WHAT GRAPH NEURAL TANGENT KERNELS FORGET: PRESERVING ATTRIBUTE–STRUCTURE INTERACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph classification can depend on the relationship between node attributes and local graph topology. In MUTAG, we identify molecules with opposite mutagenicity labels and different attachment positions despite identical atom counts and degree histograms. We study the ability of graph neural tangent kernels (GNTKs) to distinguish such attribute–structure relations. For GNTKs with self-looped, symmetrically normalized propagation and mean readout, we construct graph families that remain indistinguishable at every depth, even when pooled outputs from all layers are combined. On a balanced binary classification task defined over these graphs, any classifier using these GNTK representations is limited to chance accuracy. We propose Fixed-Memory GNTK (FM-GNTK), which combines a multi-depth GNTK with a memory kernel built from products of original node attributes at pairs of nodes, weighted by finite-step graph propagation. At a fixed memory order, the retained input-based interaction statistics remain unchanged as network depth increases. This memory admits a perfect separator for the constructed task. We further prove that fusion with positive weights on both normalized branches preserves every graph distinction captured by either branch. Experiments on ten TU and three large molecular benchmarks evaluate predictive performance, component contributions, and Nyström approximation fidelity. Matched comparisons and independently tuned controls show that the predictive benefits of combining the branches depend on the dataset and comparison setting.

1 INTRODUCTION

Molecular mutagenicity prediction illustrates why the relationship between atom types and local connectivity matters. Kazius et al. (2005) refined prediction rules for aromatic nitro groups by considering other groups attached to the same ring and their relative positions. In a molecular graph, atom types are node attributes and connectivity defines the topology. The relevant information therefore includes how these attributes are arranged within local structures.

Figure 1 illustrates such arrangements in MUTAG (Morris et al., 2020). Graphs 117 and 119 have opposite observed mutagenicity labels, although both contain 12 carbon, 3 nitrogen, and 6 oxygen nodes, 22 edges, and the same degree histogram. Each graph contains two six-carbon cycles joined by a bridge. On the cycle with a single C–N$(O)_2$-shaped attachment, the attachment carbon is two edges from the bridge endpoint in graph 117 and one edge away in graph 119. These matched inputs give a concrete motivation for studying which relations between node attributes and local graph topology a representation preserves.

Graph neural tangent kernels (GNTKs) (Du et al., 2019) extend neural tangent kernels (Jacot et al., 2018) to infinite-width message-passing networks (Gilmer et al., 2017). Their representations depend on both node attributes and graph topology, but this dependence does not guarantee that every attribute–structure distinction is retained. Existing analyses of over-smoothing and deep GNTKs study how representations change with increasing depth and how residual connections can mitigate degradation (Li et al., 2018; Oono & Suzuki, 2020; Huang et al., 2022). We study exact distinguishability for GNTKs with self-looped, symmetrically normalized propagation (Kipf & Welling, 2017) and mean graph readout. The central issue is whether particular input relations can become indistinguishable at the graph level, even when the representations from every depth are available.

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

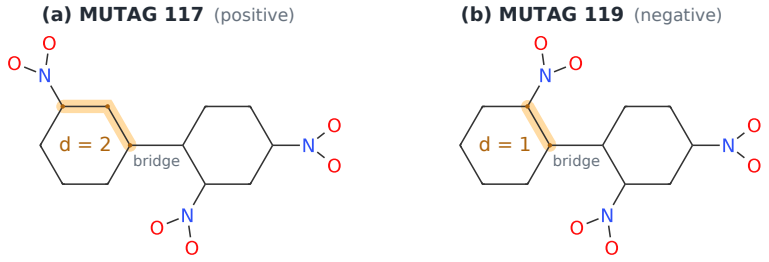

Figure 1: **Identical atom counts and degree histograms, different attachment positions and observed labels.** Unlabeled vertices represent carbon atoms. Both MUTAG graphs contain 12 carbon, 3 nitrogen, and 6 oxygen nodes, with 22 edges; the numbers of nodes with degrees one, two, and three are 6, 7, and 8, respectively. Orange paths connect the attachment carbon to the bridge endpoint on the same cycle. The corresponding graph distances are $d = 2$ for graph 117 and $d = 1$ for graph 119, excluding the bridge itself. Lines show untyped connectivity, not chemical bond orders.

We describe these relations through the input-based statistics

$$M_p(G) = \frac{1}{|V_G|} X_G^\top P_G^p X_G, \tag{1}$$

where X_G contains the original node attributes, P_G is a fixed propagation operator, and $|V_G|$ is the number of nodes. The matrix $P_G^p X_G$ contains attributes aggregated over p propagation steps. Multiplication by $X_G^\top$ pairs these propagated attributes with each node's original attributes. For one-hot atom types, the carbon–nitrogen entry sums the nitrogen-labelled signal received by carbon nodes after p weighted propagation steps and divides it by the total number of nodes. Thus M_p records which node types receive which attribute signals through the graph's connections.

We prove that the studied GNTK can map graphs with different attribute–topology interactions to identical representations. Specifically, we construct graph families of varying sizes in which paired graphs differ in M_p for some finite order p, yet have identical GNTK representations at every depth. We call this failure to represent interaction differences present in the input graphs *finite-order interaction loss*. Combining the already pooled layer outputs, an output-level variant of jumping knowledge (Xu et al., 2018), does not resolve these collisions. On a theoretical binary classification task constructed from these graph families, with labels determined by their interaction differences, the two graphs in each colliding pair have opposite labels and are sampled with equal probability. Classifiers restricted to their GNTK representations therefore have expected accuracy at most $1/2$, whereas a classifier using the corresponding M_p statistic can separate the two classes perfectly. Further constructions establish distinct roles for the propagation order and network depth: higher-order statistics can reveal additional differences, while nonlinear recursion can distinguish graphs whose statistics agree over a bounded range of propagation orders.

These results lead to **Fixed-Memory GNTK (FM-GNTK)**. A dynamic multi-depth branch combines recursively computed layer kernels, while a fixed memory branch retains the statistics in Eq. (1). Memory is computed directly from the original input and does not feed back into the dynamic recursion. We normalize the two branch kernels separately and combine them with nonnegative weights. For the constructed families, we prove that the identified separations survive this normalization and fusion when the separating branches receive positive weight. Depth and memory order remain separate controls for recursive feature computation and input-based structural information.

Our contributions are:

- **A specific representation failure and its task consequence.** We identify graph families that the studied GNTK cannot distinguish at any depth and prove the resulting limitation on a constructed classification task.
- **A targeted repair with separation guarantees.** We characterize the missing input statistics, retain them in a fixed memory kernel, and prove recovery after normalization and fusion, with distinct roles for memory and recursion under a bounded order.

- **An empirical evaluation of FM-GNTK.** Benchmarks on 13 datasets are supplemented by matched ablations, independently tuned controls on selected datasets, and exact–Nyström fidelity tests.

2 Background and Related Work

Structural graph kernels. Graph kernels encode graph structure through walks (Vishwanathan et al., 2010; Sugiyama & Borgwardt, 2015), shortest paths (SP) (Borgwardt & Kriegel, 2005), and Weisfeiler–Lehman refinement (WL) (Shervashidze et al., 2011). Subsequent methods enrich these descriptions or how they are compared: DGK learns representations of graph substructures (Yanardag & Vishwanathan, 2015); ASP augments graphs for shortest-path comparison (Ye et al., 2024); and the filtration-based WL kernel (FWL) incorporates graph filtrations (Schulz et al., 2022). WWL compares WL-derived representations through optimal transport, with TWWL accelerating its categorical formulation using a tree metric (Togninalli et al., 2019; Sando et al., 2025). GAWL uses WL-based graph alignment (Nikolentzos & Vazirgiannis, 2023), and MMD-GK compares distributions of node representations through maximum mean discrepancy (Sun & Fan, 2024). These approaches provide structural graph-kernel references for our TU experiments.

Graph neural networks and neural tangent kernels. GCN (Kipf & Welling, 2017) and GraphSAGE (Hamilton et al., 2017) aggregate neighborhood information, while GIN connects injective aggregation to WL distinguishability (Xu et al., 2019; Morris et al., 2019). DIFFPOOL (Ying et al., 2018) learns hierarchical graph pooling; an MLP provides a reference without message passing. These architectures form the neural comparison group in our large-dataset experiments. GNTK connects message-passing networks to kernel methods through the infinite-width limit (Du et al., 2019), and GraphQNTK (Tang & Yan, 2022) develops a quantum neural tangent kernel for graphs. Large-graph GNTK convergence (Krishnagopal & Ruiz, 2023) has also been studied. Our analysis concerns the information retained by a GNTK with self-looped, normalized aggregation and mean readout. Jumping Knowledge (JK) (Xu et al., 2018) combines node representations from different layers, while GCKN (Chen et al., 2020) analyzes complementary distinctions captured by shallow path and multilayer kernels, including their sum. From an optimization perspective, Huang et al. (2022) study large-depth GNTK behavior and residual remedies.

3 Finite-Order Interaction Loss

We study two aspects of graph discrimination: interaction loss under normalized aggregation and pooling, and the distinct information captured by moment order and nonlinear depth (Figure 2).

Setup. For an undirected attributed graph $G = (V, E, X)$ with $n = |V|$, define

$$P_G = \widehat{D}_G^{-1/2}(A_G + I)\widehat{D}_G^{-1/2}, \qquad M_p(G) = \frac{1}{n} X_G^\top P_G^p X_G, \tag{2}$$

where $\widehat{D}_G$ contains the degrees of $A_G + I$. The normalized GCN (Kipf & Welling, 2017) updates $H_{\ell+1} = \sigma(P_G H_\ell W_\ell + \mathbf{1}a_\ell^\top)$, starting from $H_0 = X_G$; the shared bias a_ℓ is omitted in our bias-free ReLU GNTK. Readout uses $b_G = n^{-1}\mathbf{1}$; output-level JK combines already pooled layer outputs. The moment M_p pairs original attributes with p-step propagated attributes. For a PSD kernel with positive diagonals,

$$\overline{K}(G, H) = \frac{K(G, H)}{\sqrt{K(G, G)K(H, H)}}, \qquad d_K^2(G, H) = 2 - 2\overline{K}(G, H). \tag{3}$$

Thus $d_K^2 = 0$ means that the normalized kernel representations coincide.

Definition 1 (Finite-order interaction loss)**.** For a graph pair (G, H) and order p, a graph-kernel family exhibits *finite-order interaction loss* if $M_p(G) \neq M_p(H)$ but either (i) *aggregation blindness*: $d_{K_L}(G, H) = 0$ for the considered depths, or (ii) *recursive attenuation*: $d_{K_L}(G, H) \to 0$ with depth while the fixed moment difference remains nonzero.

We focus on exact collisions; the linear attenuation result is discussed at the end of this section.

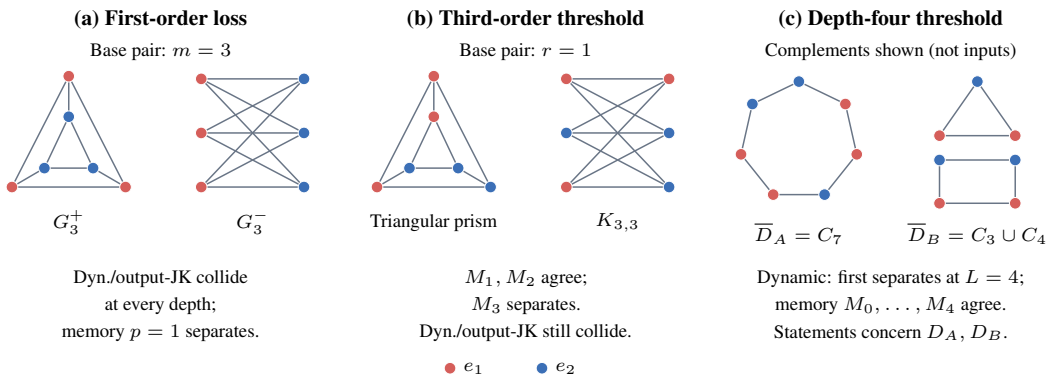

Figure 2: **Three graph pairs with distinct separation mechanisms.** Colors encode binary one-hot node attributes; self-loops are omitted. Panels (a) and (b) use the same two unlabelled graphs with different attributes. Panels (a) and (b) show the base cases $m = 3$ and $r = 1$, respectively. Panel (c) displays the *complements* of the input graphs. The depth-four separation in (c) concerns the ReLU GNTK.

3.1 Interaction loss under aggregation and pooling

For $m \geq 3$, split $2m$ vertices into equal binary-attribute groups V_0, V_1, with $x_v = e_1$ for $v \in V_0$ and $x_v = e_2$ for $v \in V_1$, where e_1, e_2 are the standard basis vectors of $\mathbb{R}^2$. Here m denotes the number of vertices in each attribute group, so each graph contains $2m$ vertices. In G_m^+, each group forms an m-cycle, with matching edges between groups. In G_m^-, vertex $i \in V_0$ connects to $i-1, i, i+1 \pmod m$ in V_1. Both graphs are connected and 3-regular, so $P = (A + I)/4$ (Figure 2(a)).

Theorem 1 (Exact GCN/output-JK collision and memory separation)**.** *Consider the graph family $G_m^{\pm}$ defined above, with its binary one-hot inputs and unit-self-loop propagation $P_G = (A_G + I)/4$. For the GCN update specified in the setup, fix any pointwise activation applied identically to all vertices in both graphs and choose either sum or mean pooling. Then, for every integer $m \geq 3$, every depth $L \geq 0$, any layer widths, and every choice of weights and biases shared across vertices and between the two graphs, the graph outputs on G_m^+ and G_m^- are identical. Their tangent features coincide wherever derivatives exist, as do any existing infinite-width GNTK limits. Output-JK that adds or concatenates pooled layer outputs also collides; normalized distances are zero whenever normalization is defined. Conversely, for binary one-hot X and every odd p,*

$$\|M_p(G_m^+) - M_p(G_m^-)\|_F^2 = 4^{-p} > 0. \tag{4}$$

Thus, for this graph family, fixed odd-order moments retain attribute–topology distinctions that the studied GNTK fails to represent at any depth.

Mechanism of the exact collision. Within each attribute group, states are identical. Group propagation uses $Q_+ = \left[\begin{smallmatrix} 3/4 & 1/4 \\ 1/4 & 3/4 \end{smallmatrix}\right]$ and $Q_- = \left[\begin{smallmatrix} 1/4 & 3/4 \\ 3/4 & 1/4 \end{smallmatrix}\right] = SQ_+$, where S swaps the groups. At layer ℓ, their states differ only by S^ℓ. Balanced pooling removes this permutation, giving identical outputs and gradients. Memory instead retains which original attribute receives each signal. Appendix A.7 gives the proof and normalized-kernel calculation.

Corollary 2 (When the lost interaction determines the label)**.** *Draw any size $m \geq 3$, then independently draw $Y \in \{+1, -1\}$ uniformly and return G_m^Y. Every classifier using only a dynamic or output-JK representation covered by Theorem 1 has population accuracy at most $1/2$. First-order memory admits a linear separator with accuracy one.*

Proof. Conditional on m, both labels have the same dynamic representation and equal probability. Write $M_{1,ij} = [M_1(G)]_{ij}$ for the (i, j) entry of the first-order moment matrix. Here $i, j \in \{1, 2\}$ index the two node-attribute classes, and the leading subscript 1 denotes one propagation step. In contrast, the linear statistic $t(G) = M_{1,11} + M_{1,22} - M_{1,12} - M_{1,21}$ is $+1/2$ on G_m^+ and $-1/2$ on G_m^-. Its sign recovers Y at every size and is preserved by positive feature normalization. □

This constructed-task result establishes separator existence, not finite-sample SVM success or MUTAG's labeling mechanism.

