1.5	48.2±1.2	54.4±2.2	50.9±1.6	61.5±1.9
GOTDA	76.1±1.2	74.0±0.3	68.5±0.8	74.3±0.3	65.8±0.6	62.4±1.7	77.3±0.5	80.1±0.9	66.7±0.4	63.9±1.7	71.4±0.6	72.5±0.3
MASH	74.4±1.2	70.0±0.7	66.1±0.5	71.8±0.7	52.2±1.2	53.0±1.1	74.3±0.6	78.8±1.2	55.1±1.2	60.5±2.1	59.8±1.5	67.0±0.5
GeoAdapt	75.5±1.2	74.3±0.9	67.5±1.2	74.4±0.8	60.5±0.6	61.1±1.2	77.0±1.0	79.9±0.7	64.4±0.6	64.3±1.3	70.0±0.7	72.2±0.5
DisRFM-$\mathbb{E}^n$	77.6±0.4	75.5±0.5	70.2±1.2	75.5±1.0	66.1±0.3	66.8±4.7	77.9±0.4	82.3±0.1	67.0±0.8	69.9±3.2	71.8±0.2	74.4±0.4
DisRFM-$\mathbb{S}^n$	78.0±0.5	**76.6±0.9**	71.0±0.6	75.8±0.6	**68.2±1.0**	**71.5±1.1**	78.7±0.3	83.1±0.4	**71.8±0.4**	72.8±3.7	73.1±0.5	75.6±0.8
DisRFM-$\mathbb{H}^n$	**78.3±0.7**	76.3±1.4	**71.3±0.4**	**76.1±1.3**	67.7±0.9	69.2±1.5	**79.0±1.4**	**83.3±1.5**	68.3±1.9	**73.0±1.3**	**73.3±1.4**	**76.0±1.6**

Table 23: The classification results (in %) on the ogbg-molhiv dataset under node density domain shift (source → target). H0, H1, H2, and H3 denote the sub-datasets partitioned with node density. **Bold** results indicate the best performance.

