BEYOND PAIRWISE CO-OCCURRENCE: GALOIS CLOSURE CURVATURE FOR HIGHER-ORDER RELATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Hypergraphs represent interactions among arbitrary sets of entities, yet hypergraph learning methods can often consume incidence primarily through pairwise co-occurrence, i.e., which nodes appear together. This reduction can obscure set-level dependencies: two hypergraphs may share every node degree and pairwise co-occurrence count while differing in which node sets necessarily imply others. We ask how the geometry of such higher-order dependencies can be characterized beyond observed connectivity. Our key observation is that Galois closure, which encodes implications among nodes induced by incidence, reveals this geometry through how closure states contract or expand under shared structural perturbations, following the local contraction principle underlying Ollivier–Ricci curvature. Building on this observation, we introduce **Galois Closure Curvature (GCC)**, which quantifies the perturbation response of closure states, together with a curvature profile that retains contraction, expansion, and variability otherwise lost through cancellation in the scalar curvature. We show that this profile can distinguish hypergraphs with identical pairwise co-occurrence statistics even when their scalar GCC coincides, and characterize conditions under which the closure response vanishes on purely pairwise structures. By lifting the profile to node–hyperedge incidences, GCC provides a context-dependent structural representation of higher-order implication geometry. Across controlled implication-recovery tasks and established hypergraph benchmarks, GCC achieves the best mean rank among curvature descriptors on Hy-WL and is the most accurate on collections whose classes differ only in planted implications. The code is available at `https://anonymous.4open.science/r/GCC-7A1C/`.

1 INTRODUCTION

Graphs are the standard abstraction for pairwise relations, yet many real-world systems are driven by interactions among three or more entities at once, including collaboration, biological interactions, and chemical reactions (Battiston et al., 2020; Benson et al., 2018). Hypergraphs represent such higher-order interactions directly, motivating a broad family of hypergraph neural networks (HGNNs) (Feng et al., 2019; Chien et al., 2022; Wang et al., 2023; Duta et al., 2023). However, representing data as a hypergraph does not by itself guarantee that higher-order information is exploited. Chitra & Raphael (2019) showed that random walks on hypergraphs with edge-independent vertex weights are equivalent to random walks on an edge-weighted clique graph, implying that a broad class of random-walk-based operators can collapse to pairwise structure. More recently, graph-level architectures applied to hypergraph expansions were found to often outperform hypergraph-level architectures even on data naturally represented as hypergraphs (Pellegrin et al., 2025). Together, these findings suggest that higher-order incidence can often be consumed primarily through *co-occurrence*: which nodes appear together, rather than how their joint participation constrains the surrounding structure.

What is lost under such a reduction is *set-level dependency*. Two hypergraphs can agree on every node degree and every pairwise co-occurrence count, yet differ in whether the presence of a set of nodes necessarily implies another node. For example, in one hypergraph, every interaction containing both a and b may necessarily contain c, yielding the implication $a \wedge b \Rightarrow c$, whereas another hypergraph with identical pairwise co-occurrence statistics may exhibit no such dependency (Proposition 2). Discrete curvature provides a natural language for describing local structural geometry and

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

has been generalized from graphs to hypergraphs through transport- and combinatorial-based constructions (Eidi & Jost, 2020; Coupette et al., 2023; Leal et al., 2021; Yang et al., 2025). These formulations, however, derive geometry from node-level transport or local combinatorial incidence statistics rather than from dependencies among node sets. This raises a fundamental question: *beyond observed connectivity, how can we define the geometry of dependencies intrinsic to higher-order incidence relations?*

We address this question through *Galois closure*. Hypergraphs and formal contexts are formally connected through their binary incidence relations (Cattaneo et al., 2016), and the incidence matrix of a hypergraph can be used to construct a concept lattice whose structure reflects hyperedge intersections (Rawson et al., 2023). Under this view, the Galois closure of a node set contains the nodes necessarily implied by its observed incidence pattern, thereby exposing dependencies that are not represented by pairwise co-occurrence alone. Closure itself, however, describes a static implication state and does not indicate how the relationship between two such states changes under structural variation. Our key observation is that the *contraction and expansion* of closure states under perturbation admit a geometric interpretation analogous to the local contraction underlying Ollivier–Ricci curvature (Ollivier, 2009). This perspective is conceptually related to formal concept stability, which measures how frequently the intent of a concept is preserved across subsets of its extent (Kuznetsov, 2007), but differs fundamentally in that we compare the signed responses of pairs of closure states under shared structural perturbations.

Building on this observation, we propose **Galois Closure Curvature (GCC)**, a response-based curvature that quantifies the geometry of higher-order incidence structure through perturbation-induced deformation of Galois closure states. GCC summarizes the average response as a scalar curvature, while its underlying responses can be characterized through contraction, expansion, and variability. This decomposition retains signed and dispersion information that may disappear in the scalar mean: hypergraphs that are indistinguishable from pairwise co-occurrence and share the same scalar GCC can nevertheless exhibit different response structures (Proposition 2). Our perspective is therefore that *the geometry of higher-order relations is determined not only by which nodes co-occur, but by how their incidence-induced implications reorganize under structural perturbation.*

Our contributions are summarized as follows:

- We propose **GCC**, a curvature that characterizes higher-order implication geometry through the contraction and expansion of Galois closure states under shared structural perturbations.
- We show that the GCC profile is not determined by node degrees and pairwise co-occurrence counts, and that scalar averaging can erase differences between profiles. We also establish a zero-response result for simple graphs with minimum degree at least two.
- We evaluate GCC as a hypergraph descriptor under a common distance-based classification protocol, comparing it with size and degree statistics and existing curvature descriptors on Hy-WL and planted implications.

2 Related Work

2.1 Hypergraph Representation Learning

Hypergraph neural networks have evolved from spectral convolutions and graph-based approximations (Feng et al., 2019; Yadati et al., 2019) to node–hyperedge message passing (Huang & Yang, 2021; Chien et al., 2022), equivariant diffusion (Wang et al., 2023), and sheaf-based architectures (Duta et al., 2023). Despite these advances, explicitly representing higher-order interactions does not necessarily ensure that models exploit information beyond pairwise structure: random walks with edge-independent vertex weights admit an equivalent weighted clique-graph representation (Chitra & Raphael, 2019), and graph architectures operating on hypergraph expansions can remain competitive with, or outperform, hypergraph-specific architectures (Pellegrin et al., 2025). Related expressivity analyses further characterize limitations of conventional message-passing HGNNs and motivate higher-dimensional constructions (Huang & Yang, 2021; Zhang et al., 2025). These findings highlight a broader limitation of connectivity-driven representations: higher-order inci-

dence may contain structural dependencies that are not fully characterized by pairwise co-occurrence or its induced expansions.

2.2 Discrete Curvature

Discrete curvature provides a geometric characterization of local relational structure, with Ollivier–Ricci curvature measuring contraction under probability transport (Ollivier, 2009) and related notions being used to identify graph bottlenecks, guide rewiring, and construct structural encodings (Topping et al., 2022; Fesser & Weber, 2024). Extensions to hypergraphs include Ollivier-type formulations (Eidi & Jost, 2020; Coupette et al., 2023), Forman–Ricci curvature (Leal et al., 2021), and more recent curvature constructions and learning applications (Yang et al., 2025; Pellegrin et al., 2025; Zhou et al., 2026). While these approaches demonstrate the utility of geometry for higher-order learning, their geometric quantities are primarily derived from transport, diffusion, or local combinatorial structure. Consequently, the geometry induced by *set-level implication dependencies* in the incidence relation remains comparatively unexplored.

2.3 Formal Concept Analysis

Formal Concept Analysis (FCA) represents binary object–attribute relations through Galois connections and concept lattices (Ganter & Wille, 1999). Its connection to hypergraphs is well established: hypergraphs can be represented as formal contexts (Cattaneo et al., 2016), and their incidence structure gives rise to concept-lattice representations closely related to hyperedge intersections (Rawson et al., 2023). Galois lattices have consequently been used to characterize social and epistemic structures and as structural priors for learning (Freeman, 1996; Roth & Bourgine, 2006; Shao et al., 2023). Concept stability further studies the robustness of a formal concept by measuring how its intent is preserved across subsets of its extent (Kuznetsov, 2007). These works establish closure and stability as meaningful descriptors of incidence-induced dependency, but do not characterize the *signed geometric relation between closure states*—in particular, whether their discrepancy contracts or expands under a shared structural perturbation.

3 Preliminaries

3.1 Hypergraph Learning

Let $\mathcal{H} = (V, E)$ denote a hypergraph with nodes $V = \{v_1, \dots, v_n\}$ and hyperedges $E = \{e_1, \dots, e_m\}$, where each $e \subseteq V$. The incidence relation is $I = \{(e, v) \in E \times V \mid v \in e\}$. Unlike ordinary graph edges, a hyperedge can directly associate an arbitrary number of nodes and thereby represent higher-order interactions.

We write $\mathbf{B} \in \{0,1\}^{m \times n}$ for the incidence matrix, with $\mathbf{B}_{ev} = 1$ iff $v \in e$. The matrix $\mathbf{B}^\top \mathbf{B}$ encodes node degrees $d(v) = (\mathbf{B}^\top \mathbf{B})_{vv}$ on its diagonal and pairwise co-occurrence counts on its off-diagonal, and coincides with the weighted clique expansion of $\mathcal{H}$. Statistics that are functions of $\mathbf{B}^\top \mathbf{B}$ therefore describe $\mathcal{H}$ only through degrees and pairwise co-occurrence.

Hypergraph neural networks commonly alternate two aggregation stages,

$$\mathbf{h}_e = \mathrm{AGG}_{v \in e}\left(\psi_{V \to E}(\mathbf{h}_v)\right), \qquad \mathbf{h}_v^+ = \mathrm{AGG}_{e \ni v}\left(\psi_{E \to V}(\mathbf{h}_e)\right), \tag{1}$$

where the message functions ψ and the aggregation rule AGG are specific to the underlying backbone.