A general criterion behind the example. An equitable partition groups nodes with equal total propagation weight to each group. Its *quotient matrix*, such as Q_+ above, propagates group-constant states. This yields a general collision criterion.

Proposition 3 (Quotient-permutation collision criterion)**.** *Suppose two graphs admit equitable partitions with the same class proportions π and the same class-constant input features. Let their quotient operators be Q and SQ, where S is a permutation satisfying $SQ = QS$ and $\pi^\top S = \pi^\top$. Then every depth of the shared GCN has identical mean-pooled output on the two graphs, as does any combination of these pooled layer outputs. Their empirical NTKs therefore collide wherever parameter derivatives exist, and any existing infinite-width limits inherit this collision.*

Let U_ℓ and V_ℓ collect the hidden states of the partition classes in the first and second graph at layer ℓ, respectively; each row is the common hidden state of the vertices in one class. The proof establishes $V_\ell = S^\ell U_\ell$. Since the common class-proportion vector satisfies $\pi^\top S = \pi^\top$, mean pooling yields $\pi^\top V_\ell = \pi^\top U_\ell$. Appendix A.1 gives the proof and a finite-depth perturbation corollary. The criterion concerns this aggregation, not every center-preserving message-passing architecture.

3.2 Distinct roles of interaction order and depth

The next pairs test whether first-order memory suffices and whether finite-order memory replaces nonlinear recursion.

Higher orders reveal additional distinctions. Figure 2(b) uses a specially attributed triangular prism and $K_{3,3}$. To vary graph size, let the integer $r \geq 1$ denote the number of vertices replacing each base vertex. Replace each base vertex by a clique of r vertices with the same node attribute, and each base edge by all edges between the corresponding cliques. Each resulting graph contains $6r$ vertices, with $r = 1$ recovering the original six-vertex graph. Let G_r and H_r denote the expanded graphs obtained from the attributed triangular prism and the attributed $K_{3,3}$ shown in Figure 2(b), respectively.

Proposition 4 (A strict third-order witness)**.** *For every $r \geq 1$, the resulting $6r$-vertex pair collides for every depth of the normalized GCN, Vanilla GNTK, and output-JK. Its fixed moments satisfy*

$$M_1(G_r) = M_1(H_r), \quad M_2(G_r) = M_2(H_r), \quad \|M_3(G_r) - M_3(H_r)\|_F^2 = \frac{1}{9216}. \tag{5}$$

Moments up to order two cannot distinguish this graph pair, whereas including the third-order moment makes it separable. Adding higher-order moments preserves this distinction as long as the third-order moment is retained.

Recursion still matters at a bounded memory order. The third pair consists of 7-vertex, 4-regular base graphs with complements C_7 and $C_3 \cup C_4$ (Figure 2(c)), followed by the same clique expansion. The input graphs are the complements of the displayed cycles.

Theorem 5 (Analytic depth-four separation)**.** *For every blow-up scale, this pair collides for GCN layers $0, 1, 2, 3$ but has strictly positive normalized last-layer ReLU GNTK distance at $L = 4$. Let $\alpha_\ell \geq 0$ denote the weight assigned to the layer-ℓ kernel in output-JK. The nonnegative layer-kernel sum in Eq. (9) also separates whenever $L \geq 4$ and $\alpha_4 > 0$. All fixed moments $M_0, \ldots, M_4$ agree, whereas M_5 differs.*

At layer four, node-type frequencies first differ; linear independence of the Gaussian ReLU features then makes the pooled features nonproportional (Appendices A.9 and A.10). Recursion adds a distinction absent at $P \leq 4$; the fifth moment resolves this particular memory collision.

Architectural scope. For the first pair with the one-hot inputs defined above, replace $(A + I)/4$ by $(A + 2I)/5$ and use a single ReLU neuron with weight $w = (1, -1)^\top$, zero bias, and mean readout. Its outputs on G_m^+ and G_m^- are $3/10$ and $1/10$, respectively (Appendix A.1). Our comparisons include this self-information reweighting alternative. For every fixed positive integer self-loop weight c, an adjusted graph family still collides at all depths, while unchanged unit-loop memory separates it (Appendix A.2). The pair depends on c; another weight can separate that pair.

A separate large-depth diagnostic. Appendix A.11 proves that equal degree profiles can produce vanishing normalized linear-GNTK distance despite shallow separation, including an equally weighted cumulative-readout rate. This linear attenuation result is distinct from the exact ReLU collision above; it does not assert universal raw-ReLU collapse.

Implications for the method. Section 4 combines input moments with recursion, controlling P and L separately. Let $\eta \in [0, 1]$ denote the memory-branch weight in the convex combination of the separately normalized dynamic and memory kernels. FM-GNTK separates the first family before and after diagonal normalization when $\eta > 0$, an included odd-order coefficient is positive, and the active branch diagonals are positive. Under these normalization conditions, retaining the first-order moment with positive weight in memory and using $\eta > 0$ preserve the perfect linear separability established in Corollary 2.

4 FIXED-MEMORY GNTK

The graph families in Section 3 motivate preserving both input attribute interactions and nonlinear recursive features. The first two families exhibit distinctions that pooled layer outputs lose, whereas the third retains a role for recursion when memory order is bounded. Fixed-Memory GNTK (FM-GNTK) addresses these requirements with an unchanged dynamic GNTK backbone and a separate branch that stores the input moments. The construction directly preserves the identified missing interactions while keeping interaction order and network depth as separate controls.

4.1 DYNAMIC MULTI-DEPTH BRANCH

The dynamic branch computes nonlinear graph representations at successive depths and retains them through a nonnegative sum of layer kernels. For a graph pair (G, H), initialize $\Sigma_0^{GH} = X_G X_H^\top$ and $\Theta_0^{GH} = \Sigma_0^{GH}$. A bias-free, He-scaled ReLU GNTK layer (He et al., 2015) is:

$$S_\ell^{GH} = P_G \Sigma_{\ell-1}^{GH} P_H^\top, \tag{6}$$

$$\Sigma_\ell^{GH} = \mathcal{C}_{\mathrm{ReLU}}(S_\ell^{GH}), \tag{7}$$

$$\Theta_\ell^{GH} = \Sigma_\ell^{GH} + \dot{\mathcal{C}}_{\mathrm{ReLU}}(S_\ell^{GH}) \odot (P_G \Theta_{\ell-1}^{GH} P_H^\top). \tag{8}$$

Here, $\mathcal{C}_{\mathrm{ReLU}}$ and $\dot{\mathcal{C}}_{\mathrm{ReLU}}$ are the standard ReLU covariance and derivative maps; the former is the degree-one arc-cosine kernel (Cho & Saul, 2009).

Let $b_G = |V_G|^{-1}\mathbf{1}_{|V_G|}$ and $b_H = |V_H|^{-1}\mathbf{1}_{|V_H|}$ denote the mean-pooling weight vectors for graphs G and H, respectively, where $\mathbf{1}_n$ is the n-dimensional all-ones vector. The graph-level layer kernel is $k_\ell(G, H) = b_G^\top \Theta_\ell^{GH} b_H$. The dynamic kernel combines these already pooled layer kernels:

$$K_{\mathrm{dyn}}^{(L)}(G, H) = \sum_{\ell=0}^{L} \alpha_\ell k_\ell(G, H), \qquad \alpha_\ell \geq 0. \tag{9}$$

We call this output-level combination Dynamic-JK; "Vanilla output" retains only k_L. Attaching independent, zero-mean-initialized trainable readouts to the selected GCN layers yields this kernel in the infinite-width limit (Appendix A.12).

Theorem 5 establishes the role of this branch: nonlinear recursion can separate graphs whose fixed moments agree through a bounded order. Theorem 1 identifies the information that it needs from memory: every layer kernel collides on that family, so their weighted sum also collides. Retaining the original attribute's pairing with its propagated context supplies the missing distinction.

4.2 FIXED STRUCTURAL MEMORY

The memory branch pairs original node attributes with their propagated counterparts and stores these interactions independently of the dynamic recursion. It uses the fixed input X at every order, so increasing the network depth does not overwrite the retained statistics. To expose distinctions at different propagation orders, we include moments from zero through P. For nonnegative order

weights β_p, define

$$T_p^{GH} = \Sigma_0^{GH} \odot \left(P_G^p \Sigma_0^{GH} (P_H^p)^\top \right), \tag{10}$$

$$K_{\mathrm{mem}}^{(P)}(G,H) = \sum_{p=0}^{P} \beta_p \, b_G^\top T_p^{GH} b_H. \tag{11}$$

Each entry of the Hadamard product multiplies the similarity of the original node attributes by the similarity of their propagated attributes. Mean pooling turns these node-pair interactions into a graph kernel with an explicit moment representation.

Proposition 6 (Explicit memory feature map)**.** *The memory kernel is PSD and has the explicit form*

$$K_{\mathrm{mem}}^{(P)}(G,H) = \sum_{p=0}^{P} \beta_p \langle M_p(G), M_p(H) \rangle_F, \quad \phi_{\mathrm{mem}}(G) = \bigoplus_{p=0}^{P} \sqrt{\beta_p}\, \mathrm{vec}(M_p(G)). \tag{12}$$

Thus the node-pair Hadamard construction computes an inner product of the input moments studied in Section 3. The direct-sum feature map keeps the selected orders in separate coordinates, linking the memory kernel directly to the first- and third-order separation results.

An independent product-feature branch gives the neural interpretation. For each order, multiply fixed, independent isotropic random projections of x_i and $(P^p X)_i$, mean-pool the products, and train a linear readout. Use independent copies across orders. Independence and isotropy yield Eq. (11) in the infinite-width limit (Appendix A.12). This branch represents the retained input interactions in a fixed feature space.

For computation, run the dynamic recurrence once to depth L and compute the moments once from X, then normalize and combine the branch kernels before fitting the SVM. Memory is computed independently of the evolving dynamic states and does not feed back into the recursion. Depth L controls the recursive branch, while order P controls the retained input interactions.

4.3 Normalization-compatible combination

We separately normalize the branch kernels and combine them in the classifier kernel. Let $\overline{K}_{\mathrm{dyn}}$ and $\overline{K}_{\mathrm{mem}}$ denote their diagonal-normalized forms as in Eq. (3). The final kernel is

$$\boxed{K_{\mathrm{FM}} = (1-\eta)\overline{K}_{\mathrm{dyn}} + \eta \overline{K}_{\mathrm{mem}}, \qquad 0 \le \eta \le 1.} \tag{13}$$

For positive diagonals of the active branches, the normalized mixture has $K_{\mathrm{FM}}(G,G) = 1$; an inactive branch is omitted at an endpoint. The standard direct-sum identity then gives

$$d^2_{K_{\mathrm{FM}}}(G,H) = (1-\eta) d^2_{K_{\mathrm{dyn}}}(G,H) + \eta d^2_{K_{\mathrm{mem}}}(G,H) \ge \eta d^2_{K_{\mathrm{mem}}}(G,H). \tag{14}$$

Eq. (14) connects memory separation to the kernel used by the classifier: a positive normalized memory distance remains positive in the full kernel whenever $\eta > 0$.

The graph families in Section 3 satisfy the required normalization condition. For nonzero feature maps, a positive normalized distance requires that the two features are not positive scalar multiples; unequal unnormalized features alone do not suffice. On our regular one-hot witnesses, every moment has total mass one. A difference in any positive-weight moment therefore makes the memory features nonproportional. The moment distinctions proved in Section 3 thus survive both branch normalization and positive-weight fusion.

An interior mixture, $0 < \eta < 1$, also retains positive normalized dynamic distances. With positive weights on the separating orders and layers, the full kernel therefore preserves both directions of separation established under the bounded memory-order comparison. The endpoints $\eta = 0$ and $\eta = 1$ recover dynamic-only and memory-only kernels, respectively. This guarantee concerns representation separation; predictive selection may favor a single-branch endpoint. Section 5 evaluates the predictive contribution of each branch and their combination.

Computational profile. With d input channels, memory requires a per-graph moment calculation and $O(Pd^2)$ cached features, followed by inner products. The moments are reused across graph comparisons, adding no further node-pair GNTK recursion. Appendix B.3 gives the full cost, including the graph-to-landmark computation retained by Nyström.

5 EXPERIMENTS

5.1 EVALUATION SETUP

We evaluate FM-GNTK on 10 TU datasets (Morris et al., 2020) and three large molecular benchmarks. TU inputs combine topology, categorical node labels, and a development-selected degree-one-hot channel, with nodewise ℓ_2 normalization; edge labels, continuous attributes, and multi-hop preprocessing are excluded. Five development repetitions select kernel and SVM configurations; frozen configurations are evaluated with 10 repetitions of stratified 10-fold CV. SD is across the 10 seed means. Development and reporting reuse the same graphs and labels with different partitions: *frozen-configuration repeated CV* (Appendix B).

PC-3 and MCF-7 (Wale et al., 2008) use one locked stratified 10-fold evaluation. Molhiv uses its official scaffold split (Hu et al., 2020), validation-based selection, train-plus-validation refitting, and 10 landmark/projection repeats. Matched kernels share splits, landmarks, budgets ($M = 500$ for PC-3/MCF-7; 450 for Molhiv), and classifier-selection procedures. Large-dataset inputs omit edge labels and appended degree features. Molhiv repeats measure algorithmic variability on one test set.

5.2 BENCHMARK RESULTS AND COMPONENT CONTRIBUTIONS

Tables 1 and 2 compare FM-GNTK with the graph-kernel and neural baselines described in Section 2. Dagger-marked reference values retain their source-specific protocols; our matched variants share inputs and splits (Appendix C.1).

Table 1: TU accuracy (%). †: inherited reference results with source-specific protocols and SD definitions. FM-GNTK reports mean ± SD over ten 10-fold CV seed means. Bold identifies our method; matched ablations are in Table 4.

Method	MUTAG	PTC_MR	BZR	COX2	PROTEINS	IMDB-B	IMDB-M
SP†	84.4±1.7	59.9±2.0	81.7±3.0	80.9±1.2	75.8±0.6	72.2±0.8	50.9±0.9
WL†	82.1±0.4	61.3±0.9	87.3±0.8	81.2±1.1	75.5±0.3	73.8±3.9	51.1±2.7
DGK†	87.4±2.7	60.1±2.6	87.4±5.5	78.3±0.3	75.7±0.5	67.0±0.6	44.6±0.5
WWL†	87.3±1.5	66.3±1.2	80.8±3.7	83.1±3.0	74.3±0.6	73.3±4.1	50.4±3.4
FWL†	85.7±7.5	59.3±7.3	87.6±5.0	79.2±2.9	74.6±3.8	72.0±4.6	50.0±2.9
GAWL†	87.3±6.3	59.0±4.4	88.3±3.8	79.9±4.9	74.7±3.0	74.5±4.1	51.7±5.2
ASP†	89.9±2.9	66.0±3.6	79.8±2.9	81.0±2.9	77.0±4.2	76.0±4.1	51.4±3.1
MMD-GK†	81.9±7.3	61.1±3.7	78.8±0.5	76.7±0.6	75.0±1.5	67.8±5.3	45.7±4.8
TWWL†	88.3±6.2	66.0±6.8	84.5±4.3	82.5±2.8	74.9±3.0	73.1±3.5	50.8±3.2
GNTK†	89.9±6.0	58.1±1.7	84.7±3.6	81.2±3.3	74.2±3.7	75.4±5.0	51.4±4.0
GraphQNTK†	88.4±6.5	59.4±9.9	83.1±4.7	79.5±1.6	71.1±3.2	73.3±3.6	48.1±4.3
FM-GNTK	**90.5±0.7**	**67.0±1.2**	**89.1±0.4**	**84.8±0.5**	**76.8±0.3**	**76.1±0.6**	**52.0±0.3**

Matched ablations inherit Full's structural settings and reselect their classifiers on development data. Full versus Dynamic-JK removes memory; Full versus Full/output removes multi-depth readout while retaining memory. Full is best or tied-best among matched variants on the seven TU tasks in Table 1. Across all 10 TU datasets, it improves Dynamic-JK on 9 and ties on one, averaging 0.791 accuracy points; the largest gains are on MUTAG ($+1.944$) and PTC_MR ($+1.842$). It improves Vanilla output and Memory-only on all ten, averaging 0.996 and 2.532 points, respectively (Appendix Tables 4 and 5; Figure 4). Reported gains are computed from unrounded means.

