Supplementary Material

Class-Support Mismatch Dominates Protocol-Driven Optimism in Spatial Transcriptomics, with a Larger Residual Penalty for Graph Neural Networks

Chimdi Walter Ndubuisi

Department of Electrical Engineering & Computer Science
University of Missouri–Columbia

Contents

S1 Executed configuration

This section records what was actually run, as distinct from what was intended. Every number in the main text and in this supplement traces to one of the artifact tables listed in Section S10.

S1.1 Software environment

Table S1: Interpreter and library versions read from the running interpreter that produced the result tables. $n = 1$ environment; analysis unit = software environment. Not aspirational pins.

Component	Version
Python	3.12.3
Platform	Linux 6.17.0 x86_64, glibc 2.39 (Ubuntu 24.04.4 LTS)
numpy	2.4.3
scipy	1.16.3
pandas	2.3.3
scikit-learn	1.8.0
scanpy	1.12.1
anndata	0.12.10
statsmodels	0.14.6
networkx	3.6.1
matplotlib	3.10.8
torch	2.11.0+cu128 (CUDA)
torch-geometric	2.7.0
ripser	0.6.14
persim	0.3.8
XGBoost	3.2.0
GraphST	1.1.1 (PyPI)
GPU	NVIDIA GeForce RTX 2080 Ti (11 GB), driver 580.159.03

S1.2 Datasets as executed

Table S2: Dataset provenance as verified by direct file inspection. $n = 6$ data objects; analysis unit = data file. "Dual protocol" indicates whether the file received both random and spatial-block cross-validation arms.

Object	Spots	Classes	Label source	Type	Dual protocol
Breast S1	3,798	11	Leiden res. 0.8 on combined expression + spatial graph	algorithmic	yes (primary)
Breast S2	3,986 (3,987 pre-QC)	12	Leiden res. 0.8, same local pipeline	algorithmic	no (transfer target)
Mouse brain (fluo)	2,800	15	Leiden res. 0.7, shipped inside the squidpy dataset	algorithmic	yes, previously mislabelled DLPFC
Mouse brain (H&E)	2,688	15	Leiden res. 0.7, shipped inside the squidpy dataset	algorithmic	no (block CV only)
"DLPFC" cache file	2,800	15	byte-identical copy of the mouse fluo file	algorithmic	yes, results published as DLPFC
DLPFC (12 sections)	47,280	7 (5 for donor Br5595)	Manual cortical layer annotation, Maynard et al. [2021]	expert	yes

Corrections carried by Table S2.

1. The file consumed under a DLPFC filename by the earlier benchmark script has identical observation names, variable names, cluster values, expression sum (1.1830895×10^7) and spatial

coordinates to the mouse-brain fluorescence file, and carries mouse gene symbols. Re-running that script's preprocessing on it reproduces the stored table entry bit-for-bit (SVM, fold 0, seed 42: macro-F1 = 0.9237596597082417; weighted-F1 = 0.9251084021685065). All five-method "DLPFC" inflation figures in earlier versions are mouse brain.

2. Both mouse files store `uns['leiden']['params'] = {resolution: 0.7, random_state: 0, n_iterations: -1}`, and the cross-tabulation of `leiden` against `cluster` is a permutation matrix — zero rows and zero columns with more than one non-zero entry — in both files. The two files also disagree on the naming of the same brain structures, which cannot occur under a common atlas. There is no atlas structure identifier, reference-space coordinate, or annotation-source field anywhere in the observation, unstructured or multi-dimensional annotations.

3. Breast Section 2 is 3,987 spots as distributed and 3,986 after quality control (one spot dropped by the minimum-genes filter). A repository-wide search for the previously published figure of 2,512 returns no occurrence in any data file, log, result object or code path — only in prose. It is withdrawn.

4. Breast Section 1 carries **11** Leiden classes, not 7.

S1.3 Methods and hyperparameters as executed

Table S3: The six classifiers in the protocol ladder. $n = 6$ methods $\times$ 5 folds $\times$ 5 seeds per arm; analysis unit = fold–seed run. "Reads graph" determines whether the graph-masking axis is active for that method.

Method	Input	Configuration	Reads graph
SVM-PCA	50 PCs	RBF kernel, balanced class weights	no
XGBoost-PCA	50 PCs	200 trees, depth 6, learning rate 0.1	no
kNN-PCA	50 PCs	Euclidean, default k	no
MLP	50 PCs	feed-forward, same optimiser as the GNNs	no
GCN	50 PCs + graph	Kipf and Welling [2017], 3 layers	yes
GAT	50 PCs + graph	Veličković et al. [2018], 3 layers	yes

Preprocessing under the global arms: library-size normalisation to 10^4, $\log_{1p}$, 2,000 highly variable genes (Seurat v3 on raw counts), scaling with clipping at ± 10, PCA to 50 components, all fit on all 3,798 spots. Under fold-local arms every fittable step — gene detection filter, highly variable gene selection, scaling, PCA, and the class weights — is fit on training spots only. Fold-local PCA explains 23.4–34.7% of variance depending on the fold, and each fold selects its own 2,000 highly variable genes.

S1.4 Run inventory

Table S4: Runs executed. n as stated per experiment; analysis unit as stated. All counts are of completed runs recorded in the corresponding per-run table.

Experiment	Runs	Analysis unit
Protocol ladder, base arms (7 arms × 6 methods × 5 folds × 5 seeds)	1,050	independent spatial block ($n = 4$)
Protocol ladder, masked arms (4 arms × 6 methods × 5 folds × 5 seeds)	600	independent spatial block ($n = 4$)
Fold-local persistent homology (4 arms × 5 folds × 5 seeds)	100	independent spatial block ($n = 4$)
DLPFC within-section (12 sections × 2 methods × 3 protocols × 5 folds × 2 seeds)	720 attempted, 712 completed, 8 skipped	section within donor ($n = 12$, 3 donors)
DLPFC cross-section (LOSO 12 + LODO 3, × 2 variants × 2 methods × 2 seeds)	120	held-out section (24) or donor (6)
GraphST dual protocol (2 variants × 3 protocols × 5 folds × 2 seeds)	60	independent spatial partition ($n = 4$; folds 1 and 3 share a test partition)

S2 Split protocol and the fold-1/fold-3 duplication

The spatial block protocol overlays a 3 × 3 grid on the section, assigns whole blocks to train, validation and test roles, and removes every block adjacent to a test block as a buffer. Table S5 gives the resulting split sizes.

Table S5: Block-protocol split composition, breast cancer Section 1 (3,798 spots). Values are identical for arms B0, B1, B2, R2, R3 and the masked arms by construction, which is asserted in code. $n = 5$ nominal folds (4 independent spatial blocks); analysis unit = fold.

Fold	Train	Validation	Test	Buffer (excluded)	% excluded
0	874	429	363	2,132	56.1
1	809	480	428	2,081	54.8
2	857	446	439	2,056	54.1
3	809	401	428	2,160	56.9
4	791	480	446	2,081	54.8

S2.1 Folds 1 and 3 are the same spatial partition

Table S6: Fold-to-independent-block mapping, asserted in code. $n = 5$ nominal folds mapping onto 4 independent spatial blocks; analysis unit = fold. Folds 1 and 3 have identical train and test blocks and differ only in which block is held out for validation.

Fold	0	1	2	3	4
Train blocks	—	$\{6, 8\}$	—	$\{6, 8\}$	—
Test blocks	—	$\{7\}$	—	$\{7\}$	—
Validation blocks	—	$\{1\}$	—	$\{3\}$	—
Independent block index	0	1	2	**1**	3

Three assertions in the decomposition harness confirm this and pass: *folds 1 and 3 are the SAME spatial partition (train and test identical)* — observed train blocks $[6, 8]$ vs $[6, 8]$, test blocks [7] vs [7], validation blocks [1] vs [3]; *effective independent spatial blocks = 4* — observed mapping {0:0, 1:1, 2:2, 3:1, 4:3}; and *fold-local B2 features identical for folds 1 and 3* — maximum absolute difference exactly 0, because the two folds share their training spots.