3.2 Galois Closure on Hypergraphs

We view $\mathcal{H}$ as the formal context $\mathcal{K}_\mathcal{H} = (E, V, I)$ with eIv iff $v \in e$. For $X \subseteq V$ and $Y \subseteq E$,

$$X' = \{e \in E \mid X \subseteq e\}, \qquad Y' = \bigcap_{e \in Y} e, \qquad \phi(X) = X'' = \bigcap_{e \colon X \subseteq e} e, \tag{2}$$

with $\bigcap_{e \in \emptyset} e = V$. The closure operator ϕ induces implications $X \Rightarrow Y$ iff $Y \subseteq \phi(X)$: every interaction involving X also involves Y. Its closed sets form the concept lattice of $\mathcal{K}_\mathcal{H}$, isomorphic

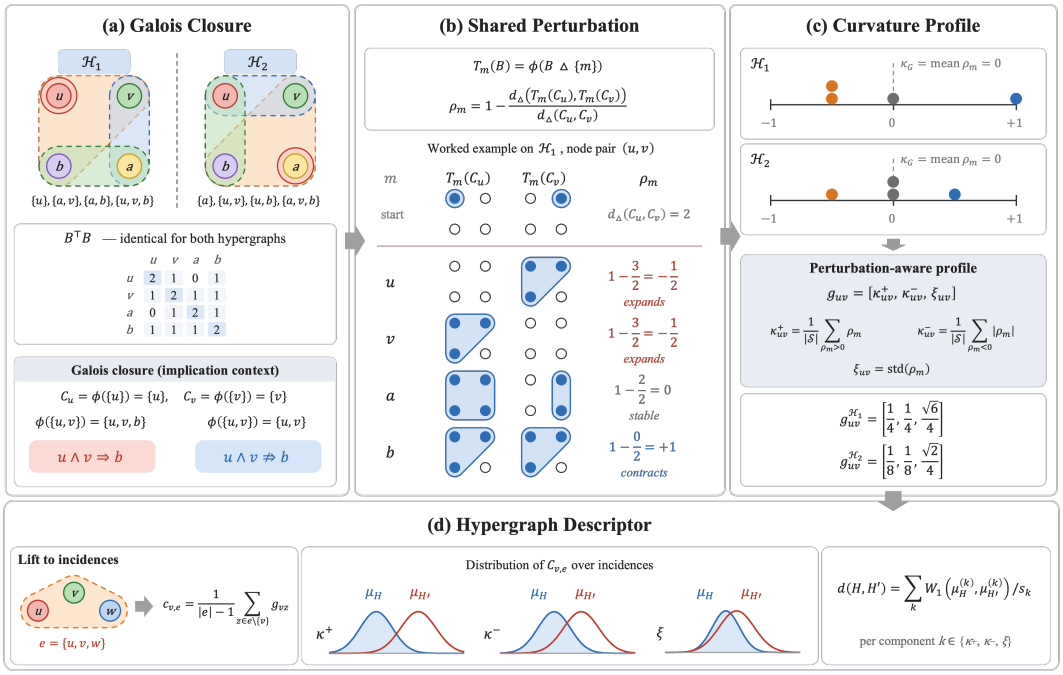


Figure 1: Overview of Galois Closure Curvature (GCC). (a) Galois closure reveals higher-order implications beyond pairwise co-occurrence. (b) Shared perturbations measure whether closure states contract, expand, or remain stable. (c) The curvature profile preserves signed responses and variability lost by scalar averaging. (d) Incidence-level profiles form a hypergraph descriptor via distributional comparison.

to the intersection structure of the hyperedges (Rawson et al., 2023). Since ϕ is a set function, $\phi(\{u, v\})$ is in general not determined by $\phi(\{u\})$, $\phi(\{v\})$, or the co-occurrence of u and v.

We associate each node v with its singleton closure $C_v = \phi(\{v\})$, an *implication state* containing the nodes that necessarily accompany v across all its hyperedges, and compare closure states by $d_\triangle(B, C) = |B \triangle C|$. However, static closures can coincide across hypergraphs whose higher-order dependencies differ (Proposition 2); such differences surface only when closure states respond to structural perturbation. This perturbation response is the basis of the proposed curvature.

3.3 Ollivier–Ricci Curvature

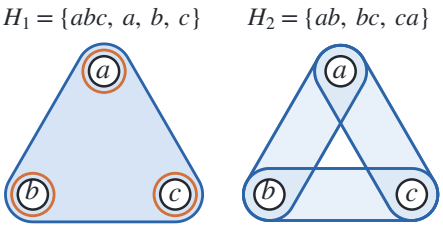
For a metric space $(\mathcal{X}, d)$ with a probability measure m_x at each point, the Ollivier–Ricci curvature is

$$\kappa(x, y) = 1 - \frac{W_1(m_x, m_y)}{d(x, y)}, \tag{3}$$

where W_1 is the 1-Wasserstein distance (Ollivier, 2009). Positive (negative) curvature means the local measures are closer (farther) than their centers, i.e., contraction (expansion). We adopt this view of curvature as a local contraction coefficient, replacing random-walk measures on nodes with closure states under perturbation.

4 Method

Our central premise is that higher-order structure is characterized not only by which nodes co-occur, but by how their incidence-induced implications reorganize under structural perturbation. We introduce *Galois Closure Curvature (GCC)*, which measures how the discrepancy between two



	H_1	H_2
Degree / co-occurrence	2 / 1	2 / 1
Singleton closure C_v	$\{v\}$	$\{v\}$
HFRC at a	0	0
CIHI at a	1	1
GCC-κ at (a, b)	0	0
GCC at (a, b)	$[.33, .33, .71]$	**0**

Figure 2: Illustrative pair from Proposition 2. The hypergraphs H_1 and H_2 are identical in node degrees, pairwise co-occurrence counts, singleton closures, HFRC, CIHI, and the scalar curvature κ_G. In contrast, the GCC profile captures the higher-order implication $a \wedge b \Rightarrow c$ present in H_1, whose contribution is eliminated by aggregation into κ_G.

Galois closure states changes under shared feasible perturbations (§4.1). As illustrated in Figure 1, GCC summarizes these responses as a profile of contraction, expansion, and variability that refines the scalar curvature, captures information invisible to co-occurrence statistics (§4.2), and lifts the profile to node–hyperedge incidences for aggregation into hypergraph descriptors (§4.3).

4.1 Galois Closure Curvature

Closure perturbation. For a closed set $B = \phi(B)$ and a node $m \in V$, we define a local perturbation by toggling m and restoring closure,

$$T_m(B) = \phi(B \triangle \{m\})\,. \tag{4}$$

The symmetric difference inserts m if $m \notin B$ and removes it otherwise; re-applying ϕ propagates this modification through the implication structure, so $T_m(B)$ is again a closed state. To compare two states under matched conditions and exclude configurations never supported by the data, we restrict to their common feasible perturbations,

$$\mathcal{S}(B, C) = \{m \in V : \mathrm{supp}(B \triangle \{m\}) > 0,\ \mathrm{supp}(C \triangle \{m\}) > 0\}\,, \tag{5}$$

where $\mathrm{supp}(X) = |\{e \in E \mid X \subseteq e\}|$.

Curvature. For distinct closed states $B \neq C$ and $m \in \mathcal{S}(B, C)$, the normalized response and the *Galois Closure Curvature* are

$$\rho_m(B, C) = 1 - \frac{d_\triangle(T_m(B), T_m(C))}{d_\triangle(B, C)}, \qquad \kappa_{\mathrm{G}}(B, C) = \mathbb{E}_{m \in \mathcal{S}(B,C)}[\rho_m(B, C)]\,. \tag{6}$$

Normalizing by the original discrepancy makes ρ_m a *relative* change: $\rho_m > 0, = 0$, or < 0 according to whether the shared perturbation contracts, preserves, or expands the discrepancy. Eq. 6 is the Ollivier form of equation 3, with closure states in place of random-walk measures and shared perturbations in place of transport. For a node pair we set $\kappa_{\mathrm{G}}(u, v) := \kappa_{\mathrm{G}}(C_u, C_v)$ with $C_v = \phi(\{v\})$.[1]

Proposition 1 (Curvature range and sign)**.** *For closed $B \neq C$ with $\mathcal{S}(B, C) \neq \emptyset$, κ_{G} is symmetric and satisfies $1 - n/d_\triangle(B, C) \leq \kappa_{\mathrm{G}}(B, C) \leq 1$, with $\kappa_{\mathrm{G}}(B, C) = 1$ iff $T_m(B) = T_m(C)$ for all $m \in \mathcal{S}(B, C)$. Moreover, $\kappa_{\mathrm{G}} > 0$ (< 0) iff shared perturbations decrease (increase) the closure discrepancy on average.*

Thus GCC inherits the defining property of Ollivier curvature, namely that its sign measures local contraction versus expansion, and reaches its maximum exactly when shared perturbations collapse two implication states into one (proof in Appendix A.2).

4.2 Perturbation-Aware Curvature Profiles

Because κ_{G} is a mean, strong contraction and expansion can cancel. We therefore retain the signed parts and dispersion of the same responses. With $\mathcal{S}_{uv} = \mathcal{S}(C_u, C_v)$ and $\rho_m = \rho_m(C_u, C_v)$,

$$\kappa^+_{uv} = \mathbb{E}_{m \in \mathcal{S}_{uv}}\big[[\rho_m]_+\big], \quad \kappa^-_{uv} = \mathbb{E}_{m \in \mathcal{S}_{uv}}\big[[-\rho_m]_+\big], \quad \xi_{uv} = \Big(\mathbb{E}_{m \in \mathcal{S}_{uv}}\big[(\rho_m - \kappa_{\mathrm{G}}(u, v))^2\big]\Big)^{1/2}, \tag{7}$$

[1] If $C_u = C_v$ or $\mathcal{S}(C_u, C_v) = \emptyset$, the pair is assigned a zero profile; see Appendix A.7.

and define the *Galois curvature profile* $\mathbf{g}_{uv} = [\kappa^+_{uv}, \kappa^-_{uv}, \xi_{uv}]$. Since $x = [x]_+ - [-x]_+$, we have $\kappa_{\mathrm{G}}(u, v) = \kappa^+_{uv} - \kappa^-_{uv}$: the profile refines, rather than replaces, the scalar curvature. The following result shows that this refinement is necessary, and that both are invisible to co-occurrence.