Full also has the highest matched mean on the three large datasets. Its gains over Dynamic-JK are $+1.096$ and $+1.047$ AUC points on PC-3 and MCF-7, while Molhiv is essentially tied ($+0.016$); BZR also ties Dynamic-JK. This matched design measures component contributions while holding Full-selected structural settings fixed.

Table 2: Large-dataset ROC–AUC (%). †: inherited neural-baseline results. The lower block uses a shared protocol. SD is across 10 folds for PC-3/MCF-7, and 10 landmark/projection repeats on one official Molhiv test split.

Method	PC-3	MCF-7	ogbg-molhiv
Archived neural-baseline results			
MLP†	61.15±2.32	62.29±1.39	64.90±2.44
GCN†	64.59±2.38	66.46±1.41	76.06±0.97
GraphSAGE†	75.17±2.32	70.90±1.86	69.23±3.19
GIN†	85.24±1.42	82.39±1.62	75.58±1.40
DIFFPOOL†	73.46±2.68	70.65±1.76	67.56±2.61
Matched kernel variants (this work)			
Vanilla output	85.09±2.25	82.37±1.34	75.46±1.07
Dynamic-JK	84.93±2.22	82.16±1.37	76.12±0.90
Memory only	66.19±4.91	67.50±1.84	73.36±0.42
Full / output	85.16±2.30	82.36±1.32	75.75±1.07
FM-GNTK (Full)	**86.03±2.27**	**83.21±1.37**	**76.13±0.91**

5.3 Independently tuned controls

A follow-up exact-kernel study on MUTAG, BZR, PTC_MR, and COX2 uses two repetitions of five outer folds and three inner folds. Kernel, input, and classifier selection is contained within outer-training data. Controls include self-loop reweighting, input retention, marginal-product memory, and zero-order memory (Appendix D). Full exceeds Dynamic-JK on PTC_MR by 0.87 points and is lower on the other three datasets. On PTC_MR, it exceeds marginal-product memory but scores below the zero-order mixture. Input retention achieves the highest mean accuracy across the four datasets in the label-only comparison.

Independently tuned IMDB-M and AMES comparisons yield small Full–Dynamic gains with intervals spanning zero (Appendix E). The matched IMDB-M graphs are input-isomorphic; memory ranks highest on the prespecified AMES matched-pair subset. The N–O diagnostic does not meet its incremental-prediction criterion.

5.4 Witness and approximation checks

Exact checks reproduce the theoretical thresholds: first-order memory separates the first family, cumulative memory first separates the second at $P = 3$, and the third first separates dynamically at $L = 4$ while $P = 3$ memory collides (Appendix Figure 3). Development-only sweeps favor intermediate depth and positive memory weight on MUTAG, PTC_MR, and IMDB-M (Figure 5). On MUTAG, Nyström with $M = 80$ gives a mean approximate–exact accuracy gap of -0.010 points over 30 repeats. Appendix C reports approximation error, runtime, and memory measurements for this shared approximation scheme.

6 Discussion and Conclusion

Fixed memory preserves input attribute–structure interactions that the studied GNTK loses at every depth. Constructed classification tasks establish the consequence of this loss, and separation survives normalization and positive-weight fusion on these families. The depth-four construction demonstrates the distinct value of nonlinear recursion at bounded memory order.

Matched comparisons favor Full overall, while independently tuned comparisons vary in both direction and magnitude. Molecular examples illustrate the attribute–topology relations motivating this design. FM-GNTK turns the separation results into a computable kernel that preserves the identified interactions alongside nonlinear propagation.

Reproducibility Statement

Appendix A supplies complete proofs, including the analytic ReLU separation bridge. The supplement includes exact graph certificates, kernel code, frozen configurations and archived summary tables. The new controlled audit additionally saves graph-indexed predictions, every inner selection score and all split indices. Input processing, selection protocols, Nyström budgets and limitations are specified in Appendix B and Appendix D. The supplied scripts reproduce the new certificates and audit directly; archived broad results retain their stated experimental provenance.

AI Use Statement

Generative AI tools were used solely for language polishing and manuscript revision. All authors have fully reviewed the entire manuscript, and take full responsibility for all content in this submission.

References

K.M. Borgwardt and H.P. Kriegel. Shortest-path kernels on graphs. In *Fifth IEEE International Conference on Data Mining*, pp. 74–81, 2005. doi: 10.1109/ICDM.2005.132.

Dexiong Chen, Laurent Jacob, and Julien Mairal. Convolutional kernel networks for graph-structured data. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1576–1586. PMLR, 2020. URL https://proceedings.mlr.press/v119/chen20h.html.

Youngmin Cho and Lawrence Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

Simon S Du, Kangcheng Hou, Barnabás Póczos, Ruslan Salakhutdinov, Ruosong Wang, and Keyulu Xu. Graph neural tangent kernel: Fusing graph neural networks with graph kernels. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1263–1272. PMLR, 2017.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1026–1034, 2015.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 22118–22133, 2020.

Wei Huang, Yayong Li, Weitao Du, Jie Yin, Richard Yi Da Xu, Ling Chen, and Miao Zhang. Towards deepening graph neural networks: A GNTK-based optimization perspective. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2103.03113.

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

Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of Medicinal Chemistry*, 48(1):312–320, 2005. doi: 10.1021/jm040835a. URL https://doi.org/10.1021/jm040835a.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Sanjukta Krishnagopal and Luana Ruiz. Graph neural tangent kernel: Convergence on large graphs. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17827–17841. PMLR, 2023. URL https://proceedings.mlr.press/v202/krishnagopal23a.html.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4602–4609, 2019.

Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. TUDataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.

Giannis Nikolentzos and Michalis Vazirgiannis. Graph alignment kernels using weisfeiler and leman hierarchies. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 2019–2034. PMLR, 2023. URL https://proceedings.mlr.press/v206/nikolentzos23b.html.

Kenta Oono and Taiji Suzuki. Graph neural networks exponentially lose expressive power for node classification. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1ldO2EFPr.

Keishi Sando, Tam Le, and Hideitsu Hino. Tree structure for the categorical wasserstein Weisfeiler–Lehman graph kernel. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=VwoSsFK22P.

Till Schulz, Pascal Welke, and Stefan Wrobel. Graph filtration kernels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8196–8203, 2022.

Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler–Lehman graph kernels. *Journal of Machine Learning Research*, 12(77):2539–2561, 2011.

Mahito Sugiyama and Karsten Borgwardt. Halting in random walk kernels. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Yan Sun and Jicong Fan. MMD graph kernel: Effective metric learning for graphs via maximum mean discrepancy. In *The Twelfth International Conference on Learning Representations*, 2024.

Yehui Tang and Junchi Yan. GraphQNTK: Quantum neural tangent kernel for graph data. In *Advances in Neural Information Processing Systems*, volume 35, pp. 6104–6118, 2022.

Therapeutics Data Commons. ADMET benchmark group. Official benchmark documentation, n.d. URL https://tdcommons.ai/benchmark/admet_group/overview/. Accessed September 15, 2026.

Matteo Togninalli, Elisabetta Ghisu, Felipe Llinares-López, Bastian Rieck, and Karsten Borgwardt. Wasserstein Weisfeiler–Lehman graph kernels. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

S. V. N. Vishwanathan, Nicol N. Schraudolph, Risi Kondor, and Karsten M. Borgwardt. Graph kernels. *Journal of Machine Learning Research*, 11:1201–1242, 2010. URL https://jmlr.org/papers/v11/vishwanathan10a.html.

Nikil Wale, Ian A Watson, and George Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 14(3):347–375, 2008.

Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5453–5462. PMLR, 2018.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

Pinar Yanardag and S. V. N. Vishwanathan. Deep graph kernels. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1365–1374, 2015.

Wei Ye, Hao Tian, Shuhao Tang, and Xin Sun. Enhancing shortest-path graph kernels via graph augmentation. In Albert Bifet, Povilas Daniušis, Jesse Davis, Tomas Krilavičius, Meelis Kull, Eirini Ntoutsi, Kai Puolamäki, and Indrė Žliobaitė (eds.), *Machine Learning and Knowledge Discovery in Databases. Research Track and Demo Track*, pp. 180–198, Cham, 2024. Springer Nature Switzerland.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

A Proofs and Derivations

This appendix supplies the proofs and exact constructions used in Sections 3 and 4. The three graph families are illustrated together in Figure 2. Appendices A.7, A.8, and A.10 give their edge sets or quotient descriptions, type certificates, and moment calculations. The depth-four separation uses the analytic feature-independence result in Appendix A.9; numerical kernel distances provide an independent check in Figure 3.

A.1 A general quotient criterion and its architectural boundary

Proof of Proposition 3. Let R_G, R_H be the class-membership matrices, with $P_G R_G = R_G Q$ and $P_H R_H = R_H S Q$. The initial features have the forms $X_G = R_G F$ and $X_H = R_H F$. Writing class states as U_ℓ, V_ℓ, the identity $V_\ell = S^\ell U_\ell$ holds at layer zero. If it holds at layer ℓ, then

$$SQV_\ell = SQS^\ell U_\ell = S^{\ell+1} Q U_\ell.$$

The shared affine transformation, a shared bias, and a row-wise activation commute with this row permutation, so the identity holds at the next layer. If π is the common vector of class proportions and $\pi^\top S = \pi^\top$, then $\pi^\top V_\ell = \pi^\top U_\ell$. The entire pooled output is the same function of the shared parameters. Equality of gradients follows where they exist, and any function of the pooled layer outputs is also identical. This includes output-level concatenation and nonnegative sums of layer NTKs.

For class one-hot features, lumpability gives $M_p(G) = \mathrm{diag}(\pi) Q^p$ and $M_p(H) = \mathrm{diag}(\pi)(SQ)^p$. These can differ because their rows remain indexed by the original class rather than being pooled after the permutation. This criterion diagnoses the collision; it does not assert that every pair of row-stochastic matrices satisfying it is realizable by a simple unweighted graph. □

An explicit self-loop counter-control. For the first family, replacing self-loop weight one by $c \geq 0$ gives

$$Q_+(c) = \frac{1}{c+3}\begin{bmatrix} c+2 & 1 \\ 1 & c+2 \end{bmatrix}, \qquad Q_-(c) = \frac{1}{c+3}\begin{bmatrix} c & 3 \\ 3 & c \end{bmatrix}.$$

With a scalar weight $w = (1, -1)^\top$ and mean-pooled ReLU, their outputs are $(c+1)/(2(c+3))$ and $|c-3|/(2(c+3))$, respectively. For $c \geq 0$ they agree precisely at $c = 1$. At $c = 2$ they are $3/10$ and $1/10$. With $w = (1, 1)^\top$, both outputs equal one. Thus even their pooled one-layer random ReLU features are not proportional as functions of w: continuity and Gaussian full support turn this into a strictly positive normalized covariance distance. Projection onto the trainable-readout component gives the same conclusion for the corresponding one-layer GNTK. Our collision theorem consequently cannot be extended to arbitrary self-loop weights or center-preserving residual architectures.

Corollary 7 (Finite-depth stability under continuous perturbations)**.** *Fix a witness pair, its vertex sets and input dimension, a finite depth $L_{\max}$, and a finite memory order with fixed nonnegative weights. Assume that the dynamic distances vanish through $L_{\max}$, that the normalized squared memory distance is $\Delta > 0$, and that the node variances and all branch diagonals used in normalization are positive at this pair. Keep a mixture weight $\eta \in (0, 1]$ fixed. For every $\epsilon > 0$, sufficiently small perturbations of its real-valued features and symmetric nonnegative edge weights, with the self-loops fixed, preserve $\max_{\ell \leq L_{\max}} d_{k_\ell}^2 < \epsilon$ and $d_{K_{\mathrm{mem}}}^2 > \Delta/2$. The perturbed full kernel therefore has squared distance greater than $\eta\Delta/2$.*

Proof. Self-loops keep degrees positive. Fixed-order moments and finite-depth ReLU covariance and tangent recursions are continuous near the witness, whose node variances and graph-kernel diagonals are positive. Diagonal normalization is consequently continuous there. Apply continuity to the positive memory distance and to the finitely many zero dynamic distances, and intersect the resulting neighborhoods. Equation (14) then gives the full-kernel bound. □

The neighborhood can depend on $L_{\max}$, graph size, memory weights, and the base pair. The result concerns continuous perturbations at a fixed finite budget.

A.2 Extension to each fixed positive integer self-loop weight

Fix $c \in \mathbb{N}_{>0}$ and $m \geq 3c$. There are two sets of vertices, $A = \{a_i : i \in \mathbb{Z}_m\}$ and $B = \{b_i : i \in \mathbb{Z}_m\}$. The node features are $x_{a_i} = e_1$ and $x_{b_i} = e_2$. Construct $G^+_{m,c}$ by joining each vertex within its own class to the vertices at offsets $\pm 1, \ldots, \pm c$, and by adding the cross-class edges $\{a_i, b_{i+s}\}$ for $s = 0, \ldots, c-1$. Construct $G^-_{m,c}$ using only cross-class edges $\{a_i, b_{i+s}\}$ for $s = 0, \ldots, 3c-1$. All indices are reduced modulo m and all edges are undirected.

Proposition 8 (Graph validity)**.** *Both graphs are simple connected $3c$-regular graphs on $2m$ vertices. The construction is valid also at the minimum values $c = 1, m = 3$.*

Proof. Since $m \geq 3c > 2c$, the offsets $\pm 1, \ldots, \pm c$ are nonzero and pairwise distinct modulo m. The cross offsets are distinct because their number is at most m. There are no loops or parallel edges. Each vertex of G^+ has $2c$ same-class and c other-class neighbors; each vertex of G^- has $3c$ other-class neighbors. In G^+ each class contains the cycle of offset one, and cross offset zero joins the two classes. In G^- offsets zero and one give a two-step path from a_i to a_{i-1}; all vertices of A and then all vertices of B are therefore connected. □

Let A_G denote the adjacency matrix, and distinguish the dynamic operator from the fixed memory operator:

$$P_c(G) = \frac{A_G + cI}{4c}, \qquad P_1(G) = \frac{A_G + I}{3c+1}.$$

These are exactly symmetric GCN normalization on these regular graphs with self-loop weights c and one, respectively. The equal-size attribute partition is equitable. For P_c, its two quotient matrices are

$$Q_+ = \begin{pmatrix} 3/4 & 1/4 \\ 1/4 & 3/4 \end{pmatrix}, \qquad Q_- = \begin{pmatrix} 1/4 & 3/4 \\ 3/4 & 1/4 \end{pmatrix} = SQ_+, \qquad S = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

In particular, $SQ_+ = Q_+S$ and the quotients do not depend on m or c.