Consequences, applied throughout.

1. The hierarchical bootstrap resamples 4 units, not 5. Every component table carries both the $n = 4$ (primary) and $n = 5$ (nominal, for comparability with earlier reports) columns; the $n = 5$ intervals are uniformly narrower and are anti-conservative.
2. The exact two-sided sign-permutation test over 4 paired blocks has a minimum attainable p-value of $2/2^4 = 0.125$. No paired fold-level test in this design can reach $p < 0.05$. Reporting a non-rejection as a null, or as equivalence, is not available here, and all such statements from earlier versions are withdrawn.
3. Any per-fold average that treats folds 1 and 3 as independent double-weights one spatial partition. The same defect appears in the earlier block-CV comparator table, where the conditions `A1_no_topology` and `B5_gat_no_topo` are the same 25 runs recorded twice (identical per fold and seed to machine precision); a mixed-effects model fitted on that table double-weights that condition, and the two entries must not be counted as two comparators.

S3 Zero-contamination graph audit

S3.1 What buffering left behind

Table S7: Cross-role edges in the buffered block protocol before pruning, and after. Counts are exhaustive enumerations of the $k = 6$ spatial graph (11,685 undirected edges over 3,798 spots), not estimates, so no interval is given. $n = 5$ folds; analysis unit = fold.

Role pair	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	After pruning
test–train edges	35	82	19	82	47	0 (all folds)
train–val edges	0	0	47	29	0	0 (all folds)
test–val edges	0	0	0	0	0	0 (all folds)
buffer–train edges	58	67	127	38	49	0 (all folds)
buffer–val edges	91	133	38	79	133	0 (all folds)
buffer–test edges	29	20	129	20	38	0 (all folds)
buffer–buffer edges	6,458	6,292	6,181	6,569	6,292	0 (all folds)

Table S8: Train–test and train–validation spot pairs within the message-passing receptive field, before pruning. After pruning every entry is 0 and every minimum distance is infinite. Exhaustive breadth-first-search counts, not estimates. $n = 5$ folds; analysis unit = fold.

Pair	Hop	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4
test–train	within 1	35	82	19	82	47
test–train	within 2	153	370	101	370	217
test–train	within 3	413	1,027	287	1,027	614
test–train	minimum distance	1	1	1	1	1
train–val	within 3	0	0	614	391	0
train–val	minimum distance	18	24	1	1	24
test–val	within 3	0	0	14	0	0
test–val	minimum distance	32	24	2	10	24

The train–validation channel is worth stating separately: in folds 2 and 3 the validation set was within one hop of the training set, so the early-stopping and model-selection signal was contaminated, not only the reported test score.

S3.2 Assertions and independent verification

Table S9: Hard assertions in the pruning code, which abort the run on failure. All pass in all folds. $n = 5$ folds $\times$ 8 assertions = 40 checks; analysis unit = fold–assertion.

Assertion	Observed (all 5 folds)
cross-role edges = 0	0
train–test pairs within 3 hops = 0	0
buffer nodes in message passing = 0	0 nodes / 0 edges
minimum train–test hop is infinite or > 3	infinite
all cross-role pairs disconnected	train–test, train–val, val–test all infinite
no self-loops	0
pruned graph symmetric	0 asymmetries
pruned edges a subset of the full graph	0 violations

A second verifier, written independently against the split definition files rather than against the pruning code, re-checks nine properties per fold: roles match the source split, cross-role edges zero, buffer-touching edges zero, buffer nodes degree zero, train–test pairs within 3 hops zero, train–validation pairs within 3 hops zero, no mixed-role connected components, no self-loops, and edge-index symmetry. All **45** checks pass.

S3.3 What pruning costs

Table S10: Connectivity retained after pruning, by role. Degree retention is the ratio of mean within-role degree after pruning to mean degree before. $n = 5$ folds $\times$ 3 non-buffer roles = 15 role–folds; analysis unit = role within fold.

Fold	Train retention	Val retention	Test retention	Test mean degree after
0	98.3%	96.6%	97.2%	6.05
1	97.0%	95.5%	96.1%	5.90
2	96.3%	96.9%	94.5%	5.81
3	97.0%	95.6%	96.1%	5.90
4	98.0%	95.5%	96.9%	5.98
Mean	97.3%	96.0%	96.2%	**5.93**

Across folds, 4,868–5,144 of the 11,685 global edges survive (41.7–44.0%, mean 43.2%), and **no train, validation or test node is isolated** in any fold. All 2,056–2,160 buffer nodes are isolated by construction; they enter no loss term and no evaluation set.

S4 Class support per fold

Table S11: Class support in the block folds, breast cancer Section 1, 11 Leiden classes. "Shared" is the number of classes present in both training and test, and is the set over which the primary closed-set macro-F1 is averaged. "Test-only" classes are unpredictable by construction. $n = 5$ nominal folds (4 independent spatial blocks); analysis unit = fold. Identical for arms B0, B1, B2, R3 and R3m by construction (asserted per class, per role, per fold).

Fold	Classes in train	Classes in test	Shared	Test-only	Test-only class IDs
0	6	3	3	0	—
1	6	5	5	0	—
2	7	5	3	2	{3, 7}
3	6	5	5	0	—
4	5	6	5	1	{5}

Under the random arms R0, R1 and R2 all 11 classes appear in train, validation and test in every fold, so the shared and all-class metrics coincide exactly and the open-set penalty is identically zero. Under the block arms and the class-matched random arms the open-set penalty (all-class minus shared-class macro-F1) is -0.051 to -0.068 pooled, depending on the arm.

In fold 2 the two test-only classes cover 343 of the 439 test spots (78%). Fold 2 is by far the hardest fold under every arm and drags every method's block-CV macro-F1 down; it is the single largest contributor to between-fold variance.

S5 The full protocol decomposition

S5.1 Arm definitions and the harness control test

The eleven arms are defined in the main text. Two diagnostic arms exist specifically to make the harness falsifiable: R3m and R3p share a split *and* a fold-local feature matrix and differ only in the graph handed to the model, so any method that never reads the graph must produce bit-identical predictions.

Table S12: Harness control test. Maximum absolute difference in macro-F1 between the two arms, over all runs. $n = 25$ runs per method per contrast (5 folds $\times$ 5 seeds); analysis unit = fold–seed run. Non-graph methods must be bit-identical; graph methods must respond.

Contrast	Method	Reads graph	Max $\lvert\Delta\rvert$ (closed-set)
R3m vs R3p	GAT	yes	1.27×10^{-1}
R3m vs R3p	GCN	yes	3.31×10^{-1}
R3m vs R3p	MLP	no	**0 (bit-identical)**
R3m vs R3p	SVM-PCA	no	**0 (bit-identical)**
R3m vs R3p	XGBoost-PCA	no	**0 (bit-identical)**
R3m vs R3p	kNN-PCA	no	**0 (bit-identical)**
R2m vs R2p	GAT	yes	6.79×10^{-2}
R2m vs R2p	GCN	yes	7.95×10^{-2}
R2m vs R2p	4 non-graph methods	no	**0 (bit-identical)**

A related expectation is wrong and worth correcting explicitly: R3m does *not* reproduce R3 for non-graph methods, and should not. R3m changes two things relative to R3 — the graph and the preprocessing scope — and only the first is a no-op for a method that never reads the graph. The non-graph gaps in the R3 − R3m column below are entirely preprocessing scope, and the R3p arm proves it.