Proposition 2 (Co-occurrence blindness)**.** *Let* $V = \{a, b, c\}$ *and*

$$\mathcal{H}_1 = \big\{\{a, b, c\}, \{a\}, \{b\}, \{c\}\big\}, \qquad \mathcal{H}_2 = \big\{\{a, b\}, \{b, c\}, \{c, a\}\big\}.$$

Then $\mathbf{B}_1^\top \mathbf{B}_1 = \mathbf{B}_2^\top \mathbf{B}_2$ *and all singleton closures coincide, so any statistic determined by* $\mathbf{B}^\top \mathbf{B}$ *is identical on* $\mathcal{H}_1$ *and* $\mathcal{H}_2$*. The scalar curvature also coincides, yet the profiles differ: for every node pair* (u, v)*,*

$$\mathcal{H}_1: \quad \kappa_{\mathrm{G}}(u, v) = 0, \qquad \mathbf{g}_{uv} = \big[\tfrac{1}{3},\ \tfrac{1}{3},\ \tfrac{1}{\sqrt{2}}\big],$$
$$\mathcal{H}_2: \quad \kappa_{\mathrm{G}}(u, v) = 0, \qquad \mathbf{g}_{uv} = \mathbf{0}.$$

In $\mathcal{H}_1$, jointly engaging a and b forces c ($a \wedge b \Rightarrow c$), so some perturbations collapse the two states to V (contraction) while others push them apart (expansion); these cancel in the mean but not in the profile. In $\mathcal{H}_2$ no such implication exists and every response is zero. This is not specific to the example: on any simple graph with minimum degree at least two, all responses vanish (Figure 2; Appendix A.5, A.4).

Incidence-level lifting. Message passing operates on node–hyperedge incidences, so we aggregate pairwise profiles within each hyperedge,

$$\mathbf{c}_{v,e} = \frac{1}{|e \setminus \{v\}|} \sum_{u \in e \setminus \{v\}} \mathbf{g}_{vu}, \tag{8}$$

with $\mathbf{c}_{v,e} = \mathbf{0}$ for $|e| = 1$. Because the averaging set depends on e, the same node can satisfy $\mathbf{c}_{v,e_i} \neq \mathbf{c}_{v,e_j}$: its structural description varies across the higher-order contexts it participates in, rather than being a single global descriptor.

4.3 GCC as a Hypergraph Descriptor

GCC assigns a profile g_{uv} to every co-occurring node pair and, by Eq. (8), a profile $c_{v,e}$ to every incidence. We summarize a hyperedge e with $|e| \geq 2$ by its mean incidence profile $c_e = \frac{1}{|e|} \sum_{v \in e} c_{v,e}$, whose $\kappa^+ - \kappa^-$ is the mean of κ_G over the node pairs of e, and a hypergraph H by the empirical distribution μ_H of $\{c_e : e \in E,\ |e| \geq 2\}$; hyperedges of size one contain no node pair and are left out. Two hypergraphs are compared by

$$d(\mathcal{H}, \mathcal{H}') = \sum_{k=1}^{3} \frac{W_1\Big(\mu^{(k)}_{\mathcal{H}}, \mu^{(k)}_{\mathcal{H}'}\Big)}{s_k}, \tag{9}$$

where $\mu^{(k)}$ is the distribution of the k-th component and s_k its standard deviation over the hyperedges of the hypergraphs the kernel is fitted on (the training hypergraphs in classification), so that the three components enter on a common scale; GCC-κ uses the single value $\kappa^+ - \kappa^-$.

5 Experiments

We evaluate GCC as a descriptor of whole hypergraphs. We compare it with existing curvatures on a standard hypergraph classification benchmark (§5.2), test on constructed collections whether it detects set-level implications that degree and co-occurrence statistics cannot (§5.3), measure its computation time (§5.4), and illustrate on one collection how the classes separate (§5.5).

5.1 Tasks and Baselines

Hypergraph classification. We use Hy-WL, the ten collections of Feng et al. (2024) for classifying whole hypergraphs: four synthetic collections whose classes are the structural templates that generated them (RHG-Table, RHG-Pyramid, RHG-3, RHG-10), and six real ones labelled by metadata (IMDB movie collaborations by form or genre, Steam players, Twitter users; Appendix B). An SVM uses the kernel $\exp(-d/\gamma)$ with d from Eq. (9) and γ the median of d over training pairs, with 5 seeds $\times$ 5-fold cross-validation as in Feng et al. (2024); s_k and γ come from the training folds.

Method	RHG-T	RHG-P	RHG-3	RHG-10	IDFor	IDGen	IWFor	IWGen	Steam	Twitt	Rank ↓	#B ↑
Majority	50.4	50.7	33.5	10.8	42.7	51.3	45.7	26.2	50.0	59.2	–	–
Size	95.5	48.9	99.9	61.5	65.5	73.0	45.0	53.9	56.7	69.0	–	–
Degree	99.5	95.6	98.7	87.9	64.0	75.3	46.8	45.6	59.3	59.1	–	–
HFRC	99.1	95.6	96.5	83.9	66.4	**78.6**	45.7	49.8	60.1	60.2	5.65	1
CIHI	99.4	95.6	98.6	84.2	59.9	72.9	45.9	40.8	53.5	58.8	7.65	0
ORCHID	99.2	95.7	97.2	73.0	64.6	74.9	45.2	49.4	53.0	58.0	7.75	0
CE-FRC	99.4	95.6	99.4	76.2	63.7	74.1	44.9	45.7	52.6	59.0	7.85	0
CE-ORC	99.2	95.7	97.2	71.4	64.6	74.9	45.3	49.3	53.0	58.0	7.90	0
CE-AFRC$_3$	99.2	95.7	97.0	80.2	63.7	72.1	44.0	47.3	52.6	60.2	8.50	0
ETR	**100.0**	95.7	97.5	82.0	64.3	76.8	**49.4**	46.8	63.1	59.7	4.40	2
HRCD	99.6	95.7	98.2	75.9	64.7	76.7	47.6	**54.4**	62.1	58.8	4.90	1
BE	99.4	95.6	96.1	81.4	66.6	75.9	45.7	42.5	62.6	58.6	6.90	0
Res.	99.2	95.6	99.1	85.2	61.8	72.3	44.3	41.3	53.6	58.0	8.50	0
GCC-κ	**100.0**	**96.0**	95.4	81.2	66.7	77.2	43.4	43.3	62.6	60.2	5.20	2
GCC	**100.0**	95.8	**99.5**	**89.8**	**68.3**	77.5	45.2	43.1	**64.5**	**60.8**	**2.80**	**6**

Table 1: SVM classification accuracy (%) on the Hy-WL collections. Rank is the mean rank across twelve curvature measures using unrounded values, and #B counts the collections on which each measure performs best, including ties. **Bold** and underlined denote the best and second-best curvature results. Majority, Size, and Degree are unranked controls.

Twin collections. Since hyperedge sizes or node degrees alone already predict several Hy-WL classes (Table 1), we also build collections whose classes differ only in set-level implications. Enumerating hypergraphs on six nodes gives five pairs of motifs (M_0, M_1) with the same degrees, pairwise co-occurrence counts and hyperedge sizes that differ in which implications $u \wedge v \Rightarrow w$ hold. We insert $K \in \{1, 3, 5\}$ copies of M_0 into a random hypergraph (24 hyperedges of size 2–4) and copies of M_1 on the same nodes into an identical copy of it; these *twins* agree exactly on degrees, co-occurrence and hyperedge sizes. Each motif type and K gives 500 twin pairs (Appendix C). We classify as above, keeping twins in the same fold, and also cluster without labels by kernel PCA to two dimensions and k-means (Yang et al., 2025), scored by the adjusted Rand index (ARI).

Computation time. We time each curvature on four single hypergraphs that are standard for node classification and far larger than those of Hy-WL: Cora-CC, Citeseer-CC, Pubmed-CC and Cora-CA (Yadati et al., 2019), with 4,256 to 123,827 co-occurring node pairs (Appendix D).

Baselines. We compare GCC (the profile) and GCC-κ (the scalar κ_G) with ten curvatures: HFRC (Leal et al., 2021), CIHI (Yang et al., 2025), ORCHID (Coupette et al., 2023), HyperRCD (HRCD; Tian et al., 2025), Edge-transport Ricci (ETR; Hacquard, 2026) and Bakry–Émery curvature (BE; Bakry & Émery, 1985), defined on the hypergraph or its line graph, and the Forman (Forman, 2003), Ollivier (Ollivier, 2009), augmented Forman (Fesser et al., 2024) and resistance (Devriendt & Lambiotte, 2022) curvatures of the clique expansion (CE-FRC, CE-ORC, CE-AFRC$_3$, Res.). Hyperedge size and the mean node degree of a hyperedge serve as non-curvature controls. Every descriptor enters Eq. (9) with one value per hyperedge, except GCC, which uses the three components of c_e.

5.2 Hypergraph Classification Benchmark

GCC achieves the best mean rank among the twelve curvatures on Hy-WL (Table 1; 2.80 vs. 4.40 for ETR) and the highest accuracy on six of the ten collections, including one tie. The advantage is most pronounced on the synthetic collections: GCC or GCC-κ performs best among the curvatures on all four, with GCC reaching 89.8% on RHG-10 compared with 85.2% for the next curvature and 87.9% for the degree control. GCC-κ alone remains competitive (mean rank 5.20), indicating that the closure response itself captures useful structure. Separating contraction, expansion, and variability further improves RHG-10 by 8.6 points, where contraction and expansion cancel in κ_G for 35% of node pairs with nonzero profiles. GCC performs less favorably on the two IMDB writer collections, where 32% and 40% of co-occurring node pairs receive zero profiles and about a quarter of hypergraphs have an identical copy under another label (Appendix B).