All-depth collision and its exact scope

Theorem 9 (Fixed-self-loop all-depth collision)**.** *For each fixed positive integer c and each $m \geq 3c$, every depth of the normalized GCN using P_c has identical mean-pooled outputs on $G^+_{m,c}$ and $G^-_{m,c}$, for every common parameter setting. The same holds for any readout that depends only on the sequence of already pooled layer outputs. Consequently, their associated finite-width NTKs collide wherever parameter derivatives are defined, as does any existing infinite-width GNTK limit.*

Proof. Within a graph, vertices in each class retain the same hidden state. Write the two class states as the rows of $H^{\pm}_{\ell}$, starting at $H^+_0 = H^-_0 = I_2$. A layer with shared bias has the form

$$H^{\pm}_{\ell+1} = \sigma(Q_{\pm} H^{\pm}_{\ell} W_\ell + \mathbf{1} b_\ell^\top).$$

Here σ acts row-wise and hence commutes with row permutations. If $H^-_\ell = S^\ell H^+_\ell$, commutation of S and Q_+ gives

$$H^-_{\ell+1} = \sigma(S^{\ell+1} Q_+ H^+_\ell W_\ell + \mathbf{1} b_\ell^\top) = S^{\ell+1} H^+_{\ell+1}.$$

The relation follows for every layer by induction. Additional shared row-wise transformations preserve the permutation relation as well. Balanced pooling gives $\frac{1}{2}\mathbf{1}^\top H^-_\ell = \frac{1}{2}\mathbf{1}^\top H^+_\ell$. Every downstream function of these pooled states is therefore identical. These are identities as functions of the parameters; their derivatives are equal wherever defined. Thus their tangent feature maps coincide and their squared NTK distance is zero. Passing to an existing kernel limit preserves equality. For ReLU, the identities hold as functions everywhere and for ordinary derivatives almost everywhere; a common permutation- equivariant derivative convention at zero also preserves the computational gradient equality. □

The theorem includes output-level JK but not node-level concatenation of different layers before a nonlinear readout. Sum pooling also gives equal outputs within each fixed-m pair, because both have the same node count. The explicit all-m common-prototype assertion below uses mean pooling.

The unchanged unit-self-loop memory separates Define the original fixed moments

$$M_p(G) = \frac{1}{2m} X_G^\top P_1(G)^p X_G.$$

The quotient matrices for P_1 are

$$R_+ = \frac{1}{3c+1}\begin{pmatrix} 2c+1 & c \\ c & 2c+1 \end{pmatrix}, \qquad R_- = \frac{1}{3c+1}\begin{pmatrix} 1 & 3c \\ 3c & 1 \end{pmatrix}.$$

Equitability implies $P_1^p X_G = X_G R_\pm^p$, while $X_G^\top X_G = mI_2$. Hence $M_p(G^\pm) = \frac{1}{2} R_\pm^p$.

Proposition 10 (First-order separation without changing memory)**.** *For all the graphs above,*

$$M_1(G^+) - M_1(G^-) = \frac{c}{3c+1}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \qquad \|M_1(G^+) - M_1(G^-)\|_F^2 = \frac{4c^2}{(3c+1)^2} > 0.$$

Moreover, the normalized first-order memory kernel strictly separates the pair. The same is true for a finite stacked memory with nonnegative weights and strictly positive first-order weight.

Proof. Subtract the displayed matrices. Each M_1 has sum of entries equal to one. If their vectors were positive scalar multiples, their sums would force that scalar to equal one, contradicting the difference. They are therefore nonzero and not positive-collinear. Their normalized vectors are different, giving a positive squared distance in the normalized linear kernel. For a stacked feature map

$$\phi_{\mathrm{mem}}(G) = \bigoplus_{p=0}^{P} \sqrt{\beta_p}\ \mathrm{vec}(M_p(G)), \qquad \beta_p \geq 0, \quad \beta_1 > 0,$$

any positive collinearity would imply positive collinearity in its first-order block, which was just ruled out. □

For separately normalized kernels with nonzero diagonal entries,

$$\bar{k}(G,H) = \frac{k(G,H)}{\sqrt{k(G,G)k(H,H)}}, \qquad k_F = (1-\eta)\bar{k}_D + \eta \bar{k}_M,$$

and fixed $0 < \eta \leq 1$, squared distances satisfy

$$d_F^2(G^+, G^-) = (1-\eta) d_{\bar{k}_D}^2(G^+, G^-) + \eta d_{\bar{k}_M}^2(G^+, G^-) = \eta d_{\bar{k}_M}^2(G^+, G^-) > 0.$$

At $\eta = 1$, a dynamic normalization is unnecessary. A dynamic branch with zero diagonal must not silently be normalized by this formula; the statement assumes it is well-defined when that branch receives weight. The elementary PSD-mixture and distance identities are standard properties, not additional novel contributions.

An explicit classification consequence

Corollary 11 (A pair-balanced classification task)**.** *Fix $c \in \mathbb{N}_{>0}$. Choose any probability distribution π over $\{m \in \mathbb{N} : m \geq 3c\}$. Draw $m \sim \pi$, independently draw $Y \in \{+1, -1\}$ uniformly, and return $G_{m,c}^Y$ with label Y. For any fixed dynamic feature map covered by the theorem, every classifier depending only on that feature map has accuracy at most $1/2$. First-order memory admits a separator with accuracy one. A separately normalized positive-weight Full kernel also admits a linear separator with accuracy one, whenever its normalization is well-defined.*

Proof. Conditioned on m, the dynamic representation is identical for the two equiprobable labels. A deterministic classifier must make the same prediction on the pair and gets exactly one label correct; a randomized classifier independent of the hidden label also has conditional accuracy $1/2$. Averaging over π proves the bound. Abstention cannot improve accuracy when abstentions count as incorrect. The bound also holds if m is supplied to the classifier, since labels are balanced within each size.

For memory use the linear statistic

$$t(M) = M_{11} + M_{22} - M_{12} - M_{21}.$$

It takes values $(c+1)/(3c+1) > 0$ on G^+ and $(1-3c)/(3c+1) < 0$ on G^-. Its sign therefore equals Y for every m. Positive normalization does not change this sign.

For completeness, write u_+ and u_- for the unit-norm stacked memory features. They are distinct and independent of m. The normalized Full feature map is

$$\psi_F(G) = \big(\sqrt{1-\eta}\, v_D(G), \sqrt{\eta}\, u_M(G)\big).$$

Use a linear functional with zero dynamic component and memory component $u_+ - u_-$. Its values on the two classes are $\pm\sqrt{\eta}(1 - \langle u_+, u_-\rangle)$, strictly of the indicated sign. Alternatively, using the two Full prototypes themselves as a representer yields values $k_F(G, G^+_{m_0,c}) - k_F(G, G^-_{m_0,c}) = \pm\eta(1 - \langle u_+, u_-\rangle)$ for any $m_0 \geq 3c$. The dynamic terms cancel because the two reference graphs have the same dynamic feature. This proves existence of a classifier, not a training algorithm guarantee. □

The task is deliberately synthetic and its labels are specified by the construction. The result translates a representation collision into an exact population classification obstruction on a clearly stated task; it does not identify a bottleneck in MUTAG, AMES, or another natural dataset. It also does not guarantee that an SVM trained on a finite sample with a particular regularizer or a selected η learns the exhibited separator.

Counter-control and quantifier limitations

Proposition 12 (A different self-loop weight fixes this pair)**.** *Keep $G^+_{m,c}, G^-_{m,c}$ fixed but use any positive self-loop weight $c' \neq c$ in the dynamic propagation. One ReLU layer with weight vector $(1,-1)^\top$, no bias, and mean pooling distinguishes the pair.*

Proof. Now the denominator is $3c + c'$. The preactivations of the two classes are opposite numbers, with magnitudes $(c+c')/(3c+c')$ for G^+ and $|c'-3c|/(3c+c')$ for G^-. The pooled ReLU outputs are respectively

$$\frac{c+c'}{2(3c+c')}, \qquad \frac{|c'-3c|}{2(3c+c')}.$$

If $0 < c' < 3c$, equality requires $c + c' = 3c - c'$, or $c' = c$. If $c' \geq 3c$, equality would require $c = -3c$, impossible for $c > 0$. Thus every other positive c' distinguishes the pair with these weights. □

Our statement has the quantifiers

$$\forall c \in \mathbb{N}_{>0}\ \exists (G^+_c, G^-_c) \quad \text{such that dynamic aggregation with weight } c \text{ collides.}$$

It is not an assertion that one graph pair defeats all self-loop weights, nor that tuning a self-loop parameter cannot fix the constructed data. The integer restriction is explicit; this two-class simple-graph construction is not a proof for every positive real c.

A.3 The fixed-memory feature map

We first prove Proposition 6. Write $C_{G,p} = P^p_G X_G$. For a fixed order p, mean pooling of the Hadamard interaction gives

$$\begin{aligned} & b_G^\top \left[(X_G X_H^\top) \odot (C_{G,p} C_{H,p}^\top) \right] b_H \\ &= \frac{1}{|V_G||V_H|} \sum_{i \in V_G} \sum_{j \in V_H} \langle x_i, x'_j \rangle \langle C_{G,p}(i,:), C_{H,p}(j,:) \rangle \\ &= \left\langle \frac{1}{|V_G|} X_G^\top C_{G,p}, \frac{1}{|V_H|} X_H^\top C_{H,p} \right\rangle_F \\ &= \langle M_p(G), M_p(H) \rangle_F . \end{aligned} \tag{15}$$

The direct sum in Eq. (12) follows after multiplying each order by $\sqrt{\beta_p}$. It is an explicit feature map, hence the memory kernel is PSD. To see the per-node outer product explicitly, write $x_i = X_G(i,:)^\top$ and $c_{i,p} = C_{G,p}(i,:)^\top$ as column vectors. Then $M_p(G) = |V_G|^{-1} \sum_i x_i c_{i,p}^\top$. The node-pair Hadamard product and this graph-level outer-product feature are two equivalent ways to evaluate the same pooled kernel.

A.4 Relation to endpoint-weighted walk kernels

The (a, b) entry of M_p is

$$[M_p(G)]_{ab} = \frac{1}{|V_G|} \sum_{i,j} x_{i,a} (P_G^p)_{ij} x_{j,b}.$$

Expanding P_G^p sums the products of propagation weights along all length-p walks between i and j. The attributes of intermediate vertices do not enter this expression. For categorical one-hot inputs it is therefore an endpoint-attribute statistic, not the full sequence of labels along a walk. Symmetric normalization gives weighted walk mass; on irregular graphs it need not be a Markov transition probability.

More explicitly, put $S = X_G X_H^\top$, $n = |V_G|$, and $m = |V_H|$. For symmetric propagation matrices,

$$\langle M_p(G), M_p(H) \rangle_F = \frac{1}{nm} \operatorname{tr}(P_G^p S P_H^p S^\top) \tag{16}$$

$$= \frac{1}{nm} \operatorname{vec}(S)^\top (P_H^p \otimes P_G^p) \operatorname{vec}(S). \tag{17}$$

This places the feature in the product-graph/walk-kernel framework (Vishwanathan et al., 2010); it is not a claim that every labeled random-walk kernel equals K_{mem}. Our expressivity results identify when these particular fixed statistics distinguish graphs that the chosen recursive backbone maps to the same pooled representation.

A.5 Pairing-destroyed and zero-order controls

Let $x_i = X_G(i, :)^\top$ and $c_{i,p} = (P_G^p X_G)_{i,:}^\top$ be column vectors and define

$$\widetilde{M}_p(G) = \left(\frac{1}{|V_G|} \sum_i x_i \right) \left(\frac{1}{|V_G|} \sum_i c_{i,p} \right)^\top.$$

Replacing M_p by $\widetilde{M}_p$ keeps the same input channels and matrix dimensions but discards within-node center–context pairing. The resulting direct-sum feature and its linear kernel remain permutation-invariant and PSD. We normalize this kernel separately and use the same mixture coefficients as for joint memory. The zero-order control retains only $M_0 = X^\top X/|V_G|$; with a degree channel this already includes label–degree co-occurrences and must not be described as topology-free.

On the regular one-hot witness pairs, P is doubly stochastic and the label frequencies match. Consequently $\bar{c}_p = \bar{x}$ and $\widetilde{M}_p = \bar{x}\bar{x}^\top$ is identical on each pair at every order. Their M_0 also agrees. Thus marginal and zero-order additions cannot explain the strict first- and third-order witness separations. The joint, marginal, and zero-order features therefore isolate the center–context pairing on these witnesses. Their predictive contributions on real datasets are compared in Appendix D.

Degree augmentation on the regular witnesses. The theoretical separation is preserved when a finite-weight degree channel is appended to these regular one-hot witnesses and rows are normalized. On either member of a common-degree pair, the resulting input has the form $X' = XB$, where $B = [I, w_d \mathbf{1} e_d^\top]/\sqrt{1 + w_d^2}$ has full row rank. Thus $M_p' = B^\top M_p B$. A right inverse $BR = I$ recovers $M_p = R^\top M_p' R$, so moment differences cannot disappear. Nor can normalization identify the complete memory features: their proportionality would imply proportionality of the original moments, whose unit total mass forces the scalar to be one. The collision proofs allow arbitrary common class features, and the ReLU independence proof only needs independent initial type vectors, which the rows of B supply. Numerical distances need not retain their original values. This argument covers the stated regular witnesses, not arbitrary feature transformations or loss of the original label channels.

A.6 Input-retaining comparator

For a directly comparable input-retaining kernel, use independently parameterized propagated and initial-input projections at every layer. Equal branch variances yield the preactivation covariance and

tangent terms

$$S_\ell = \tfrac{1}{2}(P_G \Sigma_{\ell-1} P_H^\top + \Sigma_0), \qquad T_\ell = \tfrac{1}{2}(P_G \Theta_{\ell-1} P_H^\top + \Sigma_0).$$

Then $\Sigma_\ell = \mathcal{C}_{\mathrm{ReLU}}(S_\ell)$ and $\Theta_\ell = \Sigma_\ell + \dot{\mathcal{C}}_{\mathrm{ReLU}}(S_\ell) \odot T_\ell$. The Σ_0 in T_ℓ is the gradient contribution of the current layer's independently trainable input projection. This comparator uses the same multi-depth mean readout and separate diagonal normalization as Dynamic-JK. It is an explicitly defined input-retaining architecture, not an assertion that all residual GNNs share this kernel. The other architectural control simply uses $A + 2I$ and its corresponding symmetric degree normalization in the original recurrence.

A.7 First-order collision family

Let V_0, V_1 be the two attribute groups. Every node in G_m^+ has three closed-neighborhood entries in its own group and one in the other group, whereas every node in G_m^- has one in its own group and three in the other. The quotient propagation matrices are therefore

$$Q_+ = \begin{bmatrix} 3/4 & 1/4 \\ 1/4 & 3/4 \end{bmatrix}, \qquad Q_- = \begin{bmatrix} 1/4 & 3/4 \\ 3/4 & 1/4 \end{bmatrix} = SQ_+, \quad S = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{18}$$

The matrices Q_+ and S commute. At layer zero, every vertex in the same attribute group has the same state. Let $U_\ell^\pm$ be the two-row matrix of group states after layer ℓ. For any shared affine map followed by a row-wise pointwise activation, induction gives

$$U_\ell^- = S^\ell U_\ell^+. \tag{19}$$

Indeed, if Eq. (19) holds at ℓ, then $Q_- U_\ell^- = SQ_+ S^\ell U_\ell^+ = S^{\ell+1} Q_+ U_\ell^+$, and row-wise maps commute with the row permutation. Balanced sum or mean pooling is invariant to S, so the two graph outputs are the same function of the parameters at every depth. Their gradients are identical wherever the pooled functions are differentiable. The empirical NTK distance is therefore zero at such parameters, and the equality passes to any existing infinite-width limit. For the ReLU kernels considered here, the standard continuous-initialization NTK is understood through these almost-sure derivatives. Since the pooled output at every layer is identical, addition or concatenation of those pooled outputs cannot change the conclusion.