S5.2 Additive decomposition, closed-set macro-F1

Table S13: Additive decomposition of the closed-set random-to-block gap. Hierarchical bootstrap 95% percentile intervals, 10,000 replicates, seed 2026, resampling spatial blocks then runs within blocks. $n = 4$ independent spatial blocks; analysis unit = independent spatial block; 150 pooled runs (100 non-graph, 50 graph) per contrast, 25 per method.

Component	Contrast	All methods	Non-graph	Graph
Training-set size	R0 − R2	0.022 [0.014, 0.029]	0.027 [0.019, 0.035]	0.011 [0.002, 0.020]
Class support	R2 − R3	**0.276 [0.121, 0.429]**	0.314 [0.159, 0.466]	0.200 [0.041, 0.356]
Spatial (residual)	R3 − B1	0.098 [0.065, 0.135]	0.044 [0.010, 0.084]	0.207 [0.121, 0.280]
Preprocessing scope	B1 − B2	0.051 [0.011, 0.096]	0.057 [0.000, 0.125]	0.039 [−0.008, 0.095]
Total	R0 − B2	**0.447 [0.332, 0.562]**	0.442 [0.342, 0.553]	0.457 [0.308, 0.597]
Message passing (random)	R0 − R1	—	**0.000 exactly**	0.016 [0.011, 0.021]
Message passing (block)	B0 − B1	—	**0.000 exactly**	0.033 [−0.012, 0.075]

Table S14: Per-method components, closed-set macro-F1. Hierarchical bootstrap 95% CI on $n = 4$ independent spatial blocks; 25 runs per method per arm; analysis unit = independent spatial block. Values of exactly 0.000 for the message-passing contrasts are structural, not estimates.

Method	Size (R0 − R2)	Class support (R2 − R3)	Message passing (B0 − B1)
SVM-PCA	0.022 [0.015, 0.030]	0.314 [0.126, 0.504]	0.000 exactly
XGBoost-PCA	0.033 [0.021, 0.044]	0.307 [0.112, 0.498]	0.000 exactly
kNN-PCA	0.037 [0.025, 0.048]	0.320 [0.233, 0.414]	0.000 exactly
MLP	0.016 [0.006, 0.027]	0.315 [0.145, 0.478]	0.000 exactly
GCN	0.009 [0.000, 0.019]	0.207 [0.042, 0.376]	0.050 [0.010, 0.104]
GAT	0.012 [0.002, 0.023]	0.192 [0.039, 0.348]	0.016 [−0.068, 0.086]

S5.3 The de-confounded residual: primary table

Table S15: Residual spatial-extrapolation component, closed-set macro-F1, before and after de-confounding, with both the primary $n = 4$ and the nominal $n = 5$ bases. Hierarchical bootstrap 95% CI; 25 runs per method per arm; analysis unit = independent spatial block. The $n = 5$ column treats all five folds as independent and is anti-conservative; it is shown only for comparability with earlier reports.

	Confounded (R3 − B2)		De-confounded (R3m − B2)	
Method	$n = 4$ (primary)	$n = 5$ (nominal)	$n = 4$ (primary)	$n = 5$ (nominal)
SVM-PCA	+0.102 [+0.046, +0.150]	+0.109 [+0.061, +0.151]	+0.031 [−0.064, +0.172]	+0.018 [−0.058, +0.133]
XGBoost-PCA	+0.043 [−0.059, +0.154]	+0.074 [−0.029, +0.175]	+0.024 [−0.017, +0.072]	+0.032 [−0.009, +0.078]
kNN-PCA	+0.188 [+0.059, +0.331]	+0.206 [+0.099, +0.322]	+0.081 [−0.068, +0.308]	+0.074 [−0.046, +0.245]
MLP	+0.071 [−0.063, +0.189]	+0.096 [−0.025, +0.211]	+0.032 [−0.066, +0.122]	+0.031 [−0.087, +0.145]
GCN	+0.232 [+0.137, +0.326]	+0.248 [+0.159, +0.351]	+0.177 [+0.071, +0.275]	+0.198 [+0.100, +0.300]
GAT	+0.260 [+0.202, +0.312]	+0.265 [+0.214, +0.316]	+0.251 [+0.213, +0.289]	+0.255 [+0.218, +0.293]
Graph pooled	+0.246 [+0.191, +0.297]	+0.256 [+0.204, +0.319]	+**0.214** [+0.160, +0.273]	+0.226 [+0.173, +0.288]
Non-graph pooled	+0.101 [+0.031, +0.172]	+0.121 [+0.051, +0.187]	+**0.042** [−0.039, +0.143]	+0.039 [−0.030, +0.119]
All methods	+0.149 [+0.096, +0.208]	+0.166 [+0.112, +0.222]	+0.099 [+0.038, +0.163]	+0.101 [+0.052, +0.152]
Family gap	+0.145 [+0.074, +0.212]	+0.135 [+0.066, +0.204]	+**0.172** [+0.054, +0.274]	+0.187 [+0.083, +0.289]

Two orderings are worth recording. Before de-confounding: GAT > GCN > kNN-PCA > SVM-PCA > MLP > XGBoost-PCA, with a margin of +0.045 between the weaker graph method and the strongest non-graph method. After: GAT > GCN > kNN-PCA > MLP > SVM-PCA > XGBoost-PCA, margin +0.097. The gap widens, which is the opposite of what an axis-asymmetry artifact would produce. Before de-confounding, four methods have intervals excluding zero (SVM-PCA, kNN-PCA, GCN, GAT); after, only GCN and GAT do.

S5.4 Splitting the confound into its two halves

Table S16: The confound in R3 − B2 split into a preprocessing half and a graph-masking half, closed-set macro-F1. Hierarchical bootstrap 95% CI; $n = 4$ independent spatial blocks; 25 runs per method per arm; analysis unit = independent spatial block. The graph half is structurally zero for non-graph methods.

Method	Graph + preproc (R3 − R3m)	Preprocessing (R3 − R3p)	Graph masking (R3p − R3m)
SVM-PCA	+0.071 [−0.078, +0.175]	+0.071 [−0.078, +0.175]	0.000 exactly
XGBoost-PCA	+0.019 [−0.094, +0.112]	+0.019 [−0.094, +0.112]	0.000 exactly
kNN-PCA	+0.107 [−0.025, +0.207]	+0.107 [−0.025, +0.207]	0.000 exactly
MLP	+0.039 [−0.064, +0.164]	+0.039 [−0.064, +0.164]	0.000 exactly
GCN	+0.055 [−0.010, +0.124]	+0.032 [−0.012, +0.086]	+0.023 [−0.057, +0.091]
GAT	+0.009 [−0.045, +0.068]	+0.012 [−0.036, +0.077]	−0.003 [−0.042, +0.035]
Graph pooled	+0.032 [−0.011, +0.072]	+0.022 [−0.003, +0.051]	+0.010 [−0.038, +0.055]
Non-graph pooled	+0.059 [−0.054, +0.155]	+0.059 [−0.054, +0.155]	0.000 exactly

S5.5 Open-set (all-class) results

Table S17: Residual spatial component on the open-set metric (macro-F1 averaged over all 11 classes, scoring absent classes as zero). Hierarchical bootstrap 95% CI; $n = 4$ independent spatial blocks; 25 runs per method per arm; analysis unit = independent spatial block. Rankings match the closed-set table, so no conclusion depends on the metric choice.