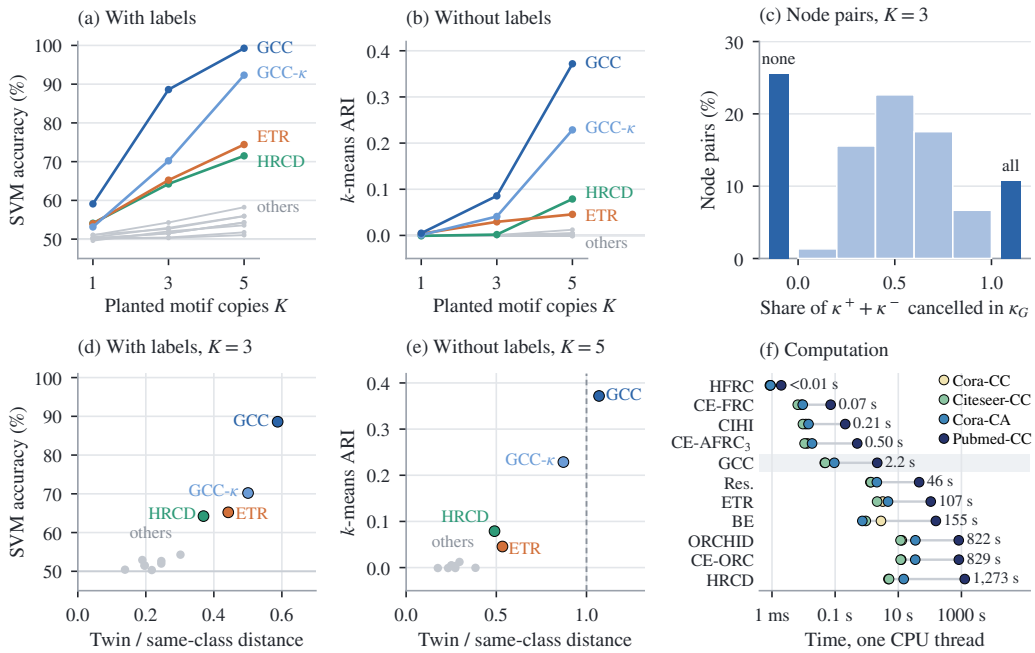


Figure 3: Performance on twin-collection benchmarks and computational cost. (a,b) Classification accuracy and unsupervised adjusted Rand index (ARI) across the number of inserted motif copies, averaged over motif types; grey curves show competing curvature measures. (c) Fraction of the total response magnitude, $\kappa^+ + \kappa^-$, cancelled in scalar κ_G for node pairs with nonzero profiles at three motif copies. (d,e) Classification accuracy at three copies and ARI at five copies versus the twin-to-same-class distance ratio (Appendix C). (f) Single-thread CPU computation time across hypergraphs of increasing size (Appendix D).

5.3 Implications beyond Co-occurrence

On the planted implications Figure 3-(a), (b), averaged over the motif types), GCC is the most accurate curvature at every number of copies: with three and five copies it reaches 88.6% and 99.3%, against 70.2% and 92.3% for GCC-κ and at most 65.2% and 74.4% for the existing curvatures, and without labels only GCC and GCC-κ substantially recover the classes (mean ARI 0.37 and 0.23 with five copies, at most 0.08 otherwise). Figure 3-(d), (e) relates both to as the median distance between twins divided by that between hypergraphs of the same class: accuracy increases with this ratio, with the four most accurate curvatures exhibiting the four largest ratios (GCC 0.59, GCC-κ 0.50, ETR 0.44, HRCD 0.37 with three copies), while clustering succeeds only as the ratio approaches one, reached only by GCC (1.07) and GCC-κ (0.87) with five copies.

GCC and GCC-κ differ only in how the same closure responses are summarized. Figure 3-(c) quantifies the information lost by scalar aggregation: with three copies, 74.5% of the node pairs with a nonzero profile exhibit both contraction and expansion, $\kappa_G = \kappa^+ - \kappa^-$ cancels on average 58% of their response $\kappa^+ + \kappa^-$, and for 10.7% of the nonzero pairs all of it. By retaining both components, the profile reduces the error of the scalar from 29.8% to 11.4% with three copies and from 7.7% to 0.7% with five.

5.4 Computational Cost

Computing a curvature requires evaluating it on every co-occurring node pair or hyperedge of a hypergraph; we measure this computation, excluding data loading, on one CPU thread (Figure 3-(f); Appendix D). On Pubmed-CC, GCC takes 2.2 s, ranking fifth among the eleven curvatures. Like the Ollivier-type curvatures, GCC measures how the distance between two states changes after a local step, but the perturbed states are paired by the shared perturbation, eliminating the need to solve an optimal-transport problem; each response is instead computed by exact set operations. GCC is

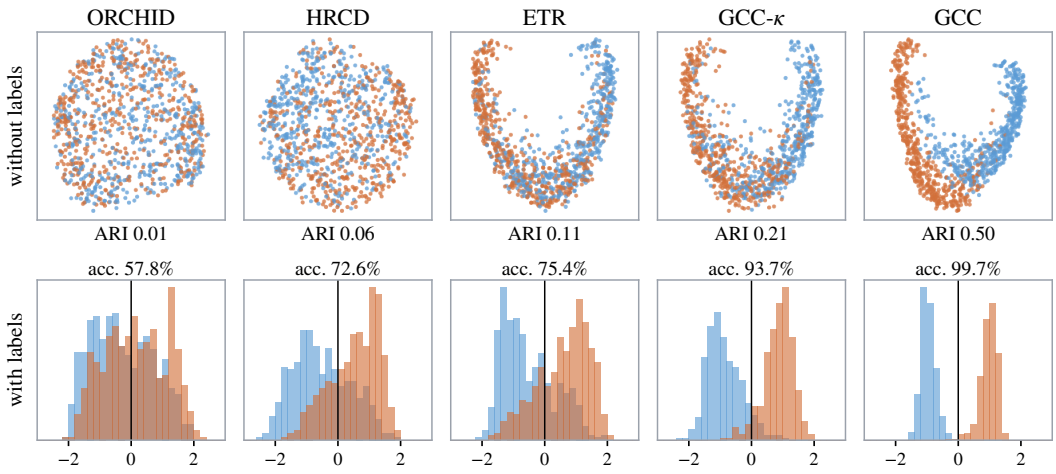


Figure 4: Visualization and classification results for a representative twin collection consisting of five copies of motif type 3 and 1,000 hypergraphs colored by class. Columns correspond to the three existing curvature measures with the highest ARI on this collection, GCC-κ, and GCC. Top: kernel PCA embeddings obtained from the kernel defined in Eq. (9), together with the corresponding k-means ARI. Bottom: held-out SVM decision values, where 0 denotes the decision boundary, together with classification accuracy averaged over 5 seeds and 5 folds. Results for the remaining motif types are provided in Appendix C.

therefore about 380 times faster than ORCHID and CE-ORC, 590 times faster than HRCD, and 50 times faster than ETR, which solve such a problem for every pair. The four faster curvatures, HFRC, CIHI, CE-FRC and CE-AFRC$_3$, are closed-form expressions in node degrees, hyperedge sizes and triangle counts that require only combinatorial counting; these quantities are identical between twins at every node, and on the planted implications these curvatures remain within six percentage points of chance.

5.5 Qualitative Analysis

Figure 4 shows one collection of §5.3, five copies of motif type 3, the only motif type on which every column shown has an ARI above zero; the columns are the three existing curvatures with the highest ARI (ORCHID, HRCD, ETR), GCC-κ and GCC, and colour marks the class. The top row is the kernel-PCA embedding of the kernel of Eq. (9), computed without labels; the bottom row shows the SVM decision values of held-out hypergraphs, with 0 as the decision boundary. From left to right the classes separate in both rows (ARI 0.01 to 0.50, accuracy 57.8 to 99.7%). ETR arranges the hypergraphs along the same arc as GCC, but its classes overlap along the bottom of the arc (ARI 0.11) and its decision values overlap (75.4%): along its first kernel-PCA direction the classes differ by 0.9 within-class standard deviations, against 2.3 for GCC. For GCC the two classes occupy the two arms and their decision values barely overlap: the inserted implication is the main axis along which GCC arranges the hypergraphs, so it can be found even without labels.

6 Conclusion

We introduced Galois Closure Curvature (GCC), a response-based curvature that characterizes higher-order implication structure through the perturbation response of Galois closure states. Its curvature profile preserves contraction, expansion, and variability that can be lost through scalar averaging. We showed theoretically that the profile distinguishes hypergraphs with identical degree and pairwise co-occurrence statistics even when their scalar GCC coincides. Empirically, GCC achieved the best mean rank on Hy-WL and most effectively recovered planted implications invisible to pairwise statistics, while remaining substantially cheaper than transport-based curvatures. These results highlight perturbation-induced changes in incidence implications as a useful geometric characterization of higher-order structure.

AI USE STATEMENT

We used generative AI tools to assist with drafting and editing parts of the manuscript and with generating and editing software code. All AI-assisted text was reviewed and revised by the authors, and AI-assisted code was verified and tested for correctness. We take responsibility for the final content of this work.

REFERENCES

Dominique Bakry and Michel Émery. Diffusions hypercontractives. In *Séminaire de Probabilités XIX 1983/84*, volume 1123 of *Lecture Notes in Mathematics*, pp. 177–206. Springer, 1985.

Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, Jean-Gabriel Young, and Giovanni Petri. Networks beyond pairwise interactions: Structure and dynamics. *Physics Reports*, 874:1–92, 2020. doi: 10.1016/j.physrep.2020.05.004.

Austin R. Benson, Rediet Abebe, Michael T. Schaub, Ali Jadbabaie, and Jon Kleinberg. Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*, 115 (48):E11221–E11230, 2018. doi: 10.1073/pnas.1800683115.

Gianpiero Cattaneo, Giampiero Chiaselotti, Davide Ciucci, and Tommaso Gentile. On the connection of hypergraph theory with formal concept analysis and rough set theory. *Information Sciences*, 330:342–357, 2016. doi: 10.1016/j.ins.2015.09.054.

Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are allset: A multiset function framework for hypergraph neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=hpBTIv2uy_E.

Uthsav Chitra and Benjamin Raphael. Random walks on hypergraphs with edge-dependent vertex weights. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1172–1181. PMLR, 2019. URL https://proceedings.mlr.press/v97/chitra19a.html.

Corinna Coupette, Sebastian Dalleiger, and Bastian Rieck. Ollivier-ricci curvature for hypergraphs: A unified framework. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=sPCKNl5qDps.

Karel Devriendt and Renaud Lambiotte. Discrete curvature on graphs from the effective resistance. *Journal of Physics: Complexity*, 3(2):025008, 2022. doi: 10.1088/2632-072X/ac730d. arXiv:2201.06385.