For the moment calculation, balanced membership and lumpability give

$$M_p(G_m^\pm) = \frac{1}{2} Q_\pm^p. \tag{20}$$

The symmetric and antisymmetric eigenvalues of Q_+ are 1 and $1/2$; those of Q_- are 1 and $-1/2$. Hence for odd p,

$$M_p(G_m^+) - M_p(G_m^-) = 2^{-(p+1)} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \tag{21}$$

whose squared Frobenius norm is 4^{-p}. For even p the two moments agree. This proves Theorem 1.

For completeness, with unit weights on cumulative orders $p = 0, 1$, $\|\phi_{\mathrm{mem}}(G_m^\pm)\|^2 = 13/16$ and their cross-inner-product is $11/16$. Thus the diagonally normalized memory distance is $2 - 2(11/13) = 4/13$. When the dynamic distance is zero and $\eta = 1/4$, Eq. (14) gives the full distance $1/13$. This is the quantity plotted in Figure 3.

A.8 A strict third-order hierarchy

We prove Proposition 4 using the middle pair in Figure 2. The labels below are categorical indices, represented by two-dimensional one-hot vectors in X.

The triangular-prism base graph has edge set

$$E_G = \{(0,1),(0,2),(0,3),(1,4),(1,5),\\ (2,3),(2,5),(3,4),(4,5)\}$$

and labels $(0,1,1,0,0,1)$. The $K_{3,3}$ base graph has

$$E_H = \{(0,1),(0,3),(0,5),(1,2),(1,4),\\ (2,3),(2,5),(3,4),(4,5)\}$$

and labels $(0, 1, 1, 1, 0, 0)$. Both are connected and 3-regular with three vertices of each label.

To see the all-depth GCN collision, assign types jointly to the two graphs: two nodes have the same type at layer $\ell + 1$ when their closed neighborhoods have the same count vector over layer-ℓ types. The common type-count histograms are

layer	0	1	2	3	4
histogram	$(3, 3)$	$(1, 4, 1)$	$(2, 2, 2)$	$(1, 2, 2, 1)$	$(2, 1, 1, 2)$

From the odd pattern to the even pattern, the common multiset of closed-neighborhood signatures is

$$2(0, 1, 2, 1),\ 2(1, 2, 1, 0),\ (1, 0, 2, 1),\ (1, 2, 0, 1), \tag{22}$$

and from the even pattern back to the odd pattern it is

$$2(2, 1, 0, 1),\ 2(1, 0, 1, 2),\ (2, 1, 1, 0),\ (0, 1, 1, 2). \tag{23}$$

For an explicit certificate of recurrence, the joint type assignments at layer three are $(2, 1, 2, 3, 1, 0)$ and $(1, 2, 0, 2, 1, 3)$, and at layer four they are $(0, 3, 1, 0, 2, 3)$ and $(0, 3, 1, 3, 0, 2)$. Applying the specified closed-neighborhood update to the layer-four assignments returns exactly the layer-three assignments, with a common lexicographic type dictionary. Thus the deterministic replacement update enters a genuine two-step cycle, not merely a recurrence of separately sorted histogram sizes. A GCN assigns one shared vector to every common type, independent of width and parameter values; equal type multiplicities imply equal pooled outputs at every layer. The same functional-equality argument as above proves the GNTK and output-JK collision.

Exact moment multiplication yields

$$M_1(G) = M_1(H) = \begin{bmatrix} 7/24 & 5/24 \\ 5/24 & 7/24 \end{bmatrix}, \tag{24}$$

$$M_2(G) = M_2(H) = \begin{bmatrix} 13/48 & 11/48 \\ 11/48 & 13/48 \end{bmatrix}, \tag{25}$$

$$M_3(G) = \begin{bmatrix} 49/192 & 47/192 \\ 47/192 & 49/192 \end{bmatrix}, \qquad M_3(H) = \begin{bmatrix} 1/4 & 1/4 \\ 1/4 & 1/4 \end{bmatrix}. \tag{26}$$

Thus $M_3(G) - M_3(H) = 192^{-1}\left[\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right]$ and its squared norm is $1/9216$.

For scale r, replace every base vertex by a same-label r-clique and every base edge by the complete bipartite graph between its two cliques. Every lifted vertex has ordinary degree $4r - 1$ and closed degree $4r$, with $P_r R = RP$, where R replicates base features and $R^\top R = rI$. Consequently,

$$\frac{1}{6r}(RX)^\top P_r^p (RX) = \frac{1}{6} X^\top P^p X, \tag{27}$$

so all moment values and the first-separating order are independent of r. This completes the proof of Proposition 4. The statement for a cumulative memory of order $P \geq 3$ assumes its $p = 3$ coefficient is positive.

A.9 An analytic bridge from neighborhood types to ReLU kernel separation

The following argument proves strict separation; numerical evaluation of the arc-cosine recursion is not used as a premise. We work on a finite collection of d-regular graphs with categorical one-hot inputs, common degree d, and $P = (A + I)/(d + 1)$. Types are assigned jointly across the collection. At layer zero they are input labels. At layer $\ell + 1$, the type of a node is its *normalized closed-neighborhood histogram* over layer-ℓ types. Unlike 1-WL, the previous center type is not separately retained. This is a replacement update, not the usual center-preserving color refinement.

Lemma 13 (Linear independence of ReLU ridge features)**.** *Let $u_1, \ldots, u_s$ be nonzero vectors in a finite-dimensional real inner product space, no two scalar multiples of one another. For a standard Gaussian w on their span, the functions $r_i(w) = \sqrt{2}\max\{0, \langle w, u_i\rangle\}$ are linearly independent in L^2. Consequently their covariance Gram matrix is strictly positive definite.*

Proof. An almost-everywhere zero linear combination of the r_i is everywhere zero, because it is continuous and the Gaussian has positive density on every nonempty open subset of the span. Fix i.

The hyperplane $H_i = \{w : \langle w, u_i \rangle = 0\}$ is distinct from every other H_j. If the span has dimension at least two, a finite union of the proper subspaces $H_i \cap H_j$ cannot cover H_i. Choose $w_0 \in H_i$ outside that union and restrict the alleged identity to $w = w_0 + tu_i$ for sufficiently small $|t|$. Every term with $j \neq i$ is affine on this interval: its argument has a fixed sign. The ith term, however, has a derivative jump $\sqrt{2}\|u_i\|^2$ at zero. Its coefficient must vanish. Repeating for all i proves independence. In dimension one the assumptions allow at most one vector, whose ReLU feature has positive squared norm. □

Lemma 14 (ReLU features distinguish replacement types)**.** *At every finite layer, the infinite-width He-ReLU covariance feature vectors associated with the distinct replacement types are linearly independent.*

Proof. The assertion holds at layer zero because the distinct label vectors are one-hot. Suppose the preceding type vectors $\phi_1, \ldots, \phi_s$ are linearly independent. A current type is a coefficient vector $a \in \mathbb{R}^s_+$ with $\sum_i a_i = 1$, and its preactivation vector is $u_a = \sum_i a_i \phi_i$. It is nonzero. If $u_a = cu_b$ for a real scalar c, independence gives $a = cb$. Summing the coefficients gives $c = 1$, and hence $a = b$. Thus different current types give noncollinear preactivation vectors. Their span is finite dimensional even though the preceding feature space may not be. Lemma 13 applies on that span and proves that their ReLU covariance Gram matrix is strictly positive definite. Any feature realization of that Gram matrix therefore has linearly independent type vectors, completing the induction. □

Theorem 15 (Graph-level separation from type frequencies)**.** *Let $h_G^{(\ell)}$ denote the frequency vector of the joint replacement types in G at layer ℓ. If $h_G^{(\ell)} \neq h_H^{(\ell)}$, the mean-pooled ReLU covariance kernel and the last-layer GNTK both have strictly positive graph distance before and after diagonal normalization. A multi-depth GNTK including layer ℓ with positive weight also separates the pair. Conversely, equal type frequencies at a layer imply identical finite-width GCN outputs at that layer for every shared parameter choice, and identical gradients wherever those gradients exist.*

Proof. Write the mean covariance feature as $m_G = \sum_t h_G^{(\ell)}(t)\phi_t$. By Lemma 14, $m_G - m_H \neq 0$ when the frequencies differ. More strongly, $m_G = cm_H$ would give $h_G^{(\ell)} = ch_H^{(\ell)}$. Both frequency vectors sum to one, so $c = 1$, which is impossible. The vectors are nonzero and noncollinear, proving strict separation after normalization as well as before it.

The last-layer NTK contains the covariance kernel as its trainable-readout contribution. Equivalently, in the recursion of Eq. (8), $\Theta_\ell - \Sigma_\ell$ is PSD on the joint node set: propagation preserves PSD, the derivative covariance is PSD, and their Hadamard product is PSD. Hence a graph-level NTK feature realization can be chosen as $m_G \oplus q_G$. If the complete vectors were collinear, projection onto the first component would make m_G, m_H collinear, a contradiction. The same projection argument applies to a positive-weight direct-sum component of a multi-depth kernel. Finally, for finite-width GCNs all nodes of a common replacement type share the same state by induction. Equal frequencies imply identical pooled functions of the shared parameters, and thus equal gradients at differentiability points. For continuous initialization, the NTK statement uses these almost-sure derivatives. □

This result is restricted to the stated one-hot, regular-graph setting. In particular, it does not assert that normalized GCNs implement 1-WL on arbitrary irregular graphs, or that arbitrary distinct real-valued node states remain linearly independent under ReLU.

A.10 Proof of the depth-four witness theorem

For transparency, the two 7-vertex base graphs use labels $x_A = (1, 0, 1, 0, 1, 0, 0)$ and $x_B = (1, 0, 0, 1, 1, 0, 0)$ and edge sets

$$\begin{aligned} E_A = \{&(0,1),(0,2),(0,5),(0,6),(1,2),(1,3),(1,6),\\ &(2,3),(2,4),(3,4),(3,5),(4,5),(4,6),(5,6)\},\\ E_B = \{&(0,3),(0,4),(0,5),(0,6),(1,3),(1,4),(1,5),(1,6),\\ &(2,3),(2,4),(2,5),(2,6),(3,6),(4,5)\}. \end{aligned}$$

As above, these labels specify one-hot input categories, not scalar input features. Both graphs are 4-regular. Their complements are C_7 and $C_3 \cup C_4$, respectively. Joint exact refinement produces

layer	0	1	2	3	4
A	$(4,3)$	$(1,6)$	$(2,5)$	$(1,2,4)$	$(2,0,2,3)$
B	$(4,3)$	$(1,6)$	$(2,5)$	$(1,2,4)$	$(2,1,0,4)$

The entries are counts in a *joint* type dictionary, including zeros, not separately sorted lists of counts. At each step signatures are ordered lexicographically and receive common type IDs in both graphs. The last row therefore supplies a nonzero type-frequency difference at layer four. Equal frequencies through layer three imply identical pooled GCN functions there. Theorem 15 proves strict normalized covariance and GNTK separation at layer four, including output-JK with a positive depth-four coefficient. Exact rational moment calculations give

$$\|M_p(A) - M_p(B)\|_F^2 = 0 \quad (p = 1, 2, 3, 4), \qquad \|M_5(A) - M_5(B)\|_F^2 = \frac{16}{478515625} > 0. \tag{28}$$

Also $M_0(A) = M_0(B) = \mathrm{diag}(4/7, 3/7)$ from their common label counts. Thus the complete memory through order four agrees, including its zero-order block. For a clique blow-up of scale r, each vertex has degree $5r-1$ and $P_r R = RP$. A neighborhood type's count vector is multiplied by r and its denominator changes from 5 to $5r$, leaving normalized signatures unchanged. Initial features are replicated, and mean pooling divides by $7r$ rather than 7. Covariance and tangent recursions therefore induce exactly the same graph kernels, so the result holds at every scale. The moment identity $R^\top R = rI$ also preserves all displayed M_p values. This completes the analytic proof of Theorem 5.

As an independent numerical check, a 100-decimal-digit evaluation of the bias-free He-ReLU GNTK gives zero graph distance at $L = 0, 1, 2, 3$ and $6.169658988 \times 10^{-5}$ at $L = 4$ for last-layer readout ($2.058223608 \times 10^{-5}$ for Dynamic-JK). These high-precision values are reported as reproducibility checks, not used to establish positivity. The complementarity claim is restricted to memory budgets $P \le 4$; it does not hold for this pair once the separating fifth-order moment is included.

A.11 Degree-profile collapse of the linear GNTK

Let $P_G = \widehat{D}_G^{-1/2}(A_G + I)\widehat{D}_G^{-1/2}$, let X_G be the degree-one-hot matrix, and let $b_G = |V_G|^{-1}\mathbf{1}$. Define $z_G^{(\ell)} = b_G^\top P_G^\ell X_G$. This diagnostic uses a bias-free, unit-variance linear network, so the covariance map is the identity, its derivative is one, and $\Theta_0 = \Sigma_0$. Thus

$$\Sigma_\ell^{GH} = P_G^\ell X_G X_H^\top (P_H^\ell)^\top, \qquad \Theta_\ell^{GH} = \Sigma_\ell^{GH} + P_G \Theta_{\ell-1}^{GH} P_H^\top. \tag{29}$$

Lemma 16 (Closed form of the linear tangent recursion)**.** *For all $\ell \ge 0$, $\Theta_\ell^{GH} = (\ell+1)\Sigma_\ell^{GH}$.*

Proof. The identity is immediate at $\ell = 0$. If it holds at $\ell - 1$, then

$$\Theta_\ell = \Sigma_\ell + \ell P_G \Sigma_{\ell-1} P_H^\top = (\ell+1)\Sigma_\ell.$$

□

Therefore $K_\ell^{\mathrm{lin}}(G,H) = (\ell+1)\langle z_G^{(\ell)}, z_H^{(\ell)}\rangle$, and diagonal normalization removes the scalar $\ell+1$.

For a connected graph with self-loops, the unit principal eigenvector is

$$u_G(v) = \frac{\sqrt{d_v + 1}}{\sqrt{\widehat{\mathrm{vol}}(G)}}, \qquad \widehat{\mathrm{vol}}(G) = \sum_v (d_v + 1). \tag{30}$$

Connectivity and the positive diagonal make the normalized propagation irreducible and aperiodic. Hence

$$P_G^\ell = u_G u_G^\top + R_G^{(\ell)}, \qquad \|R_G^{(\ell)}\|_2 \le \rho_G^\ell, \quad \rho_G = \max_{i \ge 2} |\lambda_i(P_G)| < 1. \tag{31}$$

If $h_r(G)$ is the number of degree-r vertices, then

$$[z_G^{(\infty)}]_r = \frac{(\sum_s h_s(G)\sqrt{s+1})h_r(G)\sqrt{r+1}}{|V_G|\widehat{\mathrm{vol}}(G)}. \tag{32}$$

Thus equal degree sequences imply the same nonzero limit z_∞. Moreover, because $X_G^\top X_G = \mathrm{diag}(h_r(G))$,

$$\|z_G^{(\ell)} - z_\infty\|_2 \le \sqrt{\frac{\max_r h_r(G)}{|V_G|}}\,\rho_G^\ell. \tag{33}$$

The normalization map $x \mapsto x/\|x\|$ is Lipschitz on a neighborhood whose norm is bounded below by $\|z_\infty\|/2$. Combining the two graph bounds and squaring gives, with $\rho = \max\{\rho_G, \rho_H\}$ and a pair-dependent constant C, for all sufficiently large L,

$$d^2_{K_L^{\mathrm{lin}}}(G, H) \le C\rho^{2L}, \qquad 0 \le \rho < 1. \tag{34}$$

For an equally weighted cumulative readout, use the direct-sum feature

$$\Phi_G^{(L)} = \bigoplus_{\ell=0}^{L} \sqrt{\ell+1}\, z_G^{(\ell)}. \tag{35}$$

The squared difference between the two direct sums is bounded by a convergent series $\sum_{\ell\ge0}(\ell+1)O(\rho^{2\ell}) = O(1)$, whereas $\|\Phi_G^{(L)}\|^2 = \|z_\infty\|^2(L+1)(L+2)/2 + O(1)$. For nonzero vectors a, b, $\|a/\|a\| - b/\|b\|\| \le 2\|a-b\|/\min\{\|a\|, \|b\|\}$. Applying this inequality gives normalized squared distance $O(L^{-2})$. The constants in both rates refer to the fixed graph pair.

