ROWS ARE NOT EVIDENCE UNITS: EVIDENCE-UNIT INVARIANCE IN WEAK TEMPORAL LOCALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Weak temporal annotations often reach the learner as table rows, but row count can change without changing the underlying supervision. A one-to-many join, selective row replication, or metadata expansion may therefore change fitted preprocessing, the relative weight of weak statements, checkpoint selection, and predictions. We define *evidence-unit invariance*: alternative row materializations of the same evidence should define the same learning problem. We enforce this property with *quotient learning*, which compiles stored rows into canonical evidence classes before preprocessing, training, and validation. Across seven temporal datasets, representing every eligible range statement by five rows changes ordinary segment F1 on all seven, with absolute changes from 0.004 to 0.228. Quotient learning leaves the clean runs unchanged and gives zero response to the same materializations. The effect persists when preprocessing is fitted once per acquisition, and separate interventions show contributions from training-risk weighting and validation-based model selection. We also evaluate six common table transformations across three dataset–model configurations. Ordinary segment F1 changes in 12 of 18 combinations and held-out probabilities in 13. Row multiplicity can therefore redefine the effective learning problem even when the underlying evidence is unchanged.

1 INTRODUCTION

Weak temporal supervision often reaches a learning pipeline through a table. A row may encode a range, timestamp, bag, negative statement, or observed fragment. Table operations can increase the number of rows without creating a new annotation or signal. A one-to-many join, selective row replication, or metadata expansion can, for example, turn one weak statement into several stored rows.

Those extra rows matter when the pipeline treats stored rows as statistical units. An acquisition can enter fitted preprocessing multiple times, one weak statement can receive more weight than another in the training objective, and the same multiplicity can alter a validation objective used for checkpoint selection. The resulting model can change even though the underlying signals and weak statements are the same.

Storage identity and evidence identity can diverge. Two matching statements may have been generated independently and should carry separate statistical weight, while several rows with different storage metadata may originate from one statement. We formalize the desired behavior as *evidence-unit invariance*: changing the row representation of fixed evidence should leave the learning problem unchanged.

Our compiler identifies evidence classes using the independent group, acquisition, annotation source and type, temporal support, annotation content, and statement lineage. It also checks that rows assigned to the same class reference the same signal and validity mask. One canonical record is emitted for each evidence class before preprocessing, training, or validation.

If two tables differ only in row order or in the number of stored copies within the same evidence classes, compilation gives

$$\mathsf{Q}(\mathcal{D}) = \mathsf{Q}(\mathcal{D}').$$

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.