Iulia Duta, Giulia Cassarà, Fabrizio Silvestri, and Pietro Liò. Sheaf hypergraph networks. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0529.

Marzieh Eidi and Jürgen Jost. Ollivier ricci curvature of directed hypergraphs. *Scientific Reports*, 10:12466, 2020. doi: 10.1038/s41598-020-68619-6.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3558–3565, 2019. doi: 10.1609/aaai.v33i01.33013558.

Yifan Feng, Jiashu Han, Shihui Ying, and Yue Gao. Hypergraph isomorphism computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3880–3896, 2024. doi: 10.1109/TPAMI.2024.3353199.

Lukas Fesser and Melanie Weber. Effective structural encodings via local curvature profiles. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GIUjLsDP4Z.

Lukas Fesser, Sergio Serrano de Haro Iváñez, Karel Devriendt, Melanie Weber, and Renaud Lambiotte. Augmentations of Forman's Ricci curvature and their applications in community detection. *Journal of Physics: Complexity*, 5(3):035010, 2024. doi: 10.1088/2632-072X/ad64a3. arXiv:2306.06474.

Robin Forman. Bochner's method for cell complexes and combinatorial Ricci curvature. *Discrete & Computational Geometry*, 29(3):323–374, 2003. doi: 10.1007/s00454-002-0743-x.

Linton C. Freeman. Cliques, galois lattices, and the structure of human social groups. *Social Networks*, 18(3):173–187, 1996. doi: 10.1016/0378-8733(95)00271-5.

Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin, Heidelberg, 1999. doi: 10.1007/978-3-642-59830-2.

Olympio Hacquard. Hypergraph clustering using Ricci curvature: an edge transport perspective. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=HMROU8MXqV. arXiv:2412.15695.

Jing Huang and Jie Yang. Unignn: A unified framework for graph and hypergraph neural networks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 2563–2569, 2021. doi: 10.24963/ijcai.2021/353.

Sergei O. Kuznetsov. On stability of a formal concept. *Annals of Mathematics and Artificial Intelligence*, 49(1–4):101–115, 2007. doi: 10.1007/s10472-007-9053-6.

Wilmer Leal, Guillermo Restrepo, Peter F. Stadler, and Jürgen Jost. Forman–ricci curvature for hypergraphs. *Advances in Complex Systems*, 24(1):2150003, 2021. doi: 10.1142/S021952592150003X.

Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Machine Learning*, 52 (3):239–281, 2003. doi: 10.1023/A:1024068626366.

Yann Ollivier. Ricci curvature of Markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009. doi: 10.1016/j.jfa.2008.11.001.

Raphaël Pellegrin, Lukas Fesser, and Melanie Weber. Higher-order learning with graph neural networks via hypergraph encodings. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5782.

Michael G. Rawson, Audun Myers, Robert Green, Michael Robinson, and Cliff A. Joslyn. Formal concept lattice representations and algorithms for hypergraphs. *arXiv preprint arXiv:2307.11681*, 2023. doi: 10.48550/arXiv.2307.11681.

Camille Roth and Paul Bourgine. Lattices for dynamic, hierarchic & overlapping categorization: The case of epistemic communities. *Scientometrics*, 69(2):429–447, 2006. doi: 10.1007/s11192-006-0161-6.

Mingwen Shao, Zhiyong Hu, Weizhi Wu, and Huan Liu. Graph neural networks induced by concept lattices for classification. *International Journal of Approximate Reasoning*, 154:262–276, 2023. doi: 10.1016/j.ijar.2023.01.001.

Yulu Tian, Jicheng Ma, Yunyan Yang, and Liang Zhao. Community detection of undirected hypergraphs by Ricci flow. *Physical Review E*, 112(4):044311, 2025. doi: 10.1103/d5n8-y58m.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=7UmjRGzp-A.

Peihao Wang, Shenghao Yang, Yunyu Liu, Zhangyang Wang, and Pan Li. Equivariant hypergraph diffusion neural operators. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=RiTjKoscnNd.

Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché Buc, Edward A. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, volume 32, pp. 1509–1520. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/hash/1efa39bcaec6f3900149160693694536-Abstract.html. arXiv:1809.02589.

Shiyi Yang, Can Chen, and Didong Li. Lower Ricci curvature for hypergraphs, 2025. URL https://arxiv.org/abs/2506.03943.

Detian Zhang, Chengqiang Zhang, Yanghui Rao, Li Qing, and Chunjiang Zhu. Improved expressivity of hypergraph neural networks through high-dimensional generalized weisfeiler-leman algorithms. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 76880–76908. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhang25dc.html.

Mengyao Zhou, Zhiheng Zhou, Xiao Han, Xingqin Qi, Guanghui Wang, and Guiying Yan. Tackling over-smoothing on hypergraphs: A ricci flow-guided neural diffusion approach. *arXiv preprint arXiv:2603.15696*, 2026. doi: 10.48550/arXiv.2603.15696.

A Theoretical Properties of Galois Closure Curvature

Throughout, unless stated otherwise, let B and C be distinct Galois-closed subsets of V with $\mathcal{S}(B,C) \neq \emptyset$. We call such a pair *admissible*. These conditions ensure $d_{\triangle}(B,C) > 0$ and that the expectation in equation 6 is well-defined; the remaining cases are handled in Appendix A.7.

A.1 Basic Properties

Proposition 3 (Closure consistency)**.** *For any $B \subseteq V$ and $m \in V$, $\phi(T_m(B)) = T_m(B)$.*

Proof. Since every closure operator is idempotent, $\phi(T_m(B)) = \phi(\phi(B \triangle \{m\})) = \phi(B \triangle \{m\}) = T_m(B)$. □

Hence the response in equation 6 is always measured between valid closure states: the perturbation acts through the implication structure rather than comparing arbitrary toggled node sets.

Proposition 4 (Symmetry)**.** *For any admissible pair, $\kappa_{\mathrm{G}}(B,C) = \kappa_{\mathrm{G}}(C,B)$.*

Proof. $\mathcal{S}(B,C) = \mathcal{S}(C,B)$ by equation 5, and $d_{\triangle}$ is symmetric. Hence $\rho_m(B,C) = \rho_m(C,B)$ for every $m \in \mathcal{S}(B,C)$, and averaging over the same set gives the result. □

A.2 Proof of Proposition 1

Symmetry is Proposition 4. We prove the remaining claims.

Upper bound and maximal contraction. For every $m \in \mathcal{S}(B,C)$, $d_{\triangle}(T_m(B), T_m(C)) \geq 0$ and $d_{\triangle}(B,C) > 0$, so $\rho_m(B,C) \leq 1$ and hence $\kappa_{\mathrm{G}}(B,C) \leq 1$. Equality of the mean with 1 requires $\rho_m(B,C) = 1$ for every m, which is equivalent to $d_{\triangle}(T_m(B), T_m(C)) = 0$, i.e., $T_m(B) = T_m(C)$ for all $m \in \mathcal{S}(B,C)$.

Lower bound. Since $T_m(B), T_m(C) \subseteq V$, $d_{\triangle}(T_m(B), T_m(C)) \leq |V| = n$, so $\rho_m(B,C) \geq 1 - n/d_{\triangle}(B,C)$ for every feasible m. Taking the expectation gives the bound.

Sign. Rewriting equation 6 as

$$\kappa_{\mathrm{G}}(B,C) = 1 - \frac{\mathbb{E}_{m \in \mathcal{S}(B,C)}\left[d_{\triangle}(T_m(B), T_m(C))\right]}{d_{\triangle}(B,C)} \tag{10}$$

and using $d_{\triangle}(B,C) > 0$, we have $\kappa_{\mathrm{G}} > 0\,(= 0, < 0)$ iff the expected post-perturbation discrepancy is smaller than (equal to, larger than) the original discrepancy. □

Remarks. (i) The range is asymmetric: maximal contraction has the universal value 1, whereas the attainable magnitude of negative curvature grows with $n/d_{\triangle}(B,C)$. This asymmetry reflects that closure can propagate a single toggle into a much larger change of implication state. (ii) $\kappa_{\mathrm{G}} = 0$ does not imply that every perturbation preserves the discrepancy, since contraction and expansion may cancel in expectation. Proposition 2 shows that this cancellation occurs in practice, which motivates the curvature profile.

A.3 Properties of the Curvature Profile

Let (u,v) be a node pair with $C_u \neq C_v$ and $\mathcal{S}_{uv} \neq \emptyset$, and write $\rho_m = \rho_m(C_u, C_v)$ and $\mathbb{E}_m = \mathbb{E}_{m \in \mathcal{S}_{uv}}$.

Proposition 5 (Signed decomposition and second moment)**.**

$$\kappa_{\mathrm{G}}(u,v) = \kappa^{+}_{uv} - \kappa^{-}_{uv}, \qquad \mathbb{E}_m[\rho_m^2] = \xi_{uv}^{\,2} + \kappa_{\mathrm{G}}(u,v)^2. \tag{11}$$

Proof. For any $x \in \mathbb{R}$, $x = [x]_+ - [-x]_+$. Applying this to each ρ_m and taking expectations gives $\mathbb{E}_m[\rho_m] = \mathbb{E}_m[[\rho_m]_+] - \mathbb{E}_m[[-\rho_m]_+]$, which is the first identity. For the second, since $\kappa_{\mathrm{G}}(u,v) = \mathbb{E}_m[\rho_m]$, the variance identity gives $\xi_{uv}^{\,2} = \mathbb{E}_m[(\rho_m - \kappa_{\mathrm{G}})^2] = \mathbb{E}_m[\rho_m^2] - \kappa_{\mathrm{G}}^2$. □

Corollary 6 (Profile recovers the first two moments)**.** *The profile* $\mathbf{g}_{uv} = [\kappa^+_{uv}, \kappa^-_{uv}, \xi_{uv}]$ *determines both the scalar curvature and the second moment of the response:*

$$\kappa_{\mathrm{G}}(u,v) = \kappa^+_{uv} - \kappa^-_{uv}, \qquad \mathbb{E}_m[\rho_m^2] = \xi_{uv}^2 + \left(\kappa^+_{uv} - \kappa^-_{uv}\right)^2. \tag{12}$$

Proof. Substitute the first identity of equation 11 into the second. □

Conversely, κ_{G} depends only on the first moment of the response distribution and therefore cannot recover $\kappa^{\pm}_{uv}$ or ξ_{uv}. Proposition 2 gives a concrete hypergraph pair realizing this non-uniqueness: identical scalar curvature (0) with distinct profiles.