The shallow-separable example used in our diagnostic is a chain of k copies of two seven-vertex gadgets. The first gadget has edges

$$(0,1),(0,4),(1,2),(1,4),(2,3),(4,5),(5,6),$$

and the second has

$$(0,3),(0,4),(1,2),(1,3),(1,4),(2,5),(2,6).$$

Join vertex 3 in consecutive copies of the first gadget, and vertex 5 in consecutive copies of the second. These inter-copy edges are bridges. The two graphs have the same degree sequence, while the first contains triangles and the second four-cycles. For $k \ge 2$, both degree histograms are $h_1 = k$, $h_2 = 3k+2$, $h_3 = 3k-2$. Relative to the second graph, the first has k more oriented degree-2-to-degree-1 edges and k fewer degree-3-to-degree-1 edges. Expanding the first-layer feature therefore gives

$$[z^{(1)}_{G_\triangle(k)}]_1 - [z^{(1)}_{G_\square(k)}]_1 = \frac{1}{7}\left(\frac{1}{\sqrt{6}} - \frac{1}{\sqrt{8}}\right) > 0. \tag{36}$$

To exclude collinearity after normalization, note that $\sum_r \sqrt{r+1}[z_G^{(1)}]_r = |V_G|^{-1}\sum_v \sqrt{d_v+1}$, since $P_G\sqrt{d+1} = \sqrt{d+1}$. This positive functional agrees for the two graphs, so proportionality would force equality, contradicting the displayed gap. They are therefore shallow-separable also after normalization, while Eq. (34) forces the deep distance to vanish. Since degree-one-hot inputs satisfy $X_G\mathbf{1} = \mathbf{1}$, they also give $\mathbf{1}^\top M_1(G) = z_G^{(1)}$. The displayed shallow gap therefore certifies a fixed first-order moment difference for this attenuation example.

A.12 Independent product-feature network origin

Dynamic readouts. Let $f_\ell(G)$ be a trainable scalar readout of the mean-pooled state at layer ℓ, with the standard NTK parameterization. The combined output is

$$f_D(G) = \sum_{\ell=0}^{L} \sqrt{\alpha_\ell}\, f_\ell(G).$$

For a shared trunk, independently initialized centered Gaussian readouts have zero conditional-mean cross-head tangent terms. At fixed finite depth, the standard i.i.d. infinite-width NTK limit concentrates these terms at zero, while each diagonal head contribution converges to k_ℓ (Jacot et al., 2018; Du et al., 2019). This gives $K_{\mathrm{dyn}} = \sum_\ell \alpha_\ell k_\ell$ as in Eq. (9); cross-head terms need not be zero at finite width. An alternative exact additive realization uses independently parameterized GCN copies for the different readout depths. Their parameter blocks are disjoint, so their finite-width NTKs add exactly, and their limits give the same K_{dyn}. Evaluating this limiting kernel still requires only the single deterministic recursion of Section 4.

Fixed product features. For each order p, draw mutually independent standard Gaussian vectors $w_{p,s}, v_{p,s} \in \mathbb{R}^d$, independently across p and s. With column node vectors x_i and $c_{i,p} = (P_G^p X_G)_{i,:}^\top$, define

$$u_{p,s}(G) = \frac{1}{|V_G|} \sum_{i \in V_G} \langle w_{p,s}, x_i \rangle \langle v_{p,s}, c_{i,p} \rangle = w_{p,s}^\top M_p(G) v_{p,s}. \tag{37}$$

The projections and graph inputs are fixed; only the coefficients $a_{p,s}$ in

$$f_{M,S}(G) = \sum_{p=0}^{P} \frac{\sqrt{\beta_p}}{\sqrt{S}} \sum_{s=1}^{S} a_{p,s} u_{p,s}(G)$$

are trained. Differentiation with respect to these disjoint readout coefficients gives the finite-feature NTK $\sum_p \beta_p S^{-1} \sum_s u_{p,s}(G) u_{p,s}(H)$. Independence and $\mathbb{E}[ww^\top] = \mathbb{E}[vv^\top] = I$ imply

$$\mathbb{E}[u_{p,s}(G) u_{p,s}(H)] = \sum_{a,b} [M_p(G)]_{ab} [M_p(H)]_{ab} = \langle M_p(G), M_p(H) \rangle_F. \tag{38}$$

For fixed finite P, the law of large numbers as $S \to \infty$ therefore gives exactly Eq. (12). This construction retains original-input interactions independently of the dynamic recursion.

Branch calibration. The final diagonal rescaling is applied to the limiting kernels. For a positive diagonal matrix D, $D^{-1/2} K D^{-1/2}$ remains PSD, and so does a nonnegative mixture of the separately normalized kernels. This is the kernel calibration used by the classifier in Eq. (13).

Numerical verification of the three witnesses. Figure 3 complements the graph drawings in Figure 2. It displays the normalized distances whose zeros and separating thresholds are established above.

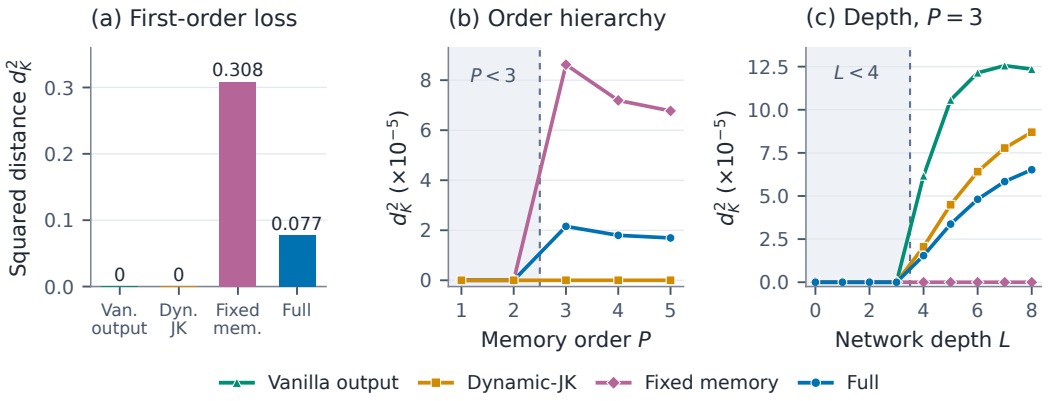

Figure 3: **Exact witness behavior.** (a) Vanilla output and Dynamic-JK collide on the first-order family, whereas memory separates. (b) The higher-order family first separates at cumulative order $P = 3$ and remains separated thereafter. (c) On the depth-needed family, a $P = 3$ memory collides while dynamic propagation first separates at $L = 4$. Values are squared distances of the separately normalized kernels. The proofs establish exact zeros and strict separation; numerical evaluation shows their magnitudes.

B Experimental Protocol and Reproducibility

B.1 Datasets, inputs, and locked selection

All reported Dynamic-JK kernels use $\alpha_\ell = 1$. Memory uses $\beta_p = \gamma^p$; an overall common scaling of these weights cancels under branch normalization. The ten TU datasets contain 188–1,500 graphs; PC-3 and MCF-7 contain 27,509 and 27,770 graphs; ogbg-molhiv contains 41,127 graphs. TU inputs use the provided categorical node labels and topology. IMDB datasets have no native node labels; their supplied constant tag is augmented by the development-selected degree channel. We do not use chemical-bond/edge labels, continuous node attributes, or precomputed multi-hop feature augmentation. Large-dataset inputs use their stored categorical node labels and topology, without edge labels or an appended degree feature.

The broad study's development selection used seeds 42–46. Candidate ranges covered $L \in \{1, 2, 3, 4, 6, 8, 10, 12\}$, $P \in \{1, 2, 3, 4, 5, 6, 8, 10, 12\}$, nonnegative mixture and geometric order-decay values, and a broad logarithmic SVM grid. Boundary audits were run only on development repetitions. After configuration freeze, final TU estimates used partition seeds 62–71 with ten stratified folds per seed. The reported standard deviation is over the ten seed-level means (divisor ten, as in the archived TU evaluator). PC-3 and MCF-7 use the ten folds of one stratified split (seed 62), reporting the sample SD across folds, not across ten independent ten-fold repetitions. For molhiv, configurations are selected on the official validation partition; the final model and landmarks use train plus validation and evaluate only on the official test partition, with ten landmark/projection seeds, reporting their sample SD. The fitting partition must therefore be considered when comparing to references trained on training data alone. Landmark sampling is restricted to the fitting portion of each evaluation. Configurations are frozen for these reporting runs. For TU, however, all graphs participated in the earlier development CV. Disjoint seed lists do not imply disjoint samples, and the estimates can retain development-selection optimism. We do not describe this design as nested CV or as an independent holdout. PC-3 and MCF-7 likewise reuse their datasets after development selection, so the same caveat applies. Appendix D gives the separately executed fold-contained audit and its more limited scope.

Table 3: Configurations fixed from development repetitions before locked reporting. γ is the geometric order decay, w_d is the degree-channel scale, and M is the Nyström landmark budget. The large datasets do not append a degree channel.

Dataset	L	P	η	γ	w_d	C	class weight	M	metric
MUTAG	6	3	0.03	0.75	1.00	100	none	–	accuracy
PTC_MR	6	1	0.50	1.00	0.00	1000	none	–	accuracy
BZR	4	1	0.01	0.10	0.25	1000	none	–	accuracy
COX2	2	1	0.99	0.10	0.25	30000	none	–	accuracy
PTC_FM	1	1	0.15	0.10	1.00	100	none	–	accuracy
PTC_FR	6	1	0.30	0.90	0.00	100	none	–	accuracy
PTC_MM	4	1	0.30	0.10	0.00	100	none	–	accuracy
PROTEINS	12	3	0.10	0.75	1.50	1	balanced	–	accuracy
IMDB-B	6	1	0.30	0.10	2.00	10	none	–	accuracy
IMDB-M	6	3	0.50	0.75	2.00	10	none	–	accuracy
PC-3	4	1	0.10	0.50	0.00	0.03	balanced	500	ROC–AUC
MCF-7	3	3	0.03	0.25	0.00	0.03	balanced	500	ROC–AUC
ogbg-molhiv	1	5	0.05	0.75	0.00	0.30	none	450	ROC–AUC

B.2 Ablation definitions

All broad-study internal variants share inputs, graph propagation, partitions and numerical precision. Structural parameters are inherited from the Full-selected configuration; each component reselects only its classifier on development repetitions:

- **Vanilla output**: the last-layer dynamic GNTK k_L and no fixed memory;
- **Dynamic-JK**: the nonnegative multi-depth dynamic kernel in Eq. (9) and no fixed memory;
- **Memory only**: Eq. (12) without the dynamic branch;

- **Full/output**: fixed memory plus the last-layer dynamic output, controlling for the multi-depth readout;
- **FM-GNTK (Full)**: Dynamic-JK plus fixed memory as in Eq. (13).

These ablations measure changes at a Full-selected structural operating point. They must not be interpreted as independently optimized competing methods. The nested audit separately tunes each method's eligible kernel and classifier choices, with all selection inside the current outer training fold.

Legacy implementation control. Some additional figures also show *Legacy*, an archived implementation with its structural interaction coefficient set to zero. It retains a unit additive covariance term at each layer, graph-average-trace scaling, and cumulative readout. It is therefore an additional implementation comparison, distinct from both the bias-free Vanilla output defined above and the literature GNTK reference. Removing one component from FM-GNTK does not produce this implementation.

B.3 Nyström construction and complexity

Computational profile. The fixed branch is computed once per graph. With d input channels, successively evaluating $C_p = P_G^p X_G$ and $M_p = |V_G|^{-1} X_G^\top C_p$ for $p = 0, \dots, P$ costs $O(P(|E_G| + |V_G|)d + (P+1)|V_G|d^2)$ time under sparse propagation. The $|V_G|$ term includes self-loops. Cached moments use $O((P+1)d^2)$ storage per graph, alongside $O(|V_G|d)$ working space; their kernel inner product costs $O((P+1)d^2)$. The dynamic branch retains the node-pair GNTK recursion. Thus increasing the memory budget does not repeat that recursion; the shared Nyström approximation controls the remaining graph-pair matrix at large scale.

Let $\mathcal{L}$ be a set of M landmarks drawn from the fitting portion, and let $\mathcal{N}$ index the graphs to be embedded. With $K_{\mathcal{L}\mathcal{L}} = U \operatorname{diag}(\mu_i) U^\top$ and numerical floor $\tau > 0$, we form

$$Z = K_{\mathcal{N}\mathcal{L}} U \operatorname{diag}(\max\{\mu_i, \tau\}^{-1/2}). \tag{39}$$

A primal linear SVM is fitted to Z. The feature storage is $O(NM)$ and the spectral step is $O(M^3)$; computing $K_{\mathcal{N}\mathcal{L}}$ costs $O(NMC_{\text{pair}})$, where $N = |\mathcal{N}|$ and C_{pair} is the graph-pair kernel cost. This replaces the full $N \times N$ Gram matrix and its dual solver with a rank-budgeted feature representation.

C Additional Results

C.1 Additional TU benchmarks and literature context

Main Tables 1 and 2 retain the complete baseline values from the authors' prior benchmark record. Their daggers identify inherited reference results. These entries were not rerun under the current protocol; their source-specific input, selection, and standard-deviation conventions may differ. The retained benchmark record is the immediate numerical source; the citations below identify methods rather than independently verifying the provenance of every numerical entry. The method references are SP (Borgwardt & Kriegel, 2005), WL (Shervashidze et al., 2011), DGK (Yanardag & Vishwanathan, 2015), WWL (Togninalli et al., 2019), FWL (Schulz et al., 2022), GAWL (Nikolentzos & Vazirgiannis, 2023), ASP (Ye et al., 2024), MMD-GK (Sun & Fan, 2024), TWWL (Sando et al., 2025), GNTK (Du et al., 2019) and GraphQNTK (Tang & Yan, 2022). The tables provide reference context alongside the common-split internal comparisons.

The neural references are MLP, GCN (Kipf & Welling, 2017), GraphSAGE (Hamilton et al., 2017), GIN (Xu et al., 2019), and DiffPool (Ying et al., 2018). These citations identify the architectures; the numerical entries are retained from the benchmark record rather than attributed to those architecture papers.

Matched TU components. Table 4 isolates the components under shared inputs, splits and Full-selected structural settings, with classifier reselection. It complements the external reference comparisons in the main text. All reported differences use unrounded archived means; table entries are rounded for display.

Table 4: Broad matched-component evaluation (accuracy in %). Mean ± SD across ten seed means, each from stratified 10-fold CV. Structural settings are Full-selected; components reselect their classifier. Reporting seeds differ from development seeds, but the graphs are reused. This is not nested CV.