	Confounded (R3 − B2)	De-confounded (R3m − B2)
SVM-PCA	+0.092 [+0.034, +0.149]	+0.009 [−0.055, +0.100]
XGBoost-PCA	+0.052 [−0.033, +0.155]	+0.019 [−0.017, +0.062]
kNN-PCA	+0.150 [+0.053, +0.257]	+0.043 [−0.061, +0.182]
MLP	+0.069 [−0.051, +0.178]	+0.022 [−0.065, +0.102]
GCN	+0.203 [+0.113, +0.300]	+0.150 [+0.057, +0.253]
GAT	+0.227 [+0.155, +0.296]	+0.220 [+0.155, +0.280]
Graph pooled	+0.215 [+0.147, +0.282]	+0.185 [+0.126, +0.255]
Non-graph pooled	+0.091 [+0.024, +0.168]	+0.023 [−0.034, +0.086]
Family gap	+0.124 [+0.063, +0.184]	+0.161 [+0.052, +0.255]

S5.6 Graph density under each arm: the structural caveat

Table S18: Graph connectivity actually available to the graph methods under each arm. $n = 5$ folds × 25 runs per graph method; analysis unit = fold. The same masking rule has very different consequences on a contiguous partition than on a random one.

Arm	Graph	Edges kept (fraction)	Mean test-node degree	Isolated test nodes
R3m	masked	0.257	3.13	12.9%
R3p	global	1.000	6.14	0.0%
R2m	masked	0.074	0.67	49.0%
R2p	global	1.000	6.15	0.0%
B1 / B2	masked	**0.432**	**5.93**	**0.0%**

R3m is matched to B2 on the masking *rule* but not on the resulting *density*, and it cannot be: a random role assignment turns most neighbour pairs into cross-role pairs, whereas a block partition is cut only at its boundaries. This is a structural property of matching a spatial partition against a spatial graph.

The direction of the resulting bias runs *against* the surviving conclusion — R3m gives the graph methods less usable connectivity than B2 does, which can only depress R3m and shrink R3m − B2 for exactly the two methods the claim concerns. Its magnitude is bounded by what message passing is worth under a random partition at all, measured directly by R0 − R1 on the identical split: GCN +0.014 [+0.008, +0.019], GAT +0.019 [+0.013, +0.024] (closed-set). That bound is an order of magnitude below the family gap of +0.172 it would have to explain away.

S5.7 Ladder consistency assertions

Table S19: Assertions checked on the completed ladder. n as stated per assertion; analysis unit = run or fold–seed–protocol cell. All pass.

Assertion	Observed
every cell present exactly once	1,050 cells, maximum multiplicity 1
non-graph methods identical under R0 and R1	max $\|\Delta\| = 0$ over 100 runs $\times$ 5 metrics
non-graph methods identical under B0 and B1	max $\|\Delta\| = 0$ over 100 runs $\times$ 5 metrics
R1: cross-role edges = 0, 3-hop pairs = 0	max 0, max 0
B1, B2: cross-role edges = 0, 3-hop pairs = 0	max 0, max 0
R0 unpruned graph does carry contamination	15,970–16,277 3-hop pairs; 6,469–6,635 edges
R2 unpruned graph does carry contamination	2,899–3,556 3-hop pairs; 7,014–7,400 edges
R3 unpruned graph does carry contamination	1,355–6,284 3-hop pairs; 2,745–4,701 edges
B0 unpruned graph does carry contamination	287–1,027 3-hop pairs; 213–360 edges
R2, R3 match block train/val/test/excluded counts	all 50 fold $\times$ seed $\times$ protocol cells match
R3 matches block per-class counts	all 11 classes $\times$ 3 roles $\times$ 25 cells match
B0/B1/B2 reproduce block-fold counts	all folds match
B2 fold-local features are not the global features	max $\|\Delta\| = 64.898$

S6 Fold-local persistent homology: audit and attribution

S6.1 What was contaminated

The shipped topological features were computed once, with $r = 3$ hops on the shared $k = 6$ graph over all 3,798 spots, and reused unchanged in every fold. The filtration is a Vietoris–Rips complex over $1 - \rho(x_i, x_j)$ on the 50-dimensional PCA embedding of the r-hop ball; H_0 and H_1 diagrams are each vectorised as a 20×20 persistence image, giving 800 features per spot.

Table S20: Persistent-homology neighbourhood contamination, both directions, before and after the role-local rule. Exhaustive counts of spots whose $r = 3$ neighbourhood contains at least one node of the stated other role. $n = 5$ folds; analysis unit = fold. Both directions are asserted in code and the run aborts on failure; all assertions passed.

Fold	n test	Test spots seeing train (before)	%	After	n train	Train spots seeing test (before)	After
0	363	50	13.8	**0**	874	53	**0**
1	428	129	30.1	**0**	809	125	**0**
2	439	32	7.3	**0**	857	39	**0**
3	428	129	30.1	**0**	809	125	**0**
4	446	75	16.8	**0**	791	76	**0**

The other channels close too: validation spots seeing training nodes (75 in fold 2, 52 in fold 3) $\rightarrow$ 0; test spots seeing validation (5 in fold 2) $\rightarrow$ 0; test spots seeing buffer (34–184) $\rightarrow$ 0; training spots seeing buffer (60–181) $\rightarrow$ 0.

S6.2 Nothing degenerates

Table S21: Persistent-homology neighbourhood sizes before and after role-local restriction. "Degenerate" means fewer than 3 spots, the threshold below which a persistence diagram is undefined. n = 8,480 train/validation/test spot-folds across 5 folds and 3 roles; analysis unit = spot within fold and role.

Fold	Mean train (before → after)	Mean val	Mean test	Min size after (tr/val/te)	Degenerate after
0	36.35 → 34.93	36.14 → 33.45	35.70 → 33.49	15 / 14 / 13	0 / 0 / 0
1	35.93 → 33.52	36.65 → 33.16	36.58 → 33.50	13 / 12 / 12	0 / 0 / 0
2	36.36 → 33.48	36.13 → 33.55	36.79 → 32.35	12 / 14 / 13	0 / 0 / 0
3	35.93 → 33.52	36.16 → 32.60	36.58 → 33.50	13 / 12 / 12	0 / 0 / 0
4	36.18 → 34.54	36.65 → 33.16	36.13 → 33.55	12 / 12 / 14	0 / 0 / 0

0 of 8,480 spot-folds fall below the 3-point threshold, or even below 10 points. The smallest role-local neighbourhood in any fold is 12 spots and the mean falls only from 36.2 to 33.5. Persistent homology under strict fold isolation is viable on this dataset: the method does not depend on transductive access to remain computable. Its previously measured benefit did.

S6.3 How much the features moved

Table S22: Fraction of spots whose persistent-homology input set changed at all, with Jaccard similarity between old and new neighbourhoods. n = 8,480 spot-folds; analysis unit = spot within fold and role.

Fold	Train changed	Val changed	Test changed	Mean Jaccard (tr/val/te)	Min Jaccard
0	147 (16.8%)	121 (28.2%)	89 (24.5%)	0.961 / 0.926 / 0.938	0.378
1	215 (26.6%)	183 (38.1%)	154 (36.0%)	0.934 / 0.905 / 0.916	0.308
2	275 (32.1%)	126 (28.3%)	199 (45.3%)	0.921 / 0.930 / 0.879	0.308
3	215 (26.6%)	161 (40.2%)	154 (36.0%)	0.934 / 0.902 / 0.916	0.308
4	140 (17.7%)	183 (38.1%)	126 (28.3%)	0.956 / 0.905 / 0.930	0.308

Between 17% and 45% of spots per fold had their persistent-homology input set altered, and the worst-affected spots retain only about 31% of their old neighbours. Cosine similarity between old and new 800-dimensional vectors is a misleadingly reassuring statistic here — persistence images share a large common baseline, so cosine stays above 0.99 even when a vector moves substantially. The informative measure is relative L_2: **13–36% of test spots per fold** change by more than 10% and **8–24%** by more than 25%, under the neighbourhood change alone.

S6.4 The comparison and the attribution

Table S23: Arm means under protocol B2. All arms share model, optimiser, schedule, early stopping and seeds; only the input feature block differs. Runs are deterministic and reproduced bit-for-bit across two invocations. $n = 25$ runs per arm (5 folds $\times$ 5 seeds); analysis unit = fold–seed run; $\pm$ is 1 SD over the 25 runs, dominated by between-fold variation.