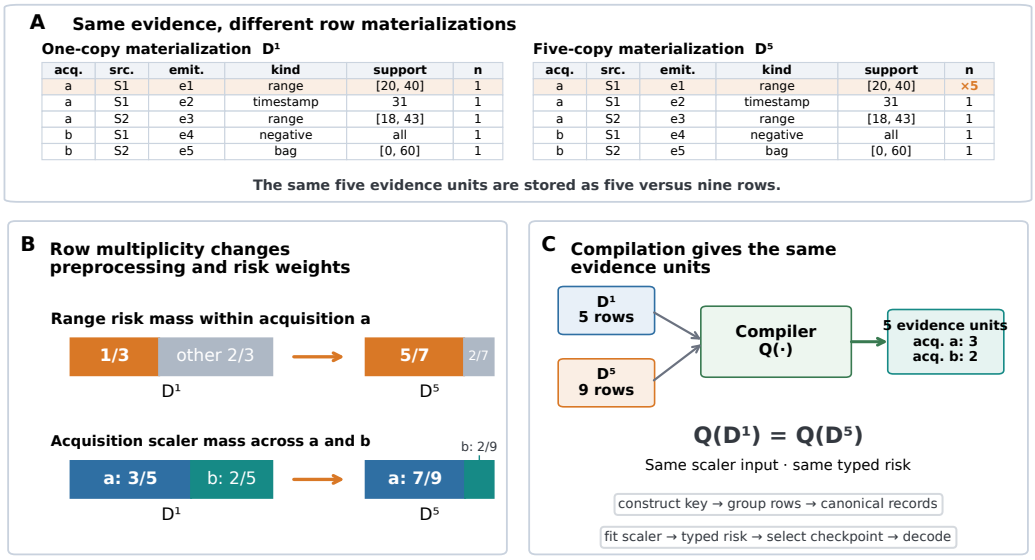

Figure 1: **Rows and evidence units. A:** Tables $\mathcal{D}^1$ and $\mathcal{D}^5$ contain the same five weak statements, but $\mathcal{D}^5$ stores one range statement five times. **B:** Ordinary row-wise learning changes that statement's within-acquisition risk mass from $1/3$ to $5/7$ and the scaler mass assigned to acquisition a from $3/5$ to $7/9$. **C:** Compilation maps both tables to the same five evidence units, so $\mathsf{Q}(\mathcal{D}^1) = \mathsf{Q}(\mathcal{D}^5)$. Preprocessing and typed risk construction therefore receive the same compiled input.

Quotient learning then applies the original preprocessing, architecture, typed weak losses, optimizer, validation objective, and decoder to the compiled records. Figure 1 shows how row multiplicity changes ordinary learning and how compilation removes that dependence.

The paper makes three contributions:

- We formalize evidence-unit invariance and implement it by compiling stored rows into canonical evidence classes before fitted preprocessing and learning.
- Across seven temporal datasets, we show that row materialization can change preprocessing, training weights, validation-based model selection, and held-out predictions while leaving the underlying evidence fixed.
- We evaluate six common table transformations across three dataset–model configurations and use statement-identity, architecture, and inverse-weighting comparisons to distinguish repeated storage from independently generated evidence.

2 Related Work

Weak supervision and temporal localization. Weak supervision combines information from labeling functions, annotators, or partial annotations and may model differences in their accuracy or dependence Ratner et al. (2016; 2017); Fu et al. (2020); Yu et al. (2022); Varma et al. (2019); Dawid & Skene (1979); Whitehill et al. (2009); Welinder et al. (2010). Multiple-instance and weak temporal methods define supervision through bags, sparse timestamps, points, ranges, video-level labels, or incomplete annotations Dietterich et al. (1997); Nguyen et al. (2018); Shou et al. (2018); Moltisanti et al. (2019); Li et al. (2021); Yin et al. (2023); Adams & Marlin (2017); Tsuboi et al. (2008); Cabannes et al. (2020). Recent theory extends this setting to weak signals generated jointly from several latent instance labels. Wang et al. formulate multi-instance partial-label learning and derive learnability conditions and generalization bounds Wang et al. (2023). Multiple-instance formulations have also moved directly into time-series learning. MILLET uses MIL for locally explainable time-series classification Early et al. (2024), while TimeMIL formulates multivariate

time-series classification as weakly supervised MIL with time-aware aggregation Chen et al. (2024). Recent temporal-localization work continues to study sparse point supervision Xia et al. (2024); Liu et al. (2025). These works define how weak signals constrain latent instance or temporal labels. Our question is which stored records carrying those signals receive independent statistical weight.

Repeated examples and implicit weighting. Training-data deduplication has been studied in connection with computational cost, memorization, privacy, and model behavior Schofield et al. (2017); Lee et al. (2022); Kandpal et al. (2022); Abbas et al. (2023). Duplicate prevalence can affect tabular learners differently across datasets and algorithms Mohammed et al. (2025). Repetition can also be deliberate: Fang et al. (2025) show that repeated exposure to filtered data and explicit changes to individual document counts can improve pretraining under fixed data or token budgets. Under additive empirical risk, multiplicity is a weighting mechanism. Statement identity determines whether that weight represents additional evidence.