Method	MUTAG	PTC_MR	BZR	COX2	PROTEINS	IMDB-B	IMDB-M
Vanilla output	89.00±0.63	65.51±1.96	88.98±0.64	83.61±0.76	75.95±0.14	75.41±0.57	50.94±0.33
Dynamic-JK	88.56±0.70	65.19±1.52	**89.13±0.35**	83.92±0.72	75.81±0.26	75.90±0.44	51.30±0.54
Memory only	88.14±0.94	60.11±1.01	82.33±0.72	84.69±0.56	75.57±0.48	75.14±0.35	51.24±0.44
Full / output	89.05±0.63	65.42±1.73	89.01±0.61	84.73±0.36	75.93±0.23	75.90±0.51	51.94±0.41
FM-GNTK (Full)	**90.50±0.74**	**67.03±1.16**	**89.13±0.35**	**84.80±0.49**	**76.82±0.27**	**76.05±0.55**	**52.03±0.32**

Table 5: Locked accuracy (%) on additional PTC variants.

Method	PTC_FM	PTC_FR	PTC_MM
Vanilla output	64.78±0.56	69.98±1.00	68.16±0.81
Dynamic-JK	64.55±0.65	71.17±0.96	68.84±0.77
Memory only	64.24±0.51	68.78±0.27	66.73±0.93
Full / output	64.78±0.66	70.00±0.94	67.77±0.75
FM-GNTK (Full)	**65.47±0.69**	**71.22±0.92**	**69.23±0.86**

Full improves Dynamic-JK by $+0.91$, $+0.06$, and $+0.39$ accuracy points on PTC_FM, PTC_FR, and PTC_MM, respectively. Together with PTC_MR in the main table, these related tasks show the range of matched-component gains, from a nearly neutral change on PTC_FR to a larger improvement on PTC_MR.

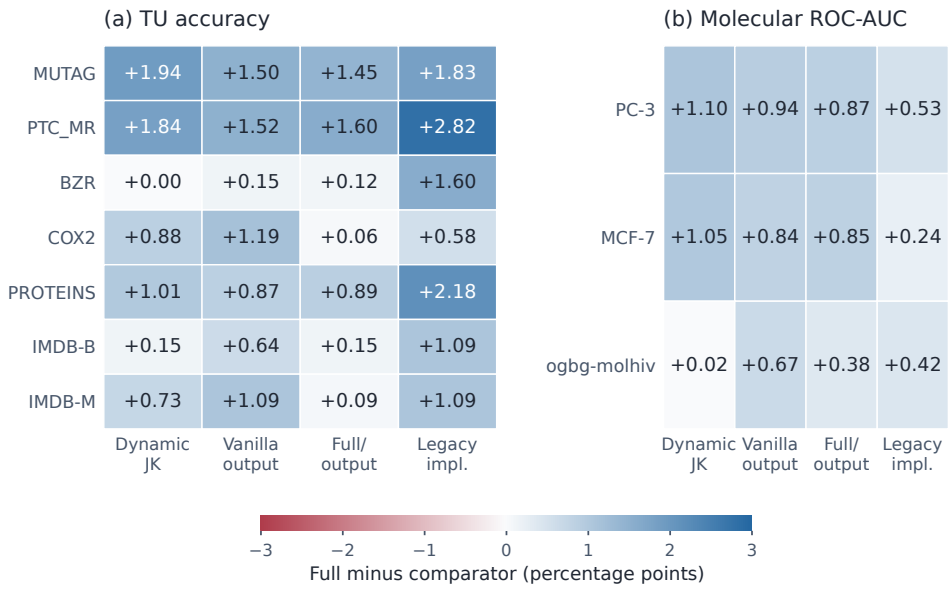

Figure 4: **Broad matched-component gains.** Each cell is Full minus a comparator in percentage points, at Full-selected structural settings with classifier reselection. Legacy is the separate archived implementation defined in Appendix B. These are descriptive matched-setting differences. TU results average ten partition-level means; PC-3/MCF-7 average ten folds; molhiv averages ten landmark/projection runs on its fixed test partition.

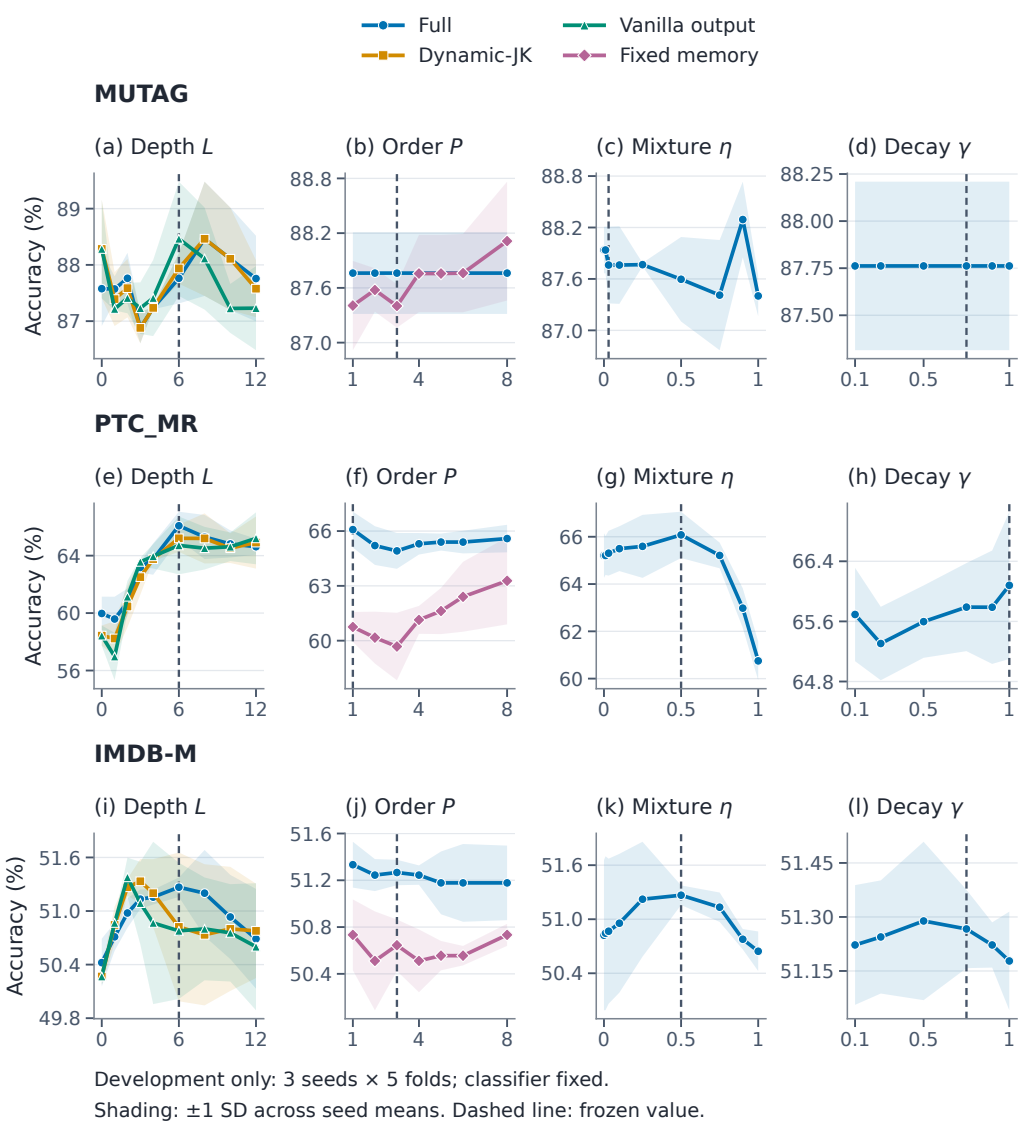

Figure 5: Development-only depth, interaction-order, mixture-weight, and decay sweeps on MUTAG, PTC_MR, and IMDB-M. These curves diagnose mechanisms and do not reuse locked reporting seeds.

C.2 Mechanism and sensitivity curves

PTC_MR provides the clearest depth example: the full model rises from roughly 60% at $L = 0$ to about 66% at $L = 6$ and then declines, while the preferred mixture has positive memory weight. IMDB-M also prefers an intermediate depth and positive η. MUTAG is small and its development curves are visibly noisier; we do not use that single sweep for a universal sensitivity claim.

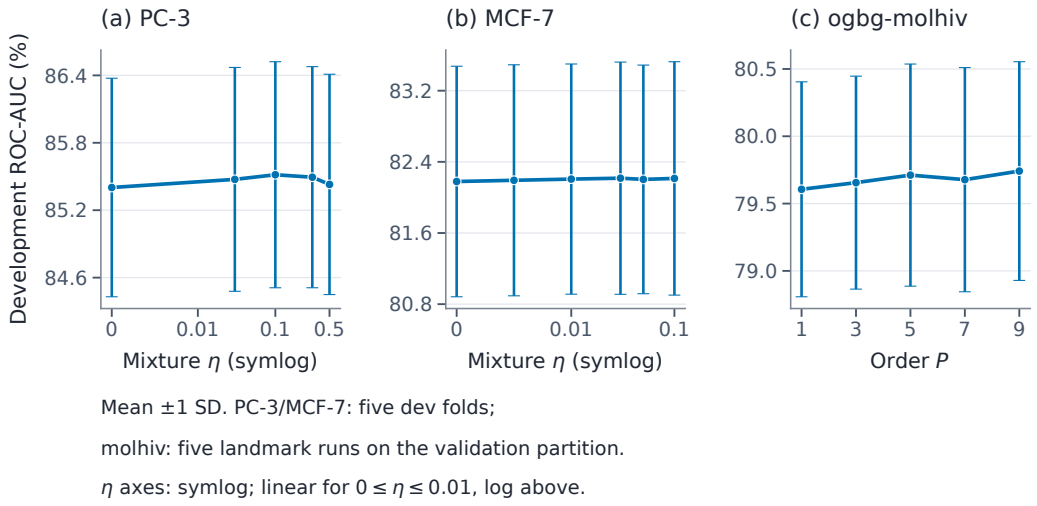

Figure 6: Development-only sensitivity on PC-3, MCF-7, and ogbg-molhiv. PC-3/MCF-7 use five development folds; molhiv uses five landmark/projection runs evaluated on validation data.

C.3 Exact versus Nyström and resource measurements

Table 6: Exact–Nyström agreement on MUTAG (30 repeats). Accuracy gap is Nyström minus exact, in percentage points.

M	Rel. Frobenius error	Alignment	Accuracy gap
20	2.99×10^{-3}	0.999995	-0.741
40	1.42×10^{-3}	0.999999	$+0.517$
80	5.58×10^{-4}	1.000000	-0.010
120	2.56×10^{-4}	1.000000	-0.341
160	1.37×10^{-4}	1.000000	$+0.000$

The alignment values in Table 6 are more stable than small-dataset accuracy, which changes in discrete fold-sized increments. At $M = 80$, the approximate and exact kernels already have alignment above 0.999999 and differ by only -0.010 accuracy points on average. Alignment is rounded to six decimal places: entries displayed as 1.000000 still have the nonzero approximation errors listed in the table.

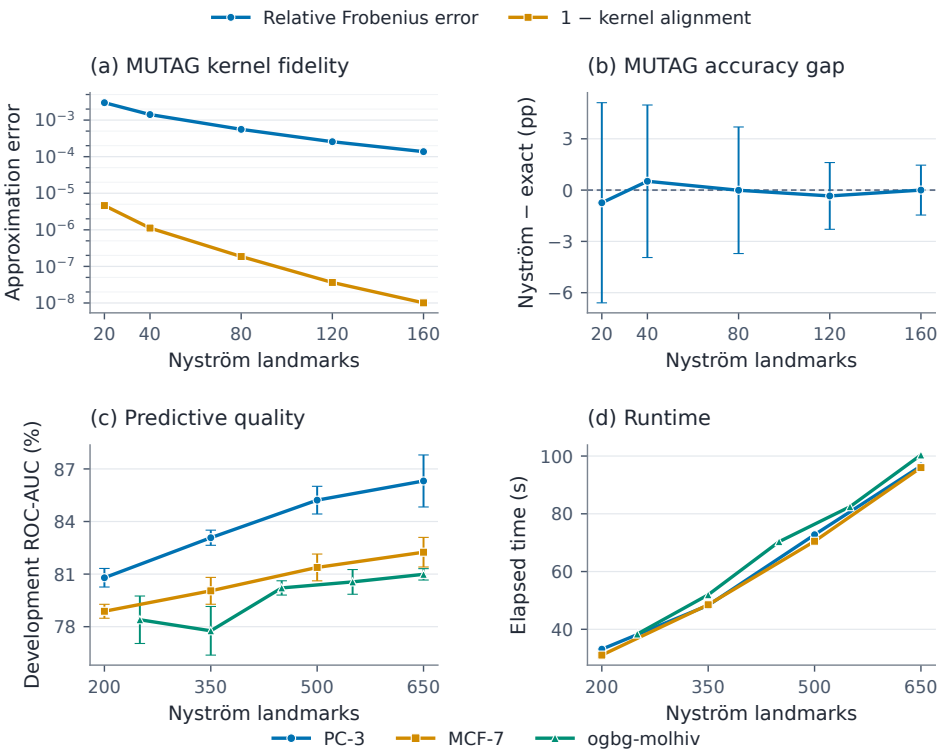

Figure 7: **Nyström quality and scalability.** Exact-kernel agreement is measured on MUTAG. The large-dataset budget–AUC and runtime curves use development data: the first three development folds on PC-3/MCF-7 and three landmark/projection runs on the molhiv validation partition. Landmarks are drawn from the corresponding fitting portion.

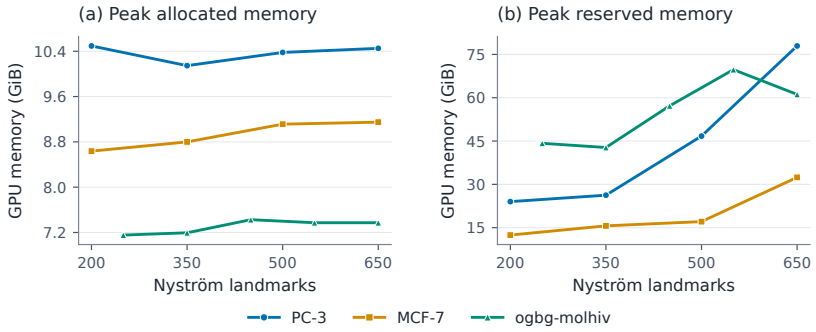

Figure 8: Allocated and reserved GPU memory over the large-dataset landmark budget audit. Allocated memory tracks active tensors; reserved memory includes the CUDA caching allocator and should not be interpreted as active usage.