A.4 Proof of Proposition 2

In both hypergraphs every node has degree 2 and every pair co-occurs exactly once, so $\mathbf{B}_1^\top \mathbf{B}_1 = \mathbf{B}_2^\top \mathbf{B}_2$, and claim (1) follows immediately. The closed sets are

$$\mathcal{H}_1 :\ \{\emptyset, \{a\}, \{b\}, \{c\}, V\}, \qquad \mathcal{H}_2 :\ \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{b,c\}, \{a,c\}, V\}. \tag{13}$$

In particular, $C_v = \{v\}$ for all v in both hypergraphs and $\phi(\emptyset) = \emptyset$. By symmetry of both hypergraphs under permutations of $\{a,b,c\}$, it suffices to consider $(u,v) = (a,b)$. Here $d_\triangle(C_a, C_b) = 2$, and every toggle yields a set contained in some hyperedge, so $\mathcal{S} = \{a,b,c\}$ in both cases.

	$\mathcal{H}_1$			$\mathcal{H}_2$		
m	$T_m(C_a)$	$T_m(C_b)$	ρ_m	$T_m(C_a)$	$T_m(C_b)$	ρ_m
a	$\emptyset$	V	$-\frac{1}{2}$	$\emptyset$	$\{a,b\}$	0
b	V	$\emptyset$	$-\frac{1}{2}$	$\{a,b\}$	$\emptyset$	0
c	V	V	1	$\{a,c\}$	$\{b,c\}$	0

Hence $\kappa_{\mathrm{G}}(a,b) = 0$ on both hypergraphs (claim 2). On $\mathcal{H}_1$, $\kappa^+ = \frac{1}{3}$, $\kappa^- = \frac{1}{3}(\frac{1}{2} + \frac{1}{2}) = \frac{1}{3}$, and $\xi = \sqrt{(\frac{1}{4} + \frac{1}{4} + 1)/3} = \frac{1}{\sqrt{2}}$. On $\mathcal{H}_2$ all components vanish (claim 3). □

The contrast arises from the implication $a \wedge b \Rightarrow c$ in $\mathcal{H}_1$: toggling c drives both states to V (contraction), whereas toggling a or b drives one state to $\emptyset$ and the other to V (expansion).

A.5 GCC Is Flat on Pairwise Structure

Proposition 7 (Flatness on graphs)**.** *Let* $\mathcal{H}$ *be a simple graph (all* $|e| = 2$*, no repeated edges) with minimum degree at least* 2*. Then for every node pair* $u \neq v$ *and every* $m \in \mathcal{S}(C_u, C_v)$*,* $\rho_m(C_u, C_v) = 0$*. Hence* $\mathbf{g}_{uv} = \mathbf{0}$ *whenever* $\mathcal{S}(C_u, C_v) \neq \emptyset$*.*

Proof. Any two distinct edges incident to v intersect in $\{v\}$, so $C_v = \{v\}$ and $d_\triangle(C_u, C_v) = 2$. If all edges shared a common node, the graph would be a star with degree-one leaves; hence $\phi(\emptyset) = \bigcap_{e \in E} e = \emptyset$. Let $m \in \mathcal{S}(C_u, C_v)$. (i) If $m = u$, then $T_m(C_u) = \phi(\emptyset) = \emptyset$. Feasibility of $\{u,v\}$ requires $uv \in E$, so $T_m(C_v) = \phi(\{u,v\}) = \{u,v\}$, and $d_\triangle = 2$. The case $m = v$ is symmetric. (ii) If $m \notin \{u,v\}$, feasibility requires $um, vm \in E$, so $T_m(C_u) = \{u,m\}$, $T_m(C_v) = \{v,m\}$, and $d_\triangle = 2$. In all cases, $\rho_m = 1 - 2/2 = 0$. □

The degree condition is needed: on the path a–b–c–d, $C_a = \{a,b\}$ and toggling b sends C_a to $\{a,b\}$ and C_b to $\phi(\emptyset) = \emptyset$, so $g_{ab} = [0, \frac{1}{2}, \frac{1}{2}]$.

A.6 Relation to Concept Stability

Concept stability (Kuznetsov, 2007) measures the fraction of sub-contexts in which a formal concept is preserved, and thus scores the robustness of a single closure to the removal of objects. GCC shares the premise that the response of closure to perturbation carries structural information, but differs in three respects: (i) it compares *pairs* of closure states under a *shared* perturbation rather than scoring a single concept; (ii) its response is *signed*, separating contraction from expansion; and (iii) it is

normalized by the original discrepancy, yielding an Ollivier-type contraction coefficient rather than a robustness score. Moreover, GCC perturbs attributes (nodes) through T_m, whereas stability perturbs objects (hyperedges).

A.7 Edge Cases and Complexity

Edge cases. If $C_u = C_v$, then $d_{\triangle}(C_u, C_v) = 0$ and ρ_m is undefined; such nodes occur in exactly the same hyperedges, and we assign $\mathbf{g}_{uv} = \mathbf{0}$. The same convention is used when $\mathcal{S}(C_u, C_v) = \emptyset$. Nodes of degree zero, for which $C_v = V$, are excluded, and $\mathbf{c}_{v,e} = \mathbf{0}$ for singleton hyperedges.

Complexity. A toggle $m \notin B$ is feasible only if $B \cup \{m\}$ lies in some hyperedge, so $|\mathcal{S}(C_u, C_v)|$ is bounded by the number of nodes co-occurring with C_u, which is at most $d_{\max} s_{\max}$, where $d_{\max}$ and $s_{\max}$ are the maximum degree and hyperedge size. Evaluating $\phi(X)$ requires intersecting the hyperedges in X', at cost $O(d_{\max} s_{\max})$. Since profiles are needed only for co-occurring pairs (equation 8), the total cost is

$$O\Big(\textstyle\sum_{e \in E} |e|^2 \cdot d_{\max}^2 \, s_{\max}^2 \Big), \tag{14}$$

and all profiles are precomputed once before training.

B Hy-WL Collections and Evaluation Protocol

Collections. The collections are those of Feng et al. (2024); each item is one hypergraph with one class label (Table 2). The four RHG collections are synthetic hypergraphs, each generated from one of several structural templates (for example check tables, pyramids and wheels), which is its class. The six real ones are the co-director and co-writer hypergraphs of movies, labelled by form or genre (IMDB); the games of a Steam player, linked when they share a tag and labelled by a preference for single- or multi-player games; and the friends of a Twitter user, labelled by which of two topics the user posted about. We keep every hypergraph and hyperedge as released. Some hypergraphs occur more than once with different labels, and no descriptor can tell such copies apart; this concerns about a quarter of the two IMDB writer collections (last column of Table 2).

Collection	Type	#HG	#Cls	Maj. (%)	$\overline{\lvert V \rvert}$	$\overline{\lvert E \rvert}$	$\overline{\lvert e \rvert}$	$\max \lvert e \rvert$	Ident. (%)
RHG-Table	synthetic	1,000	2	50.4	36.3	63.3	2.60	4	0.0
RHG-Pyramid	synthetic	1,000	2	50.7	28.8	30.6	3.00	3	8.4
RHG-3	synthetic	1,500	3	33.5	35.5	17.9	4.56	24	0.0
RHG-10	synthetic	2,000	10	10.8	31.3	29.8	3.16	22	1.8
IMDB-Dir-Form	real	1,869	3	42.7	15.7	39.3	3.37	110	0.9
IMDB-Dir-Genre	real	3,393	3	51.3	17.3	36.6	3.48	110	1.8
IMDB-Wri-Form	real	374	4	45.7	10.1	3.8	4.48	179	24.6
IMDB-Wri-Genre	real	1,172	6	26.2	12.8	4.5	4.94	266	28.1
Steam-Player	real	2,048	2	50.0	13.8	46.4	4.54	71	0.0
Twitter-Friend	real	1,310	2	59.2	21.6	84.8	4.85	64	0.0

Table 2: The ten Hy-WL collections. #HG: hypergraphs; #Cls: classes; Maj.: majority-class rate; $\overline{|V|}$, $\overline{|E|}$: mean numbers of nodes and hyperedges per hypergraph; $\overline{|e|}$, $\max|e|$: mean and largest hyperedge size; Ident.: hypergraphs with an identical copy under another label.

Protocol. Each component k of c_e, and each baseline value, gives a distribution over the hyperedges of a hypergraph; W_1 in Eq. (9) is the 1-Wasserstein distance between such distributions on the real line. The only single-node hyperedges of Hy-WL, 272 in RHG-10, contain no node pair; they are left out for GCC and every other descriptor defined on node pairs, and kept for HFRC, ORCHID and the two controls. The SVM is scikit-learn's `SVC` with a precomputed kernel and default settings ($C = 1$), on `KFold`(5, shuffled) with seeds 2021 to 2025 as in the code of Feng et al. (2024); all descriptors use the same folds. In every fold, s_k is the standard deviation of component k over the hyperedges of the training hypergraphs, and γ the median of d over distinct pairs of training hypergraphs; held-out hypergraphs use the same s_k and γ. s_k was positive in every fold. The median was zero only for the size control, in every fold of RHG-Pyramid, whose hyperedges

all have three nodes, and in 2 of the 25 folds of RHG-Table; the kernel is then constant and the SVM predicts a single class. Over the five seeds, the standard deviation of the accuracy of GCC is at most 1.2 points on every collection. A corrected resampled t-test (Nadeau & Bengio, 2003) of GCC against the most accurate existing curvature of each collection, with the false discovery rate controlled over the ten collections, finds GCC more accurate on RHG-10 (by 4.57 points) and less accurate on IMDB-Wri-Genre (by 11.25 points), and no other significant difference.