Arm	macro-F1 (default)	macro-F1, all 11 classes	macro-F1, shared classes
B5_gat_no_topo (matched GAT)	0.4077 ± 0.1962	0.1928 ± 0.0729	0.5024 ± 0.1496
TopoSpatial (fold-local PH)	0.3887 ± 0.1933	0.1829 ± 0.0643	0.4806 ± 0.1438
TopoSpatial_nbhdPH (diagnostic)	0.3934 ± 0.2017	0.1852 ± 0.0728	0.4829 ± 0.1601
TopoSpatial_oldPH (diagnostic)	0.4001 ± 0.2151	0.1863 ± 0.0714	0.4892 ± 0.1713

Table S24: Paired difference, TopoSpatial minus matched GAT. $n = 4$ independent spatial blocks (fold 3 dropped as a duplicate of fold 1); analysis unit = independent spatial block, paired within block. The exact two-sided sign-permutation test has a p-value floor of $2/2^4 = 0.125$; no entry in this table can reach $p < 0.05$ under this design.

Metric	Δ	SD	Exact sign-perm p	Per-block Δ (blocks 0, 1, 2, 3)
macro-F1 (default)	−0.0144	0.0269	0.500	+0.0043, −0.0540, +0.0004, −0.0084
macro-F1, all 11 classes	−0.0073	0.0148	0.750	+0.0012, −0.0295, −0.0012, +0.0001
macro-F1, shared classes	−0.0161	0.0327	0.750	+0.0043, −0.0648, −0.0042, +0.0003

Table S25: One-step-at-a-time attribution of the previously reported topology margin. Each row changes exactly one thing relative to the row above; every row shares the same graph, expression input, class weights and training schedule. $n = 4$ independent spatial blocks; analysis unit = independent spatial block, paired within block; 25 runs per arm.

Configuration of the PH block	Δ vs matched GAT	Shared classes	All 11 classes
Old shared-graph PH (as shipped)	+0.0017	−0.0031	−0.0021
Pruned-graph PH, global PCA, fixed ranges	−0.0125	−0.0178	−0.0066
Full B2 fold-local PH	−0.0144	−0.0161	−0.0073
Attributed to the leakage channel	**−0.0141**	−0.0148	−0.0045
Attributed to fold-local filtration + fitted ranges	−0.0019	+0.0017	−0.0008

The previously reported block-CV topology margin was +0.0154. The leakage channel alone accounts for +0.0141 of it. Within the resolution of this experiment the entire reported benefit of persistent homology under block cross-validation was the leak, and the remaining margin is negative and indistinguishable from zero.

One comparability caveat: the +0.0154 figure came from the fully transductive pipeline, whereas all four arms in Table S25 run under B2 for everything except the persistent-homology block. The decomposition is internally consistent and is the defensible basis for the attribution.

S6.5 Cost

Table S26: Runtime of the persistent-homology recompute. $n = 5$ folds; analysis unit = fold or whole study as stated. Single core except where noted.

Stage	Cost
Persistence diagrams, per fold (3,798 spots)	3.5–3.8 s
Vectorisation to 2×400 persistence images, per fold, $\times 2$ variants	2.6–2.8 s
Fold-local preprocessing (gene filter, HVG, scale, PCA), per fold	1.7–2.6 s
Neighbourhood audit (BFS on shared and pruned graphs, all roles), per fold	$\sim$1.6 s
Whole PH recompute, 5 folds, both variants, single core	**57.8 s**
100 GAT runs (4 arms $\times$ 5 folds $\times$ 5 seeds), one GPU	1.2 min

There is no computational argument for retaining the shared-graph shortcut.

S6.6 The inductive rule, and what it does not buy

Every spot's persistent-homology neighbourhood is the $r = 3$ ball around it computed inside the induced subgraph of its own partition role. Training spots see only training spots; validation spots only validation spots; test spots only test spots. Buffer spots are isolated and receive no persistent-homology vector at all. Because the pruned per-fold graph carries zero cross-role edges and gives every buffer node degree 0, breadth-first search on the pruned adjacency realises this rule exactly, and because the role subgraphs are disconnected the guarantee is not depth-limited.

Alternatives were considered and rejected. Building a test neighbourhood from the nearest *training* spots replaces a spatial construct with an arbitrary non-local one, since a held-out block contains no training spots. Reducing the neighbourhood to the spot itself is degenerate — persistent homology is undefined below 3 points, so every test vector would be zero, which deletes the feature rather than testing it. Including buffer spots as unlabelled context reintroduces train-to-test relay paths.

What the rule does *not* buy: a test spot's neighbourhood is built from other test spots, which is transductive within the held-out batch. It uses no labels and no training data, and it matches how a section is received in deployment, but a per-spot streaming deployment could not satisfy it. This is stated rather than glossed.

S6.7 Residual transductivity that no protocol here removes, and the mandated wording

Protocol B2 is stricter than the shipped pipeline on every fittable step: the gene detection filter, highly variable gene selection, scaling, PCA, persistence-image normalisation and the class weights are all fit on training spots only, and the message-passing graph carries zero cross-role edges. One thing it does not and cannot remove: **the 11 tissue-domain labels are Leiden clusters (resolution 0.8) computed once over all 3,798 spots on a combined expression-and-spatial graph. The prediction target itself is defined transductively.**

This is a property of the study design, not of the evaluation protocol, and it caps how "uncontaminated" any number in this work can be called. Accordingly, no arm in this paper is described as "leak-free" without qualification. The accurate phrase, used verbatim throughout, is *leak-free with respect to message passing, node features and preprocessing.* Any future use of these results should carry the same qualification.

S7 DLPFC: 12 sections and the hierarchical model

S7.1 Sections obtained

All 12 sections were obtained; none failed. Expression matrices came from the public S3 release (11 of 12 downloaded in this run; section 151673 resolved from a pre-existing local copy whose size matches the S3 object exactly, 12,887,164 bytes). Spatial positions and the barcode-level layer map came from the public repository of Maynard et al. [2021]. Every download attempt, including a URL pattern that returned HTTP 404 for all 12 sections before the working fallback, is recorded in the download manifest.

Table S27: Per-section provenance and quality control. Preprocessing is identical for every section: minimum 200 genes per spot, minimum 10 cells per gene, CPM-10^4 normalisation, $\log_{1p}$, 2,000 highly variable genes, PCA to 50 components. Moran's I uses the corrected formula of Section S9. $n = 12$ sections, 3 donors; analysis unit = section within donor.

Section	Donor	Spots	Layers	Imbalance	Moran's I (labels)	Moran's I (PC1)	Nbr. label agreement
151507	Br5292	4,213	7	3.97	0.9893	0.8040	0.924
151508	Br5292	4,375	7	6.91	0.9885	0.8157	0.927
151509	Br5292	4,783	7	11.34	0.9848	0.8076	0.927
151510	Br5292	4,587	7	10.02	0.9849	0.7946	0.926
151669	Br5595	3,634	**5**	9.30	0.9867	0.7607	0.955
151670	Br5595	3,480	**5**	10.40	0.9862	0.7282	0.954
151671	Br5595	4,089	**5**	7.82	0.9893	0.8253	0.953
151672	Br5595	3,885	**5**	5.20	0.9886	0.7881	0.951
151673	Br8100	3,608	7	4.54	0.9859	0.9206	0.904
151674	Br8100	3,635	7	4.12	0.9870	0.9158	0.903
151675	Br8100	3,563	7	2.81	0.9859	0.9143	0.900
151676	Br8100	3,428	7	3.29	0.9862	0.9111	0.903
Total	3 donors	**47,280**			0.985–0.989	0.728–0.921	0.900–0.955

Donor Br5595 contains no L1 or L2 spots at all in the released annotation. Its macro-F1 is computed over 5 classes rather than 7 and is therefore not on the same scale as the other eight sections; every analysis either handles this explicitly or is repeated on the 7-layer subset.