Methods	H0→H1	H1→H0	H0→H2	H2→H0	H0→H3	H3→H0	H1→H2	H2→H1	H1→H3	H3→H1	H2→H3	H3→H2
WL	49.6	51.8	54.0	54.7	48.2	48.9	50.6	51.9	38.1	51.0	51.8	50.7
GCN	57.8±1.9	57.4±2.1	56.5±1.3	54.2±1.2	54.5±1.7	51.9±1.7	61.8±2.0	55.2±1.3	49.8±1.4	56.6±1.3	61.5±1.1	56.8±1.7
GIN	59.1±1.6	53.3±1.6	50.0±1.7	54.8±1.8	56.8±1.6	50.7±1.4	55.2±1.3	58.7±1.4	57.0±1.8	49.6±1.7	53.1±1.3	55.3±1.6
GMT	51.1±1.6	57.2±1.7	50.6±1.2	55.9±1.9	58.4±1.4	53.4±1.7	56.1±1.6	57.6±1.3	60.9±1.7	54.5±2.0	55.4±1.6	51.2±1.8
CIN	52.3±1.6	53.4±1.5	54.0±1.9	57.5±1.8	56.5±1.4	51.1±1.9	55.8±1.4	63.6±1.8	48.7±1.6	56.8±1.6	49.3±1.2	52.8±1.3
PathNN	50.3±1.3	51.2±1.8	51.5±1.2	55.3±1.8	56.7±1.2	52.2±1.5	53.7±1.7	51.5±1.4	53.9±1.9	54.3±1.5	55.6±1.8	56.0±1.5
dDGM	65.9±0.4	66.5±0.6	62.4±0.6	62.3±0.4	62.3±1.8	59.4±1.3	69.3±0.5	68.5±1.0	71.9±0.7	64.7±0.6	76.3±0.4	67.4±0.4
RieGrace	66.1±0.6	65.6±0.4	62.9±1.0	61.1±1.2	62.2±1.4	58.5±1.0	69.3±1.3	**69.4±1.4**	71.8±0.8	66.7±0.9	71.9±0.9	67.7±0.5
ProGDM	59.4±1.2	62.0±1.2	59.7±1.2	62.1±0.8	58.4±0.9	59.6±0.9	61.1±1.3	61.8±1.2	60.1±1.2	60.1±1.1	60.1±0.8	59.1±0.9
D-GCN	60.4±1.6	59.6±1.7	60.0±0.8	57.5±1.1	60.6±1.9	56.0±1.2	60.1±0.9	61.0±0.9	65.0±1.0	58.6±1.6	65.2±2.0	62.1±0.9
DEAL	62.7±1.4	64.9±0.5	61.6±1.5	61.2±1.3	64.4±1.1	60.5±1.9	67.0±0.9	65.7±0.8	68.8±3.9	63.4±1.2	67.1±2.4	66.8±1.4
SGDA	67.2±1.5	67.9±1.3	62.4±1.5	61.7±1.4	59.4±1.9	57.7±0.8	69.5±1.3	67.9±0.7	71.0±0.9	64.8±0.8	75.1±1.4	67.6±0.5
A2GNN	61.9±0.9	61.0±0.7	62.7±1.3	60.1±0.7	63.5±1.9	61.1±0.8	64.3±1.1	61.8±0.8	63.4±1.5	61.0±0.9	63.8±1.8	66.5±1.3
StruRW	67.5±1.2	66.9±0.6	63.7±0.6	61.7±0.7	62.8±1.9	59.1±1.1	69.5±1.1	67.0±1.2	72.2±1.2	65.0±0.9	75.2±0.8	67.9±0.7
PA-BOTH	59.1±1.4	60.9±0.7	59.4±0.7	60.3±1.4	66.6±1.1	60.2±1.3	60.5±0.6	61.9±1.3	65.6±0.7	59.9±0.5	68.1±1.2	59.9±0.5
GAA	64.0±1.6	66.7±0.7	64.1±0.8	63.4±1.4	64.9±1.0	63.4±0.9	69.7±1.5	67.1±1.0	69.9±0.9	65.1±0.6	74.6±1.6	66.8±1.1
TDSS	61.4±0.6	60.4±1.2	60.9±0.5	60.5±1.3	62.3±0.6	60.3±0.5	61.1±0.4	62.1±1.3	61.6±0.8	62.2±0.6	61.4±0.9	61.4±0.4
GOTDA	65.5±1.5	66.4±0.5	62.4±1.4	61.3±0.8	61.1±0.9	58.7±1.3	68.1±1.4	67.4±0.5	69.0±1.2	64.6±0.8	74.8±0.6	67.4±0.6
MASH	62.2±1.0	61.2±1.0	62.2±0.9	60.7±0.6	63.1±0.5	60.6±1.1	66.2±0.8	64.2±1.0	67.9±1.2	63.2±1.7	71.4±0.5	63.8±0.8
GeoAdapt	65.8±0.6	65.5±0.6	62.3±0.7	59.8±0.6	62.3±1.0	57.6±0.7	68.3±0.5	66.8±0.4	68.5±0.4	63.8±0.7	74.0±0.6	67.0±0.4
DisRFM-$\mathbb{E}^n$	65.5±1.4	68.0±1.3	60.7±0.3	61.0±1.6	62.6±1.3	60.7±1.6	68.2±1.0	68.1±1.1	67.9±1.5	64.4±0.5	71.2±1.4	68.3±1.1
DisRFM-$\mathbb{S}^n$	66.3±1.6	67.9±1.2	61.2±1.6	61.5±1.9	65.5±3.6	61.0±0.9	69.1±1.7	69.2±0.4	71.0±1.7	65.1±1.8	74.7±2.0	**69.4±0.5**
DisRFM-$\mathbb{H}^n$	**68.1±1.5**	**69.2±1.8**	**63.4±1.3**	**62.6±1.7**	**66.3±2.2**	**63.9±0.6**	**70.0±1.0**	68.4±1.4	**73.4±2.1**	**67.2±1.0**	**75.3±1.5**	68.9±1.0

Table 24: The classification results (in %) on the ogbg-molhiv dataset under edge density domain shift (source → target). H0, H1, H2, and H3 denote the sub-datasets partitioned with edge density. **Bold** results indicate the best performance.