C TWIN COLLECTIONS

Motif pairs. We enumerate hypergraphs on six nodes with at most five distinct hyperedges of size at least two that cover every node, and group them by $B^\top B$, which fixes every node degree and every pairwise co-occurrence count. Two non-isomorphic hypergraphs form a pair if they have the same $B^\top B$ and every node has the same number of incident hyperedges of each size on both sides; pairs equal up to a simultaneous relabelling of the nodes are of the same type. This gives 1,800 pairs of five types, all with five hyperedges per side. We use one pair (M_0, M_1) of each type (Figure 5, Table 3); its two sides differ in which implications $u \wedge v \Rightarrow w$ hold.

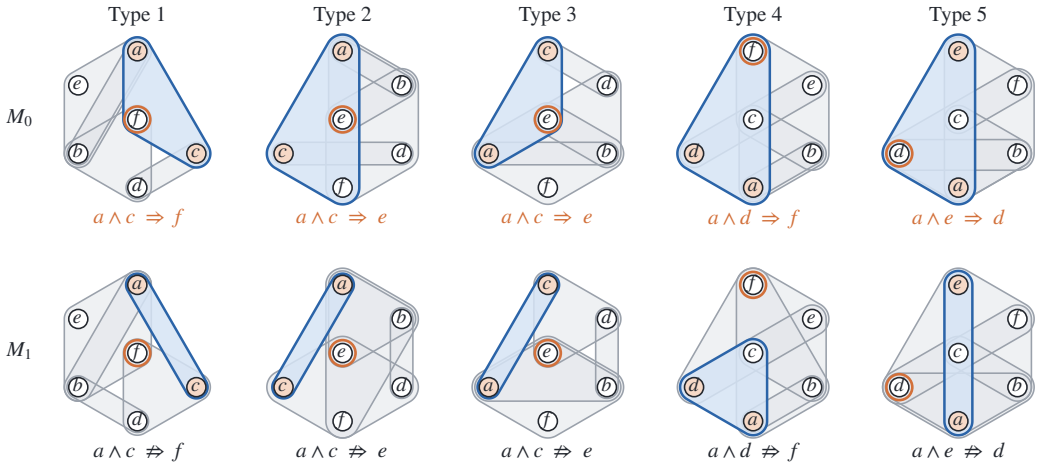


Figure 5: The five motif pairs, M_0 (top) and M_1 (bottom). In each pair, the filled nodes u, v and the ringed node w mark one implication $u \wedge v \Rightarrow w$ that holds in M_0 and not in M_1; hyperedges that contain u and v are blue.

Type	Hyperedges of M_0	Hyperedges of M_1	Example (M_0 only)	Only in M_0 / M_1
1	ab, cd, abe, acf, bdf	ac, bd, abe, abf, cdf	$a \wedge c \Rightarrow f$	6 / 5
2	$ab, cd, abe, acef, bdef$	$ac, bd, abe, abef, cdef$	$a \wedge c \Rightarrow e$	10 / 11
3	$ab, cd, ace, bde, abef$	$ac, bd, abe, cde, abef$	$a \wedge c \Rightarrow e$	4 / 6
4	$ab, abc, cde, acdf, bcef$	$ab, acd, bce, abcf, cdef$	$a \wedge d \Rightarrow f$	6 / 5
5	$abc, abd, cef, acde, bcdf$	$abd, ace, bcf, abcd, cdef$	$a \wedge e \Rightarrow d$	4 / 6

Table 3: The motif pairs on the nodes a to f. Last column: number of implications $u \wedge v \Rightarrow w$ (u, v co-occurring, $w \notin \{u, v\}$) that hold on one side of the pair only.

Construction. For each motif type and $K \in \{1, 3, 5\}$ copies, a random background has $6K + 12$ nodes and 24 hyperedges, each of a size drawn uniformly from $\{2, 3, 4\}$ with nodes drawn uniformly without replacement. A random permutation of the nodes defines K disjoint slots of six nodes; the hypergraph of class 0 is the background with a copy of M_0 on every slot, and its twin, of class 1, is the same background with M_1 on the same slots, so each has $24{+}5K$ hyperedges. A draw is repeated if either twin contains a repeated hyperedge or a node in no hyperedge. Each of the 15 collections holds 500 twin pairs; every pair has equal $B^\top B$, equal hyperedge-size counts at every node and different hyperedge sets. The SVM protocol is that of Appendix B, with each twin pair kept in one

fold. Clustering applies kernel PCA to two dimensions and k-means ($k = 2$, 20 initializations) to all 1,000 hypergraphs, with s_k and γ computed on all of them, which uses no labels.

Results for each motif type. Figure 3(a, b) shows the mean over the five motif types; Table 4 gives every descriptor on each type, with the accuracy for three and five copies and the ARI for five copies. With labels, GCC is the most accurate curvature on every motif type with three and five copies, and the profile is more accurate than the scalar in all 15 combinations of motif type and K, including one copy. Among the existing curvatures, only ETR and HRCD exceed 70% on any motif type; the other eight stay below 66%. Without labels, GCC has the highest ARI on types 2, 3 and 5, HRCD on type 1 and GCC-κ on type 4.

	Accuracy (%), $K = 3$						Accuracy (%), $K = 5$						ARI, $K = 5$					
Motif type	1	2	3	4	5	Mean	1	2	3	4	5	Mean	1	2	3	4	5	Mean
Size	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	.00	.00	.00	.00	.00	.00
Degree	52.5	52.6	50.8	50.3	50.4	51.3	54.5	55.4	54.5	51.6	53.0	53.8	.00	.00	.00	.00	.00	.00
HFRC	54.2	53.6	53.3	50.5	53.1	52.9	55.4	58.1	57.2	53.6	55.6	56.0	.00	.00	.00	.00	.00	.00
CIHI	55.1	48.5	58.4	51.3	50.0	52.7	65.6	50.1	63.5	50.6	49.8	55.9	.06	.00	.00	.00	.00	.01
ORCHID	52.8	50.7	52.8	50.1	51.2	51.5	63.5	50.7	57.8	48.9	50.7	54.3	.02	.00	.01	.00	.00	.00
CE-FRC	49.8	51.6	50.8	49.8	50.1	50.4	54.4	49.8	54.3	49.6	50.6	51.8	.00	.00	.00	.00	.00	.00
CE-ORC	52.7	50.6	52.9	50.1	51.1	51.5	63.2	50.7	57.8	48.9	50.9	54.3	.02	.00	.01	.00	.00	.00
CE-AFRC$_3$	54.2	49.6	54.3	50.8	51.1	52.0	59.6	50.1	59.0	50.3	48.8	53.6	.01	.00	.01	.00	.00	.00
ETR	67.4	73.4	64.7	62.4	58.2	65.2	80.4	84.2	75.4	68.8	63.3	74.4	.01	.09	.11	.00	.02	.05
HRCD	<u>72.5</u>	71.3	64.4	60.5	52.5	64.2	83.3	80.4	72.6	65.3	55.9	71.5	**.17**	.15	.06	.02	.00	.08
BE	57.1	56.7	51.9	52.9	53.0	54.3	62.7	62.0	57.5	53.0	56.0	58.2	.00	.00	.00	.00	.00	.00
Res.	51.3	50.5	50.8	50.1	49.0	50.3	51.2	50.8	52.9	48.0	52.4	51.0	.00	.00	.00	.00	.00	.00
GCC-κ	61.5	<u>78.7</u>	<u>69.2</u>	<u>71.6</u>	<u>70.0</u>	<u>70.2</u>	<u>84.9</u>	<u>98.1</u>	<u>93.7</u>	<u>93.0</u>	<u>91.9</u>	<u>92.3</u>	.00	<u>.32</u>	<u>.21</u>	**.29**	<u>.33</u>	<u>.23</u>
GCC	**77.5**	**91.0**	**91.5**	**87.5**	**95.5**	**88.6**	**97.4**	**99.7**	**99.7**	**99.7**	**99.8**	**99.3**	<u>.10</u>	**.49**	**.50**	<u>.03</u>	**.74**	**.37**

Table 4: Twin collections by motif type; Mean is the value in Figure 3(a, b). Bold/underline: best/second-best curvature; Size and Degree are unranked controls. Chance is 50% accuracy and 0 ARI.

Embeddings of all motif types. Figure 6 shows the embeddings of the top row of Figure 4 for all five motif types with five copies. GCC places the two classes on the two arms of its embedding for types 2, 3 and 5. On type 4 its classes differ by 2.8 within-class standard deviations along the second kernel-PCA direction and by 0.5 along the first, yet k-means on the two coordinates recovers little of this (ARI 0.03), while GCC-κ reaches 0.29. On type 1 the classes separate only partly for every curvature.

D Computational cost

Each curvature is timed on one CPU thread (AMD EPYC 7352) in a fresh process, from the loaded hypergraph to its values, including every graph it builds (clique expansion, line graph, shortest paths). Runs shorter than 10 s follow a warm-up run and report the median of three. ORCHID and CE-ORC include about 8.5 s of Julia compilation, which dominates their times on the three smaller hypergraphs. The released code of HRCD does not finish Cora-CC within 2 h, that of Res. not Cora-CA, and that of ETR and CE-AFRC$_3$ not Pubmed-CC; for these four we time faster implementations that agree with the released code where it finishes, so their times are lower than the released code would give. Table 5 gives all times.