S7.2 Per-section dual protocol

Table S28: Per-section macro-F1 under three protocols, and the decomposition of the gap into a training-set-size part and a remainder. $n = 12$ sections, 3 donors, 10 runs per section–method–arm (5 folds × 2 seeds, fewer where runs were skipped); analysis unit = section within donor. **The remainder is labelled "spatial" but is not class-support-matched**: see the caveat below.

Method	Section	Random	Size-matched	Block	Total gap	From size	Remainder (%)
SVM-PCA	151507	0.6353	0.6249	0.1766	0.4587	0.0104	0.4483 (97.7)
SVM-PCA	151508	0.5976	0.5882	0.1514	0.4462	0.0094	0.4368 (97.9)
SVM-PCA	151509	0.6595	0.6497	0.1995	0.4600	0.0097	0.4502 (97.9)
SVM-PCA	151510	0.6011	0.5850	0.2110	0.3900	0.0161	0.3739 (95.9)
SVM-PCA	151669	0.6582	0.6462	0.2549	0.4034	0.0121	0.3913 (97.0)
SVM-PCA	151670	0.6410	0.6195	0.2419	0.3991	0.0215	0.3777 (94.6)
SVM-PCA	151671	0.6571	0.6425	0.2246	0.4325	0.0146	0.4179 (96.6)
SVM-PCA	151672	0.6568	0.6483	0.1873	0.4695	0.0085	0.4611 (98.2)
SVM-PCA	151673	0.6548	0.6451	0.2018	0.4531	0.0097	0.4433 (97.9)
SVM-PCA	151674	0.6888	0.6694	0.2031	0.4857	0.0194	0.4663 (96.0)
SVM-PCA	151675	0.6347	0.6185	0.2143	0.4204	0.0162	0.4042 (96.1)
SVM-PCA	151676	0.6296	0.6082	0.1675	0.4620	0.0214	0.4406 (95.4)
XGBoost-PCA+XY	151507	0.7890	0.7030	0.0610	0.7279	0.0860	0.6419 (88.2)
XGBoost-PCA+XY	151508	0.7793	0.7164	0.0562	0.7231	0.0629	0.6602 (91.3)
XGBoost-PCA+XY	151509	0.8209	0.7580	0.1283	0.6927	0.0629	0.6298 (90.9)
XGBoost-PCA+XY	151510	0.7753	0.6949	0.1597	0.6156	0.0803	0.5353 (87.0)
XGBoost-PCA+XY	151669	0.8987	0.8675	0.2537	0.6450	0.0312	0.6139 (95.2)
XGBoost-PCA+XY	151670	0.8447	0.8050	0.2329	0.6119	0.0398	0.5721 (93.5)
XGBoost-PCA+XY	151671	0.8631	0.8179	0.2974	0.5657	0.0452	0.5205 (92.0)
XGBoost-PCA+XY	151672	0.8757	0.8317	0.1385	0.7372	0.0440	0.6932 (94.0)
XGBoost-PCA+XY	151673	0.7972	0.7566	0.1606	0.6366	0.0406	0.5960 (93.6)
XGBoost-PCA+XY	151674	0.8125	0.7585	0.1675	0.6451	0.0540	0.5910 (91.6)
XGBoost-PCA+XY	151675	0.8178	0.7603	0.1551	0.6626	0.0575	0.6052 (91.3)
XGBoost-PCA+XY	151676	0.8040	0.7601	0.1999	0.6041	0.0439	0.5603 (92.7)

Scope restriction on the "% spatial" column. The size-matched arm matches **size only**. It does not match class support. The remainder column therefore **bundles the class-support effect into the "spatial" term**, and the breast-cancer decomposition (Section S5) shows that on that dataset class support is roughly three times the residual spatial term. The correct reading of Table S28 is: the gap is not a training-set-size artifact (size explains 2–8%). It is not: 87–98% of the gap is spatial extrapolation.

Skipped runs. 8 of 240 block-CV runs could not be fitted because after buffer removal the surviving training blocks contained only one class: section 151669 fold 0 seed 43; 151670 folds 0 and 1 seed 43; 151671 fold 0 seed 43 (each × 2 methods). All four are in donor Br5595. This is a real property of a coarse grid on laminar cortex — a 3×3 block can fall entirely inside one cortical layer — not a harness bug. All carry a skipped status and a reason string and are excluded from every mean.

Restricting to block folds where the training set does cover every test class still gives block macro-F1 of only 0.213 (SVM) and 0.178 (XGBoost), so missing classes are a contributing factor rather than the whole explanation on this dataset.

S7.3 Hierarchical model

Table S29: Linear mixed models on fold-level macro-F1, fitted with donor as group and a variance component for section nested within donor. All models converged. $n = 712$ fold-level observations from 12 sections and 3 donors ($n = 480$ from 8 sections and 2 donors for M4, $n = 356$ for the per-method models, $n = 472$ for M3b); analysis unit = fold-level run nested in section within donor.

Model	Term	Estimate	SE	95% CI
M1: macro_f1 ∼ protocol	protocol[block]	−0.5488	0.0099	[−0.5682, −0.5294]
	protocol[random_matched]	−0.0340	0.0098	[−0.0533, −0.0148]
M2: ∼ protocol + method	**protocol[block]**	**−0.5488**	**0.0089**	[−0.5661, −0.5314]
	protocol[random_matched]	−0.0340	0.0088	[−0.0513, −0.0168]
	method[XGBoost-PCA+XY]	+0.0961	0.0072	[+0.0819, +0.1102]
	Intercept (random CV, SVM-PCA)	0.6850	0.0210	[0.6439, 0.7261]
M3 (SVM-PCA only)	protocol[block]	−0.4409	0.0104	[−0.4612, −0.4205]
M3 (XGBoost-PCA+XY only)	protocol[block]	−0.6567	0.0110	[−0.6783, −0.6350]
M3b (reference = size-matched)	protocol[block]	−0.5146	0.0102	[−0.5346, −0.4946]
M4 (7-layer sections only)	protocol[block]	−0.5552	0.0097	[−0.5742, −0.5362]

Table S30: Variance components and intraclass correlations. $n = 712$ fold-level observations (356 for the per-method models); analysis unit = fold-level run nested in section within donor.

Model	Donor SD	Section|donor SD	Residual SD	ICC donor	ICC section|donor
M1 (pooled)	0.0335	0.0109	0.1076	0.087	0.009
M2 (+ method)	0.0335	0.0126	0.0962	0.107	0.015
M3 SVM-PCA	0.0315	0.0147	0.0798	0.131	0.029
M3 XGBoost-PCA+XY	0.0523	0.0148	0.0847	0.270	0.022

Table S31: Effect of ignoring the donor/section nesting. M5 is an ordinary least-squares fit with the same fixed effects and no random structure. $n = 712$ fold-level observations; analysis unit = fold-level run. The consequence of ignoring the nesting is quantity-specific.

Term	Mixed SE	Naive OLS SE	Ratio	Kind of quantity
Intercept	0.0210	0.0075	**2.79**×	between-donor level
protocol[block]	0.0089	0.0093	0.96×	within-section contrast
protocol[random_matched]	0.0088	0.0092	0.95×	within-section contrast
method[XGBoost-PCA+XY]	0.0072	0.0075	0.95×	within-section contrast

Ignoring the nesting understates the uncertainty on the *absolute performance level* by 2.8×, because that level varies by donor and there are only 3 donors. It does *not* inflate the standard error of the protocol contrast, because every section contributes both protocols so the donor effect cancels within the difference. Concretely: the −0.549 protocol effect is robust and its standard error is honest, but "macro-F1 ≈ 0.69 under random CV" carries the uncertainty of an $n = 3$ experiment, not an $n = 12$ one.