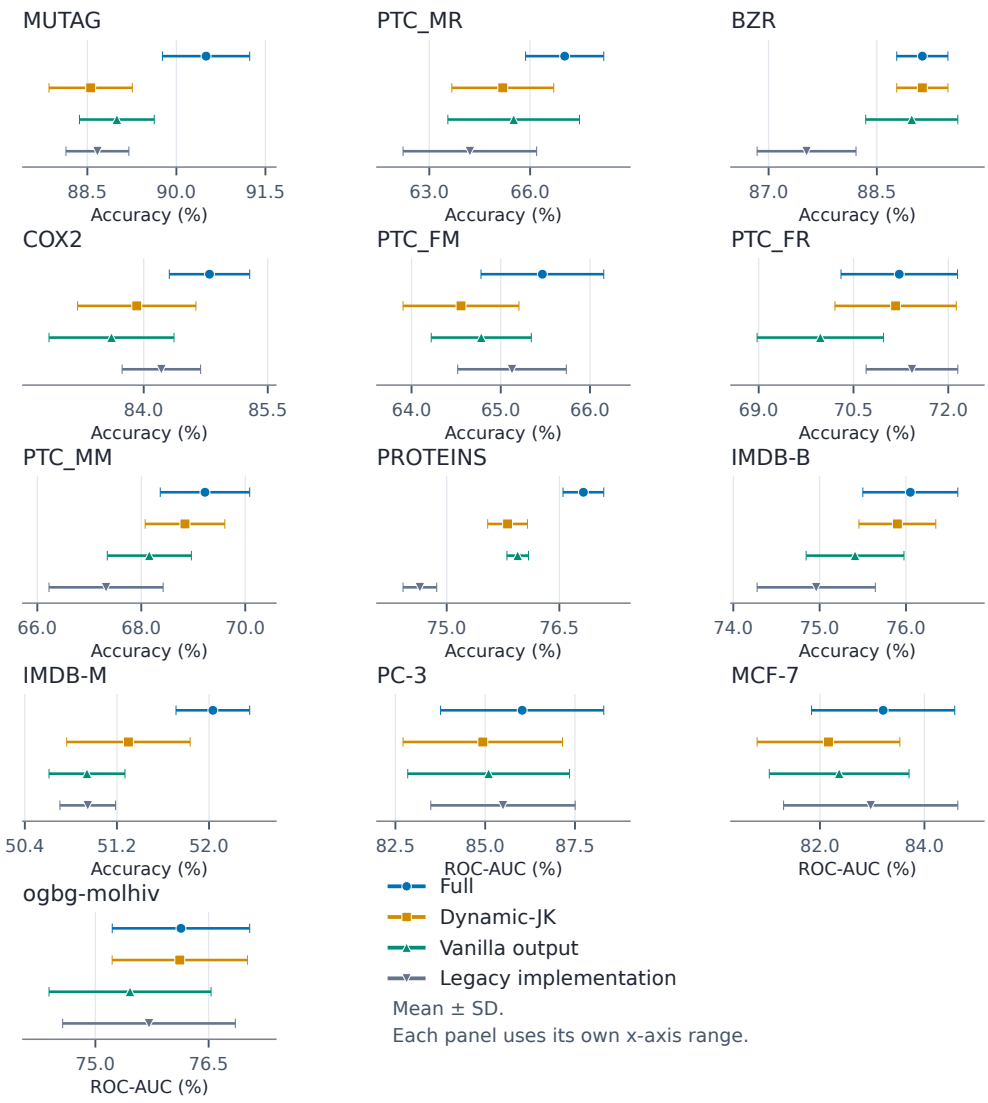

Figure 9: Locked result distributions: partition-level means for the ten TU datasets, fold scores for PC-3/MCF-7, and landmark/projection repeats on one test partition for molhiv. The error bars describe these different sources of variation, not a common independent-test uncertainty.

D FOLD-CONTAINED ROBUSTNESS AUDIT

Study sequence and scope. The broad experiment and its repeated-CV reporting were completed first. We then ran a node-label-only nested audit, followed by an extension that allows each method to select whether to append a degree-one-hot channel. Both stages are reported below. The degree extension follows the input design of the broad method and was specified after the label-only audit. These are post-development robustness checks on the existing datasets, with model selection contained within each outer training fold.

Selection and records. All four datasets use seeds 1729 and 2718, with five stratified outer folds and three stratified inner folds per outer training set. Eligible depths are $L \in \{1, 3, 6\}$; orders are $P \in \{1, 3\}$ with geometric decay 0.75. Memory mixtures allow $\eta \in \{0, 0.1, 0.5\}$, including the dynamic-only endpoint. $C \in \{0.1, 1, 10, 100, 1000\}$, with no class weighting, tolerance 10^{-3} and no iteration cap. Every fit is checked for solver convergence. The degree extension selects $w_d \in \{0, 1\}$; all node vectors are row-normalized. No edge labels, continuous attributes or input feature hops are used. Methods independently select their own eligible L, P, η, w_d, C, rather than inheriting Full's choices. The candidate counts differ because mixture methods have additional axes; the input, depth and classifier grids are shared.

Kernel matrices are exact and computed without graph-class labels. Inner training and validation indices are explicitly checked to exclude the current outer test fold. Model selection uses mean inner accuracy; ties use the listed candidate order and then increasing C. Saved files include every candidate's inner score, the selected parameters, all outer/inner split indices, and each test graph's true label, predicted label and decision score. The reported mean is the mean of two pooled out-of-fold accuracies; outer-fold SD is descriptive and is not used as an independent-sample confidence interval.

Architectural controls. Self2 changes only the self-loop weight to two. Input-retaining uses the explicit recurrence in Appendix A.6. Marginal and zero-order mixtures use Appendix A.5. The synthetic pairing controls isolate an exact algebraic distinction; the tables below evaluate the predictive contributions of these architectural choices on real graphs.

Table 7: Node-label-only audit (accuracy in %). Two 5×3 nested-CV repetitions with degree weight fixed at zero. The degree-selectable extension in Table 8 uses the same split seeds and kernel/classifier grid.

Method	MUTAG	BZR	PTC_MR	COX2
Dynamic-JK	77.66	87.78	62.35	83.19
Vanilla output	77.93	87.04	63.23	82.98
Self-loop weight 2	78.46	87.90	63.81	83.08
Input-retaining	85.90	86.79	60.76	82.23
FM-GNTK (Full)	75.80	88.64	63.52	82.98
Dynamic + marginals	76.60	87.90	61.63	82.33
Dynamic + zero order	77.39	88.52	63.52	82.12
Memory only	72.61	81.60	59.59	81.37

Implementation cross-check. The independent NumPy/SciPy implementation was compared with the archived PyTorch kernel on the first 12 MUTAG graphs at depths 1, 3 and 6, order 3, decay 0.75 and mixture 0.1. Maximum absolute Gram-entry differences were below 6.1×10^{-10}; changing the batching produced zero difference in the additional tested block. Candidate Gram matrices passed a float64 PSD smoke test. The checks cover these specified graphs, parameters, and numerical implementations.

Table 8: Independently selected controls (accuracy in %). Mean pooled out-of-fold accuracy over two 5×3 nested-CV repetitions. Each method selects its eligible kernel, degree input, and classifier parameters within the outer training data. This finite-grid audit uses a different selection protocol from Table 4.

Method	MUTAG	BZR	PTC_MR	COX2
Dynamic-JK	86.70	87.65	62.50	83.51
Vanilla output	88.83	87.04	63.23	82.44
Self-loop weight 2	88.30	87.16	63.66	83.40
Input-retaining	86.44	87.65	60.76	82.98
FM-GNTK (Full)	86.17	87.04	63.37	82.76
Dynamic + marginals	86.70	87.04	61.63	83.08
Dynamic + zero order	86.70	87.04	63.52	83.08
Memory only	86.97	84.07	59.88	82.98

E Real-Task Motivation and Predictive Checks

E.1 Matched MUTAG molecules: topology, separation, and prediction

We group all 188 MUTAG inputs by their heavy-atom count vector, edge count, and degree histogram. There are 36 non-singleton groups containing 140 graphs and 358 within-group pairs. Of these, 31 pairs have opposite labels, covering 34 graphs in eight groups. All 31 have non-isomorphic attributed inputs. The same three matching criteria are used throughout this analysis.

Figure 1 uses raw TU graph IDs 117 and 119, corresponding to legacy input IDs 13 and 132. The representative is the first eligible pair in the legacy-ID ordering, without consulting model outcomes. Both have 12 C, 3 N and 6 O nodes, 21 nodes in total, 22 edges, and degree-one/two/three counts 6/7/8. Their carbon-only subgraphs contain two six-cycles joined by one bridge. On the singly substituted cycle, the attachment carbon's shortest distance to the bridge endpoint is two in graph 117 and one in graph 119. The other cycle has two such attachments, at distances one and three in both graphs. The figure preserves the exact raw adjacency and atom types; no hydrogen or bond type is added. The supplied drawing script independently checks these counts and the two shortest-path distances.

The empirical conclusion is about the insufficiency of the matched summaries. A deterministic function of this signature must assign the same prediction to both members of any matched group; it cannot fit all opposite observed labels. Across this fixed sample, even the best such function makes at least ten errors, for an empirical fitting ceiling of $178/188 = 94.68\%$. This is neither a population bound nor a ceiling on pooled predictions from separately fitted CV models. Attachment position is an observed input difference, not an isolated chemical intervention. The contextual structural alerts in Kazius et al. (2005) use chemical information beyond our bond-agnostic input and concern different molecules.

Separation and prediction are distinct checks. With label-only inputs, Dynamic-JK numerically separates all 31 pairs at $L = 3$ and $L = 6$; its minimum squared normalized distance at $L = 3$ is 2.23×10^{-5}. Order-three memory also separates them, while zero-order label-only memory does not. The pairs therefore motivate attribute–topology relations through observed labels and structure; they are not instances of the constructed GNTK collision. With degree augmentation, zero-order memory can already contain label–degree co-occurrences and is not merely an atom-count feature.

Only 15 distinct opposite-label pairs in the inherited degree-selectable nested study are jointly held out and scored by the same model. Dynamic-JK and Full each rank 11 of these 15 pairs correctly; last-layer output ranks 12. Dynamic-JK and Full both have 50% pair-member classification accuracy, with no pair whose two members are both correctly classified. Ranking and threshold classification are different outcomes. Full selects its dynamic endpoint in seven of the ten outer fits, so this diagnostic shows no predictive improvement over Dynamic-JK. Pairs share molecules and are not independent observations. The dataset also has 15 attributed-input duplicate pairs; 44 of the 376 test-repeat records have an isomorphic input in the corresponding training fold. We retain the original data and splits.

E.2 INDEPENDENTLY SELECTED IMDB-M CONTROLS

This extension uses all 1,500 IMDB-M records, including duplicates and conflicting labels. The supplied node tag is constant; selected degree channels therefore encode degree–environment relations, not molecular attributes. Two stratified five-fold repetitions (seeds 1729 and 2718) each contain three inner folds. Exact float64 kernels and SVMs are selected independently for each method inside the outer training data. The grid is $L \in \{1, 3, 6\}$, $P \in \{1, 3\}$, degree weight $\{0, 1, 2\}$, decay 0.75, positive mixture weight $\{0.1, 0.5\}$, and $C \in \{0.1, 1, 10, 100, 1000\}$. Full and the two mixture controls add the dynamic endpoint $\eta = 0$; they do not add the memory endpoint. Positive Full is a separately labeled diagnostic that excludes zero, using the same saved inner scores. No search expansion follows the reported test outcomes.

Table 9: Independently selected IMDB-M controls. Accuracy is pooled over two out-of-fold repetitions of the same 1,500 graphs. Positive Full is a diagnostic selection rule, not a post-hoc replacement for Full.

Method	Accuracy (%)
Dynamic-JK	51.367
Vanilla output	51.133
Self-loop weight 2	51.300
Input-retaining	51.233
FM-GNTK (Full)	51.533
Dynamic + marginals	50.933
Dynamic + zero order	50.733
Memory only	50.933
Positive Full (diagnostic)	51.567

Relative to Dynamic-JK, Full corrects 50 prediction events and introduces 45 errors, netting five correct events among 3,000, or $+0.167$ accuracy points. Positive Full corrects 51 and introduces 45, or $+0.200$ points. These event counts include two predictions per graph. Resampling exact-input groups while holding the fitted models fixed gives descriptive 95% intervals of $[-0.471, +0.937]$ and $[-0.434, +0.995]$ points, respectively. The procedure uses 2,000 paired replicates; it does not capture overlapping-training or study-selection uncertainty. Full minus zero-order has difference $+0.800$ with interval $[+0.118, +1.797]$; Full minus marginals has $+0.600$ with $[-0.216, +1.619]$. These multiple comparisons are not adjusted significance tests.

The matched-group diagnostic cannot identify a recovered relation here. There are only 387 non-isomorphic observable inputs among 1,500 records, and every group matched by node count, edge count and degree histogram contains only one graph type. All 45,942 cross-label pairs inside these groups are therefore input-isomorphic. Of 3,000 outer test events, 2,384 have an isomorphic input in their training fold. Identical observable inputs receive the same prediction from any common deterministic isomorphism-invariant classifier. The original records and splits are retained in these comparisons.

E.3 AMES: PREDICTIVE VALUE BEYOND THE ILLUSTRATIVE PAIR

The AMES study follows the TDC ADMET benchmark construction (Therapeutics Data Commons, n.d.). Its archived source files contain 7,278 records (7,255 unique SMILES), with 5,093 training, 728 validation and 1,457 test records. All are retained. Predictors use heavy-atom categories and untyped topology, with validation-selected degree weight zero or one and row-normalized inputs. Bond type, charge, aromaticity, stereochemistry and chemical fingerprints are omitted. Five common seeds vary 128 training-only Nyström landmarks on the same partition; Nyström approximates the combined, separately normalized kernel, so Full receives no extra rank budget.

Methods independently select depths $\{1, 3, 6\}$, orders $\{1, 3\}$ and $C \in \{0.1, 1, 10, 100, 1000\}$ as applicable. The order decay is 0.75 and the positive mixture grid is $\{0.01, 0.1, 0.3, 0.5, 0.7, 0.9, 0.99\}$. Mean validation AUROC over the five landmark seeds selects one configuration per method; train and validation are then combined for classifier fitting, retaining the training-only landmarks. The endpoint-eligible Full search adds independently selected dynamic and memory candidates. Both this

search and Positive Full select $L = 6, P = 1, \eta = 0.1$, degree weight one and $C = 10$. These search sets differ from the smaller TU control grid and are not pooled into one benchmark estimate.

Table 10: AMES results on one fixed test partition. SD reflects five landmark draws, not five independent test sets. Accuracy is secondary.

Method	AUROC (%, mean $\pm$ SD)	Accuracy (%)
Dynamic-JK	78.744 ± 0.577	74.043
Memory only	74.842 ± 0.254	70.583
Positive Full	78.953 ± 0.605	74.139
Full (endpoints eligible)	78.953 ± 0.605	74.139
Dynamic + zero order	78.744 ± 0.577	74.043
Vanilla output	78.475 ± 0.734	73.933

The unrounded Full–Dynamic AUROC difference is $+0.208$ points, with a paired scaffold-cluster 95% interval $[-0.099, +0.525]$; Full–Memory is $+4.111$ with $[+1.141, +7.254]$. The 2,000 bootstrap replicates condition on the split, fitted models and grid; comparisons are not multiplicity-adjusted. In all 153 prespecified opposite-label strict matched test pairs, mean ranking accuracy is 65.556% for Dynamic, 69.020% for Memory and 67.778% for Full. Ties receive half credit, including input-isomorphic pairs. Memory has the highest point estimate on this matched subset. Excluding inputs isomorphic to training, validation or previous MUTAG inputs leaves 1,305 test records; their AUROCs are 77.770%, 73.683% and 77.951%, respectively. These subset results complement the full-partition results in Table 10; the overlap check covers the specified input archives.

Incremental information in a specified interaction. A subsequent training-side check asks whether the specific coordinate $q = [M_1]_{N,O} + [M_1]_{O,N}$ adds label information beyond atom counts, degree summaries, graph size and cross-fitted Dynamic scores. This coordinate measures N–O endpoint interaction, not a complete nitro-group substructure. On 5,093 training molecules, two five-fold repetitions compare a diagnostic logistic model with and without q. AUROC changes from 82.1300% to 82.1120% and from 82.2089% to 82.1956%, respectively. Average log-loss improvement is -0.000145, with conditional scaffold-cluster interval $[-0.000771, +0.001151]$. The prespecified training-side improvement criterion fails, so no new relation-deletion/restoration test is run. For this coordinate and diagnostic, the analysis does not establish incremental predictive information beyond the Dynamic score and the included summaries.

The original four-dataset controls, IMDB-M and AMES were developed after the broad benchmark study and after earlier results had been examined. Their finite grids and split-contained selection make their comparisons interpretable as post-development checks, rather than as confirmation on previously unexamined external data.