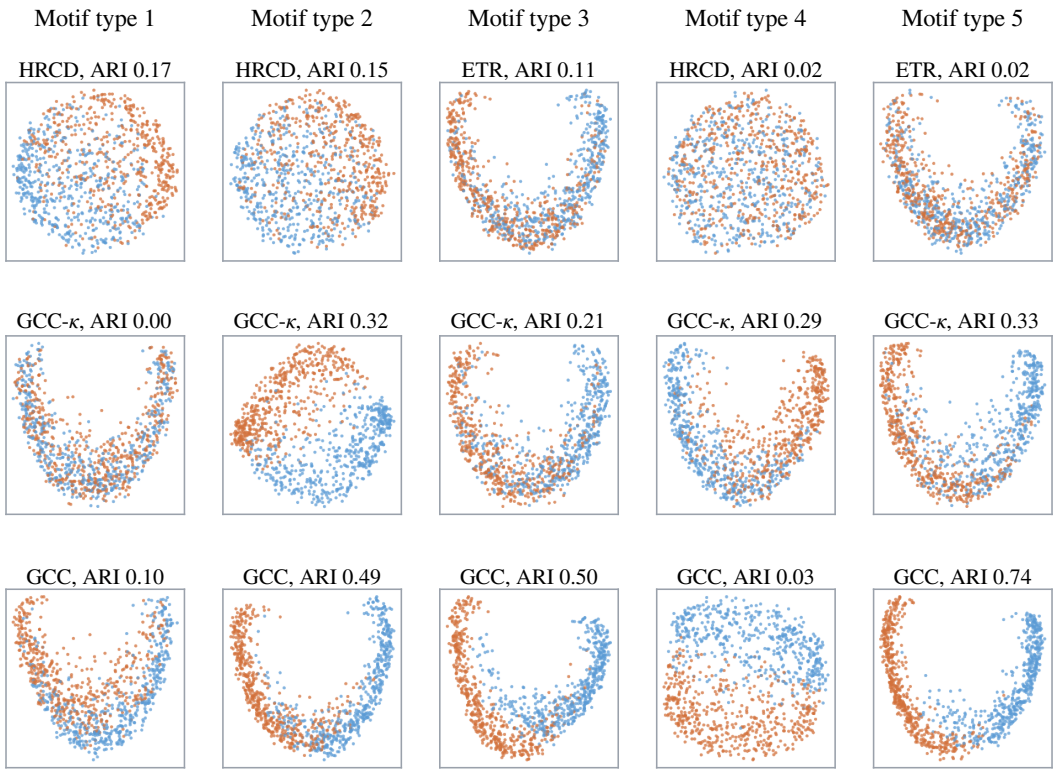


Figure 6: Kernel-PCA embeddings without labels for the five motif types with five copies (columns): the existing curvature with the highest k-means ARI on that type (top), GCC-κ (middle) and GCC (bottom); colour marks the class.

Curvature	Cora-CC	Citeseer-CC	Cora-CA	Pubmed-CC
Incidences	4,786	3,453	4,585	34,629
Co-occurring node pairs	4,256	4,307	14,942	123,827
HFRC	0.9 ms	0.9 ms	0.9 ms	2.0 ms
CIHI	0.01	9.5 ms	0.01	0.21
ORCHID	13	12	35	822
CE-FRC	8.6 ms	6.6 ms	9.4 ms	0.07
CE-ORC	12	12	34	829
CE-AFRC$_3$	0.01	0.01	0.02	0.50
ETR	3.2	2.1	4.7	107
HRCD	4.9	5.1	15	1,273
BE	2.8	0.93	0.73	155
Res.	1.3	1.4	2.1	46
GCC	0.05	0.05	0.09	2.2

Table 5: Time in seconds (ms: milliseconds) on one CPU thread for the four hypergraphs of Yadati et al. (2019).

E GCC FOR MESSAGE PASSING

As shown in Table 6, GCC-based encodings remain competitive with existing curvature encodings across five message-passing backbones. GCC-p further improves over GCC-κ in several settings, indicating that the profile can provide complementary structural information beyond the scalar curvature.

Dataset	Method	UniGCN	UniGAT	UniGIN	UniSAGE	UniGCNII
Citeseer-CC	No encoding	61.71 ±1.95	61.59 ±1.87	58.85 ±2.69	58.63 ±2.70	**62.32** ±1.92
	HFRC	61.48 ±2.00	61.49 ±1.87	58.66 ±2.67	58.87 ±2.49	61.97 ±2.02
	ORCHID	61.52 ±2.11	61.67 ±1.88	58.79 ±2.74	58.78 ±2.57	62.02 ±1.76
	CE-FRC	61.59 ±2.00	61.62 ±1.82	58.75 ±2.72	58.74 ±2.56	62.21 ±1.87
	CE-ORC	61.55 ±1.98	61.56 ±1.76	58.65 ±2.60	58.84 ±2.50	62.05 ±2.22
	CE-AFRC$_3$	61.53 ±1.98	61.67 ±1.80	58.80 ±2.51	58.65 ±2.63	62.23 ±2.01
	Edge-transport Ricci	61.69 ±2.00	61.60 ±1.90	58.71 ±2.61	58.79 ±2.63	62.01 ±2.01
	HyperRCD	61.66 ±1.85	61.69 ±1.81	58.89 ±2.68	**58.96** ±2.48	62.12 ±1.95
	Bakry–Émery	61.38 ±2.16	61.65 ±1.93	58.77 ±2.49	58.79 ±2.40	62.15 ±1.88
	GCC-κ (ours)	61.54 ±2.06	61.45 ±1.91	58.83 ±2.51	58.74 ±2.58	62.05 ±1.81
	GCC-p (ours)	**61.86** ±1.94	**61.75** ±1.86	**58.98** ±2.31	58.90 ±2.33	62.14 ±2.10
	Δ (p$-\kappa$)	+0.31	+0.30	+0.15	+0.16	+0.08
Cora-CA	No encoding	73.47 ±1.95	73.48 ±1.47	71.08 ±2.14	71.09 ±2.12	74.38 ±1.91
	HFRC	73.39 ±2.01	73.50 ±1.48	71.29 ±2.17	70.78 ±2.05	74.42 ±1.68
	ORCHID	73.51 ±1.84	73.39 ±1.69	71.26 ±1.91	71.25 ±1.94	74.47 ±1.60
	CE-FRC	73.58 ±2.12	**73.57** ±1.60	71.42 ±2.38	71.31 ±2.21	74.38 ±1.70
	CE-ORC	73.44 ±1.79	73.52 ±1.49	70.92 ±2.14	71.03 ±2.22	74.39 ±1.78
	CE-AFRC$_3$	73.62 ±1.94	73.52 ±1.40	70.95 ±1.84	71.32 ±1.82	74.43 ±1.91
	Edge-transport Ricci	73.69 ±1.78	**73.57** ±1.54	71.23 ±2.11	71.30 ±2.09	**74.64** ±1.78
	HyperRCD	**73.71** ±1.95	73.29 ±1.66	71.07 ±2.18	71.33 ±1.97	74.41 ±1.87
	Bakry–Émery	73.34 ±1.96	73.37 ±1.54	71.26 ±1.78	71.03 ±2.20	74.37 ±1.73
	GCC-κ (ours)	73.25 ±1.79	73.36 ±1.63	71.18 ±1.99	71.09 ±2.01	74.49 ±1.87
	GCC-p (ours)	73.50 ±1.95	73.49 ±1.69	**71.45** ±1.74	**71.47** ±1.98	74.38 ±1.64
	Δ (p$-\kappa$)	+0.25	+0.14	+0.27	+0.38	−0.10
Cora-CC	No encoding	68.84 ±1.64	68.43 ±2.05	63.96 ±2.86	63.81 ±3.06	**68.17** ±1.79
	HFRC	68.76 ±1.66	68.38 ±2.12	64.20 ±2.77	64.19 ±2.75	67.92 ±1.75
	ORCHID	**68.85** ±1.57	68.34 ±2.08	64.19 ±3.07	64.22 ±2.80	68.03 ±1.65
	CE-FRC	68.77 ±1.62	68.07 ±2.18	64.63 ±2.64	64.77 ±2.52	68.05 ±1.77
	CE-ORC	68.67 ±1.61	68.27 ±2.16	64.55 ±2.58	64.61 ±2.58	67.97 ±1.70
	CE-AFRC$_3$	68.76 ±1.72	68.11 ±2.29	**64.78** ±2.84	**64.87** ±2.82	68.04 ±1.83
	Edge-transport Ricci	68.80 ±1.70	**68.47** ±2.11	64.32 ±2.62	64.49 ±2.52	68.10 ±1.76
	HyperRCD	68.80 ±1.66	68.29 ±2.10	64.16 ±3.05	64.25 ±2.92	68.04 ±1.55
	Bakry–Émery	68.65 ±1.55	68.34 ±1.93	64.51 ±2.87	64.43 ±2.68	68.00 ±1.75
	GCC-κ (ours)	68.66 ±1.54	68.24 ±1.99	64.24 ±3.03	64.54 ±2.81	67.80 ±1.67
	GCC-p (ours)	68.48 ±1.82	68.19 ±2.04	64.33 ±2.60	64.20 ±2.91	67.76 ±1.55
	Δ (p$-\kappa$)	−0.18	−0.06	+0.09	−0.34	−0.04
Pubmed-CC	No encoding	74.61 ±1.12	75.02 ±1.24	74.29 ±1.41	74.24 ±1.53	74.55 ±1.22
	HFRC	74.64 ±1.12	**75.09** ±1.20	74.28 ±1.43	74.30 ±1.34	74.57 ±1.23
	ORCHID	74.60 ±1.18	75.04 ±1.31	74.27 ±1.53	74.24 ±1.53	74.59 ±1.28
	CE-FRC	74.58 ±1.17	75.08 ±1.25	74.37 ±1.50	74.34 ±1.48	74.53 ±1.34
	CE-ORC	**74.65** ±1.21	75.04 ±1.30	**74.38** ±1.60	74.25 ±1.57	74.55 ±1.31
	CE-AFRC$_3$	74.63 ±1.13	75.07 ±1.26	74.19 ±1.49	**74.39** ±1.47	74.52 ±1.30
	Edge-transport Ricci	74.60 ±1.11	75.02 ±1.21	74.34 ±1.42	74.19 ±1.50	74.58 ±1.31
	HyperRCD	74.57 ±1.18	75.00 ±1.28	74.31 ±1.42	74.32 ±1.40	74.54 ±1.28
	Bakry–Émery	74.62 ±1.19	75.08 ±1.25	74.35 ±1.46	74.23 ±1.46	74.58 ±1.32
	GCC-κ (ours)	74.55 ±1.28	75.03 ±1.25	**74.38** ±1.49	74.14 ±1.56	74.49 ±1.31
	GCC-p (ours)	74.52 ±1.31	75.01 ±1.12	74.30 ±1.55	74.38 ±1.50	**74.63** ±1.26
	Δ (p$-\kappa$)	−0.04	−0.02	−0.08	+0.24	+0.14

Table 6: Full node classification results with five HNN backbones. The Δ row reports their difference in percentage points, computed from unrounded means and shown in red when the profile is higher. Best value per dataset–backbone pair in **bold**, second best underlined.