S7.4 Cross-section and cross-donor transfer

Per-section PCA bases are arbitrary up to rotation and sign, so transfer requires a shared embedding: 14,625 genes common to all 12 sections, 2,000 highly variable genes selected jointly with section as batch key, and one PCA fitted on the pooled 47,280 × 2,000 matrix. The `joint_pca` variant uses raw joint scores with no batch correction; the `joint_pca_zscore` variant z-scores each component within each section, which uses the held-out section's *features* (never its labels) and is therefore transductive in the features. This is disclosed and reported separately rather than folded into the headline. Training pools reach about 43,000 spots and are stratified-subsampled to 15,000 for the RBF SVM, with 2 subsample seeds.

Table S32: Leave-one-section-out and leave-one-donor-out transfer. $n = 24$ section–seed folds for LOSO (12 sections × 2 seeds) and $n = 6$ donor–seed folds for LODO (3 donors × 2 seeds); analysis unit = held-out section or held-out donor. 120 runs, 0 errors.

Design	Variant	Method	Mean	SD	Min	Max
Leave-one-section-out	joint_pca	SVM-PCA	**0.5676**	0.0386	0.5002	0.6171
Leave-one-section-out	joint_pca	XGBoost-PCA	0.5090	0.0331	0.4538	0.5585
Leave-one-section-out	joint_pca_zscore	SVM-PCA	0.5404	0.0302	0.4773	0.5870
Leave-one-section-out	joint_pca_zscore	XGBoost-PCA	0.4894	0.0328	0.4163	0.5387
Leave-one-donor-out	joint_pca	SVM-PCA	**0.4740**	0.0365	0.4252	0.5175
Leave-one-donor-out	joint_pca	XGBoost-PCA	0.4069	0.0281	0.3724	0.4382
Leave-one-donor-out	joint_pca_zscore	SVM-PCA	0.4179	0.0118	0.3997	0.4324
Leave-one-donor-out	joint_pca_zscore	XGBoost-PCA	0.3553	0.0409	0.3226	0.4132

Per-donor leave-one-donor-out means (`joint_pca`, SVM-PCA): Br5292 0.4728, Br5595 0.4343, Br8100 0.5149. Holding out a donor costs substantially more than holding out a section, because leave-one-section-out still trains on three sections of the same donor and is therefore partly a within-donor measurement. Leave-one-donor-out is the only design here that estimates generalisation to a new brain, and it rests on 3 independent units; its standard deviation is computed from 3 numbers and should not be read as a precise interval.

The per-section z-scoring made results worse everywhere (-0.020 to -0.056) and worse under leave-one-donor-out than leave-one-section-out. The naive per-section standardisation removes biologically real between-section variation along with batch effect. It is reported because it is the negative result; the uncorrected joint PCA is the headline.

S7.5 Method ranking reversal

Table S33: Method ranking by protocol. $n = 12$ sections (3 donors) for within-section protocols, 24 section–seed folds for LOSO, 6 donor–seed folds for LODO; analysis unit = section or held-out unit. "Folds won" counts folds in which XGBoost-PCA+XY beats SVM-PCA.

Evaluation	SVM-PCA	XGBoost-PCA+XY	Folds won by XGBoost
Within-section, random CV	0.6429	**0.8232**	**12 / 12**
Within-section, size-matched random	0.6288	**0.7692**	**12 / 12**
Within-section, block CV	**0.2015**	0.1646	2 / 12
Leave-one-section-out	**0.5676**	0.5090	1 / 24
Leave-one-donor-out	**0.4740**	0.4069	1 / 6

The mechanism is in the feature sets. XGBoost-PCA+XY receives raw pixel coordinates in addition to the 50 principal components; under random CV, with neighbouring spots sharing a layer label 90–95% of the time and split across folds, those coordinates are close to a lookup table of the answer. Under block CV they demand extrapolation, and across sections they are not comparable at all — the cross-section analyses drop coordinates entirely. Much of XGBoost-PCA+XY's random-CV advantage and its block-CV collapse trace to that single design choice.

Not comparable across partition designs. The 3×3 grid used here discards 47% of each section to buffers and yields block-CV macro-F1 of 0.16–0.20, a lower bound on achievable performance rather than an estimate of it. The finer partition used in the earlier single-section runs gives higher block-CV scores. These numbers are therefore *not* comparable to the previously reported DLPFC block-CV figures, which used a different partition — and which, in any case, were computed on mouse brain (Table S2).

S8 GraphST and STAGATE

S8.1 Setup

GraphST 1.1.1 from PyPI, `datatype='10X'`, 600 epochs, output dimension 64, defaults otherwise, on breast cancer Section 1 (3,798 spots, 2,000 highly variable genes from raw counts), with GraphST's own preprocessing. The representation is GraphST's 2,000-dimensional reconstruction reduced to 20 principal components, exactly GraphST's own downstream step, fed to XGBoost (100 trees, depth 5). GraphST's spatial graph is a rank-based 3-nearest-neighbour graph: 3,798 spots, 12,740 directed edges, mean degree 3.35, zero isolated nodes on the full slide, and mean degree 3.37–3.43 with zero isolated nodes on every training subgraph. Every graph was checked before training; the script aborts on a zero-edge graph and that branch never fired.

The experiment was run twice end to end and reproduced **bit-identically** (60 of 60 rows, maximum $|\Delta$ macro-F1$| = 0$).

Two variants: `GraphST+XGB` fits GraphST on the full slide, which is its native transductive design and mirrors the STAGATE treatment; `GraphST_trainonly+XGB` fits GraphST on the training subgraph only and applies the learned weights to the full graph to embed held-out spots, so no test-spot expression enters GraphST training or the PCA. The train-only variant is not published GraphST usage.

S8.2 Results

Table S34: Published spatial-embedding methods under both protocols. $n = 10$ runs per method–protocol cell (5 folds × 2 seeds); analysis unit = fold within one breast-cancer section. STAGATE was run on a different partition design (KMeans blocks, no buffer removal, ∼3,000 training spots) and was *not* rerun on the buffer-removal folds, so its comparison against the GraphST arms is directional, not matched; it also has no size-matched arm.

Method	Protocol	Mean ± SD	Min	Max
GraphST+XGB	random	0.926 ± 0.010	0.913	0.939
GraphST+XGB	size-matched random	0.905 ± 0.015	0.873	0.924
GraphST+XGB	block	0.583 ± 0.247	0.160	0.877
GraphST_trainonly+XGB	random	0.899 ± 0.011	0.880	0.912
GraphST_trainonly+XGB	size-matched random	0.812 ± 0.025	0.759	0.842
GraphST_trainonly+XGB	block	0.496 ± 0.236	0.138	0.865
STAGATE+XGB	random	0.807 ± 0.018	—	—
STAGATE+XGB	block	0.402 ± 0.139	—	—

Table S35: Inflation for the published methods. The analysis unit is the independent spatial partition, not the fold-seed run: folds 1 and 3 have byte-identical test sets (Jaccard 1.000, 428 test and 809 training spots each, every other pair disjoint), so the ten fold-seed observations are four independent spatial samples, each first averaged over its seeds. Intervals are a hierarchical bootstrap (10,000 replicates) that resamples partitions with replacement and then seeds within each drawn partition, preserving the pairing between protocols. Paired t and Wilcoxon statistics computed over fold-seed runs are omitted: they treat initialisation noise inside one held-out tissue as spatial replication and are not valid spatial inference. With four paired units an exact two-sided sign test cannot fall below $p = 0.125$, so effect size and interval are reported instead of significance. Percentages use the block-protocol mean as denominator.