Methods	H0→H1	H1→H0	H0→H2	H2→H0	H0→H3	H3→H0	H1→H2	H2→H1	H1→H3	H3→H1	H2→H3	H3→H2
WL	45.4	54.9	44.9	46.7	54.4	52.2	53.7	46.0	56.1	45.3	51.1	50.8
GCN	57.5±1.8	58.0±1.5	56.1±2.1	54.3±1.2	54.1±1.4	50.9±1.4	61.2±1.6	54.8±1.9	54.9±2.1	56.6±1.4	65.4±2.0	58.1±1.6
GIN	58.1±1.1	55.3±1.2	49.7±1.5	55.2±2.2	55.5±1.9	50.5±2.0	55.1±1.3	58.8±1.2	56.6±1.6	50.0±1.1	56.8±1.8	56.2±1.6
GMT	50.7±1.0	57.5±1.7	51.1±2.1	58.8±1.5	59.8±1.4	52.0±1.4	57.0±1.8	55.4±1.0	54.0±1.7	54.9±1.0	58.1±1.3	52.0±1.3
CIN	51.8±1.4	54.1±1.7	53.6±1.4	57.3±1.6	56.9±1.7	53.9±1.3	54.5±1.7	63.6±2.1	54.2±1.2	56.9±1.8	57.6±1.3	58.7±1.2
PathNN	57.3±1.1	57.8±2.0	54.8±1.9	53.2±1.1	49.8±1.6	57.9±1.9	60.2±1.6	63.7±1.5	42.2±1.8	58.1±2.2	49.6±1.5	60.5±1.3
dDGM	66.3±0.5	65.9±1.4	61.6±0.7	60.9±1.3	59.9±0.5	59.3±0.7	67.9±1.3	68.1±1.2	70.5±1.4	66.4±0.6	74.3±1.3	66.7±0.7
RieGrace	66.8±1.4	64.5±0.6	62.2±0.8	60.3±1.3	63.7±1.5	59.4±1.0	67.4±1.4	67.2±0.6	70.8±0.6	67.6±1.3	73.1±1.3	68.6±0.6
ProGDM	58.5±1.3	61.5±1.2	59.0±1.2	60.7±1.1	58.2±1.1	58.4±1.4	61.6±1.2	62.8±1.2	60.7±1.1	60.4±1.2	62.3±1.2	61.2±0.5
D-GCN	61.9±0.9	59.3±1.5	59.6±1.6	57.8±1.9	60.6±2.9	56.8±1.9	59.5±1.7	61.0±0.8	65.1±1.0	59.8±2.3	64.5±1.3	61.4±1.4
DEAL	63.5±0.9	64.2±1.4	**63.7±1.0**	60.8±0.9	64.1±2.0	60.5±1.1	67.2±0.3	66.6±0.5	70.1±1.5	64.7±0.8	69.6±2.5	68.6±1.1
SGDA	66.7±1.4	66.0±0.6	63.4±1.4	61.4±1.3	60.6±1.9	59.8±0.8	67.3±1.3	67.8±1.5	71.0±0.6	67.1±1.3	76.1±1.4	69.5±1.4
A2GNN	62.6±1.4	61.1±0.8	63.2±1.5	61.0±1.1	65.1±1.4	60.1±1.3	64.1±1.6	61.7±1.0	64.7±1.0	61.6±0.7	65.5±1.8	63.7±1.8
StruRW	67.1±0.6	66.2±1.5	63.5±1.1	61.8±0.9	64.3±0.9	59.5±1.8	68.1±0.6	67.3±0.6	70.9±0.7	66.7±1.4	76.8±0.6	69.3±1.4
PA-BOTH	60.4±0.7	60.2±0.5	59.8±0.7	59.2±0.6	66.1±1.1	59.1±0.6	61.6±0.7	61.4±0.6	66.1±0.7	60.8±0.8	68.6±1.0	60.6±0.7
GAA	65.1±0.7	65.3±0.6	63.7±0.9	63.3±1.5	64.1±0.9	63.2±0.7	68.2±0.6	68.0±0.7	70.3±1.2	65.6±1.5	75.6±0.6	69.4±0.8
TDSS	62.2±1.5	59.5±0.5	61.5±0.6	59.5±0.5	62.7±0.8	59.5±1.3	61.7±1.3	62.8±1.4	62.0±1.0	63.0±0.4	62.0±1.0	61.7±1.3
GOTDA	65.8±0.7	65.9±0.5	61.6±0.6	60.8±0.3	61.4±1.1	59.1±1.1	67.0±0.5	67.5±0.5	69.8±1.1	65.8±0.5	75.7±0.4	69.2±0.4
MASH	64.8±0.9	59.0±1.4	62.2±1.6	58.4±1.7	64.3±0.6	60.1±1.1	65.0±0.5	63.9±0.6	68.5±0.8	64.4±0.7	72.8±0.9	65.0±1.1
GeoAdapt	67.0±0.8	64.5±0.7	61.4±0.6	59.4±1.4	62.9±2.0	57.9±1.1	66.9±0.6	67.2±1.3	70.8±0.7	65.6±0.8	74.9±1.3	68.3±0.4
DisRFM-$\mathbb{E}^n$	65.6±1.4	64.7±1.5	61.7±1.4	60.6±1.6	65.2±2.3	59.6±0.6	67.3±1.0	67.9±0.3	66.7±1.1	65.5±0.6	72.7±2.2	69.9±0.9
DisRFM-$\mathbb{S}^n$	66.5±0.9	66.0±1.0	62.3±1.4	60.8±1.6	64.6±2.7	**61.0±1.1**	**69.1±1.2**	**70.6±1.7**	71.3±3.2	66.8±1.8	74.4±2.6	69.6±0.7
DisRFM-$\mathbb{H}^n$	**67.4±1.6**	**66.4±0.7**	63.1±1.5	**62.2±1.3**	**66.2±2.2**	60.9±1.0	68.7±0.8	68.4±1.0	**72.1±1.0**	**67.3±1.0**	**77.2±1.6**	**70.5±0.9**

Table 25: Existing assets, licenses, and terms of use.