Method	Comparison	Inflation	Paired Δ	95% CI
Four independent spatial partitions, matched fold definition				
GraphST+XGB	random vs block	+62.4%	+0.356 ± 0.294	[+0.123, +0.645]
GraphST+XGB	size-matched vs block	+58.7%	+0.335 ± 0.296	[+0.106, +0.622]
GraphST_trainonly+XGB	random vs block	+82.8%	+0.407 ± 0.292	[+0.139, +0.655]
GraphST_trainonly+XGB	size-matched vs block	+65.8%	+0.323 ± 0.285	[+0.067, +0.571]
Earlier fold definition, partition independence not verified				
STAGATE+XGB	random vs block	+100.7%	+0.405 ± 0.033	—

S8.3 What the GraphST gap is made of, and the scope restriction

Table S36: GraphST's own gap decomposition. Analysis unit = independent spatial partition ($n = 4$; folds 1 and 3 share a test partition and are collapsed), each component a mean of partition means so that the three components sum exactly to the total. **The residual is not class-support-matched** — see the caveat below.

Component	Δ macro-F1	Share
Training-set size (random → size-matched)	0.021	5.8%
Open-set / unseen classes (block → block closed-set)	0.027	7.6%
Remainder, labelled "spatial"	0.308	86.6%
Total	0.356	100%

Scope restriction. The GraphST size-matched arm matches **size only**. Its open-set correction removes only the contribution of test classes entirely absent from training; it does not match the class *sets* between the random and block arms. GraphST's block folds train on 7–9 of 11 domains and test on 3–6 — exactly the mismatch that the breast-cancer decomposition identifies as the largest component. The 86.6% figure therefore **bundles the class-support effect into the "spatial" residual** and must not be read as an estimate of spatial extrapolation. The defensible conclusion from this experiment is that published methods are subject to the same protocol-driven optimism as our own baselines, not that 86.6% of their gap is spatial.

Further disclosures.

1. Block folds 1 and 3 share an identical test set here as well; the script detects and logs this. Dropping fold 3 gives GraphST random 0.925 ± 0.010, block 0.557 ± 0.273, inflation +66.1% (size-matched +62.3%); the train-only variant gives +82.9% / +65.9%. The conclusion strengthens, but 4 blocks × 2 seeds is a small basis for a ± figure.
2. Block-fold macro-F1 is averaged over 3–6 classes and random-fold macro-F1 over 11. Fewer, larger classes generally makes macro-F1 easier, so this choice favours the block protocol and cannot manufacture the inflation.
3. For the train-only variant, inference runs on the full 3,798-spot graph, so two-hop message passing can still cross the buffer. That residual path is not closed by this experiment; the pruned graphs of Section S3 address it separately.
4. Two seeds, not five. Seed-to-seed spread is at most 0.03 within a protocol against a fold-to-fold spread of 0.16–0.88, so more seeds would not change the picture; more blocks would.

S8.4 GraphST's native task, reported separately

Scored on its own unsupervised full-slide task with no train/test split at all, GraphST attains ARI 0.6528 / NMI 0.7706 (seed 42) and ARI 0.6505 / NMI 0.7679 (seed 43), at 11 clusters. These are **not comparable** to the macro-F1 values above and are never merged with them. Two further notes: GraphST's default clusterer (mclust via rpy2) is not installed in this environment, so its supported Leiden fallback was used on the same 20-component representation, and published GraphST numbers generally use mclust and may differ; and the reference labels are themselves Leiden clusters of the same expression matrix, so an ARI of about 0.65 measures agreement between

two algorithmic partitions of one dataset, not recovery of expert-annotated anatomy. It should not be quoted as external validation.

S9 The Moran's I correction

The single-section implementation used in the earlier analysis computed

$$I = \frac{n}{\mathrm{nnz}(W)} \cdot \frac{z^\top W_{\mathrm{norm}} z}{z^\top z}, \tag{1}$$

where W_{norm} is *already* row-normalised. This applies the normalisation twice. Under row-normalised weights the total weight S_0 equals n, so the n/S_0 prefactor is exactly 1 and the correct expression is

$$I = \frac{z^\top W_{\mathrm{norm}} z}{z^\top z}. \tag{2}$$

The erroneous prefactor $n/\mathrm{nnz}(W) \approx 1/6.1$ deflated every reported value about six-fold.

Table S37: Published versus corrected Moran's I for DLPFC section 151673. $n = 1$ section for the published comparison and $n = 12$ sections for the corrected range; analysis unit = section. The binary-weight cross-check uses unnormalised adjacency and is an independent confirmation.

Quantity	Published	Corrected	Binary-weight cross-check
Labels (ordinal L1→WM coding), 151673	0.161	**0.986**	1.006
PC1, 151673	0.150	**0.921**	—
Labels, range across all 12 sections	—	0.985–0.989	—
Neighbour label agreement, all 12 sections	—	0.900–0.955	—

The published value of 0.161 was described in the text as "high", which it is not on the standard scale. The substantive conclusion is unchanged and in fact strengthened: cortical layer labels are almost perfectly spatially autocorrelated, which is exactly the condition under which random cross-validation leaks. The label statistic uses an ordinal L1→WM coding, which is meaningful for laminar cortex but coding-dependent; the coding-free neighbour-label-agreement statistic is reported alongside and tells the same story.

S10 Reproducibility

S10.1 Materials available for reproduction

The archived release accompanying this paper contains the complete set of materials needed to reproduce every number, table and figure reported here:

- the fixed spatial split definitions, including the block assignments and buffer membership used throughout, so that the partitions themselves need not be regenerated;
- the per-fold pruned graphs, together with the contamination audit that establishes zero cross-role edges and infinite minimum train–test hop distance;
- the complete per-run result tables for the eleven-arm protocol ladder, the fold-local persistent-homology comparison, the twelve DLPFC sections and the published-method runs, each row carrying its protocol, method, partition and seed;

- the analysis code that derives every reported component, interval and hierarchical model from those tables;
- the figure-generation code and, for each figure, a machine-readable table of exactly the values plotted.

Execution order, per-stage runtimes, software versions and file-level checksums are documented in the repository. Complete machine-readable results and exact execution instructions are provided in the archived repository.

S10.2 Determinism and verification

- The GraphST experiment was run twice end to end and reproduced bit-identically: 60 of 60 rows, maximum $|\Delta$ macro-F1$| = 0$.
- The topology arms are deterministic and reproduced bit-for-bit across two invocations.
- The pruned graphs are checked by two independently written verifiers, one against the pruning code and one against the split definition files; 45 of 45 checks pass.
- The protocol ladder carries 13 structural assertions (Table S19), all of which pass, including a bit-identity control for methods that never read the graph.
- Every figure ships with a sidecar CSV containing the exact plotted values, and the figure manifest records source artifacts, generating script, n, analysis unit and uncertainty type for each figure.
- A hard figure gate is run on both compiled documents and must exit 0. It checks that every graphics include resolves on disk, that the compile log reports no missing file, that the file-list records each figure as a compile-time input, that the output document contains at least one embedded form object per include, and that the rendered pages carry measurable ink. It exists because an earlier round shipped two documents in which every figure include had silently failed while the build reported no warnings.

S10.3 Random seeds

Model seeds are 42–46 for the protocol ladder and topology arms (5 seeds), 42–43 for the DLPFC and GraphST experiments (2 seeds). The hierarchical bootstrap uses seed 2026 with 10,000 replicates. Leiden label assignment used random state 0 at resolution 0.8 (breast cancer) and resolution 0.7 (mouse brain).

References

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

Kristen R Maynard, Leonardo Collado-Torres, Lukas M Weber, et al. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nature Neuroscience*, 24:425–436, 2021. doi: 10.1038/s41593-020-00787-0.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, et al. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.