Asset	Creator / owner	Use in this paper	License / terms	Compliance
PROTEINS, DD	Original dataset papers; TUDataset / PyTorch Geometric access	Graph classification benchmarks and domain-shift construction	License not explicitly specified by upstream provider; TUDataset requests citing the dataset collection and original papers	Cite original papers and TUDataset; access via official loaders; no raw-data redistribution
NCI1	Original dataset paper; TUDataset / PyTorch Geometric access	Graph classification benchmark and domain-shift construction	License not explicitly specified by upstream provider; TUDataset citation terms apply	Cite original paper and TUDataset; access via official loaders; no raw-data redistribution
MUTAGENICITY	Original dataset paper; TUDataset / PyTorch Geometric access	Graph classification benchmark and domain-shift construction	License not explicitly specified by upstream provider; TUDataset citation terms apply	Cite original paper and TUDataset; access via official loaders; no raw-data redistribution
BZR, BZR_MD, COX2, COX2_MD	Original dataset paper; TUDataset / PyTorch Geometric access	Feature-shift graph classification benchmarks	License not explicitly specified by upstream provider; TUDataset citation terms apply	Cite original paper and TUDataset; access via official loaders; no raw-data redistribution
ogbg-molhiv	Open Graph Benchmark / Stanford SNAP	Molecular graph property prediction benchmark	OGB code and loaders are released under the MIT License; dataset is accessed through the official OGB loader and citation terms	Cite OGB; use official splits/loaders; no raw-data redistribution
PyTorch	PyTorch contributors / Meta and contributors	Deep learning framework	BSD-style license	Retain copyright, license notices, and disclaimers in the software environment and released code
PyTorch Geometric	PyG Team	Graph data loading and graph neural network utilities	MIT License	Retain copyright and license notices; cite/use official package
OGB package	OGB / Stanford SNAP	Dataset loader, splits, and evaluator for ogbg-molhiv	MIT License	Retain license notices; use official evaluator and splits
Baseline methods	Original authors of WL subtree, GCN, GIN, CIN, GMT, dDGM, RieGrace, ProGDM, D-GCN, DEAL, SGDA, StruRW, A2GNN, PA-BOTH, GAA, TDSS, GOTDA, MASH, GeoAdapt, and related methods	Experimental comparison	Original papers and, if official code is used, the corresponding code licenses	Cite original papers; use our own implementations unless otherwise stated; do not redistribute third-party code

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately summarize the paper's scope and contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the scope and limitations of the work. The theoretical analysis is explicitly conditional and does not claim universal topology–semantics identifiability or global convergence for nonconvex training. The conclusion also identifies future work, including adaptive curvature learning and extensions to heterogeneous and generative graph tasks.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper provides formal theoretical results with explicit assumptions, including structural failure of unconditional domain confusion, first-order orthogonality of polar endpoint losses, radial structural proxy bounds, angular margin preservation, polar flow endpoint stability, and a target-risk bound. Full proofs are provided in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides the model formulation, objectives, datasets, domain-shift construction, baselines, hyperparameters, training protocol, evaluation metrics, and implementation details needed to reproduce the main experimental results. Results are reported across multiple public graph benchmark datasets and averaged over five independent runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All the datasets and baseline codes are publicly available, and we introduce them in Appendix H and I.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the experimental datasets, domain-shift settings, baselines, preprocessing protocols, backbone model, optimizer, learning rate, weight decay, hidden dimension, number of GNN layers, curvature, confidence threshold, loss weights, evaluation metrics, and number of independent runs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The main experimental tables report results with error bars using the "±" format, and the implementation details state that results are averaged over five independent runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide GPU consumption and training time for DisRFM in Appendix. K.4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research is an algorithmic study on graph domain adaptation using public benchmark datasets. It does not involve human subjects, private data, scraped sensitive content, or high-risk model release, and no deviation from the NeurIPS Code of Ethics is indicated.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The appendix includes a dedicated Broader Impacts section discussing potential positive impacts in robust graph learning and scientific graph applications, as well as potential negative impacts related to high-stakes misuse, biased graph structures, pseudo-label errors, privacy-sensitive relational data, and computational cost.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper poses no such risks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The Appendix. L includes an "Existing Assets, Licenses, and Terms of Use" section that credits the original creators and owners of all datasets, software packages, and baseline assets used in the paper.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.