Sets, bags, and relational learning. Deep Sets establishes permutation-invariant learning over sets Zaheer et al. (2017), while repeated elements remain meaningful under multiset semantics. Database theory similarly distinguishes set and bag semantics and uses lineage to describe how result records arise from source records Dayal et al. (1982); Buneman et al. (2001); Green et al. (2007); Cheney et al. (2009). Recent relational-learning systems encounter this multiplicity directly. TablePuppet studies learning over relational tables assembled through joins and unions and includes explicit handling of duplicate tuples introduced by joins Xu et al. (2026). Its learning procedures target the result of the relational query. Evidence-unit invariance determines which rows in such a materialized result represent distinct statistical evidence when several originate from one weak statement.

Data preparation and pipeline behavior. MLInspect, mlwhatif, and related work trace relational data transformations and measure their downstream effects Grafberger et al. (2021; 2023); Schelter et al. (2024). Datascope extends data-importance analysis across preprocessing and feature-extraction pipelines Karlaš et al. (2024). Metamorphic testing studies transformations under which system behavior should remain stable Segura et al. (2016); Murphy et al. (2008); Xie et al. (2011). Evidence-unit invariance gives such a condition for weak temporal supervision: joins, exports, metadata expansion, and row reordering that preserve the evidence units should preserve the compiled learning problem.

3 Evidence-Unit Invariance and Quotient Learning

3.1 Stored rows and row-wise risk

An acquisition a from independent group g contains a sequence $x_a = (x_{a1}, \ldots, x_{aT_a})$ and validity mask v_a. A stored weak row has the form

$$r = (g, a, s, \tau, \omega, x_a, v_a),$$

where s is the annotation source, τ the annotation type, and ω its typed temporal content.

Each row carries a lineage token $\lambda(r)$. Statement-level lineage is used when available. If the schema permits session identity to supply missing lineage, the session identifier provides that token instead. Rows without either form of lineage receive distinct row tokens. The full rule is given in Supplementary Section B.

Let D_a be the multiset of stored weak rows associated with acquisition a, and let $\ell_r(\theta)$ be the typed loss for row r. Ordinary learning assigns equal mass to independent groups, acquisitions within each group, and stored rows within each acquisition:

$$\mathcal{R}_{\mathrm{row}}(\theta; \mathcal{D}) = \frac{1}{|G|} \sum_{g \in G} \frac{1}{|A_g|} \sum_{a \in A_g} \frac{1}{|D_a|} \sum_{r \in D_a} \ell_r(\theta).$$

Copying one row preserves the total weight of its acquisition but redistributes that weight among its weak statements. The fitted scaler is similarly row-dependent: an acquisition contributes its valid bins once for every stored row that references it. The typed losses are given in Supplementary Section D.

3.2 COMPILATION INTO EVIDENCE CLASSES

Let $\sigma_\tau(\omega)$ denote the canonical temporal support of a statement and $\pi_\tau(\omega)$ its remaining annotation content. Define

$$\kappa(r) = \big(g, a, s, \tau, \sigma_\tau(\omega), \pi_\tau(\omega), \lambda(r)\big), \qquad \chi(r) = (x_a, v_a).$$

Here, $\kappa(r)$ specifies the evidence key and $\chi(r)$ the referenced observation. Two rows belong to the same evidence unit when

$$r \sim r' \quad \Longleftrightarrow \quad \kappa(r) = \kappa(r') \text{ and } \chi(r) = \chi(r').$$

Signal and validity-mask equality is elementwise and accounts for missing values. Independently generated statements remain separate through their lineage even when their annotation contents are numerically identical.

For evidence class C, let

$$q_C = (\kappa_C, \chi_C)$$

denote its canonical record. Compilation returns one record per evidence class:

$$\mathsf{Q}(\mathcal{D}) = \mathrm{sort}_\kappa\big\{q_C : C \in \mathcal{C}(\mathcal{D})\big\}.$$

Storage fields are discarded. Rows with the same complete key and different referenced observations are rejected.

Quotient learning applies the original preprocessing, architecture, typed losses, optimizer, validation objective, and decoder to $\mathsf{Q}(\mathcal{D})$. Algorithmic details and executable compiler tests appear in Supplementary Section B.

3.3 EVIDENCE-UNIT INVARIANCE

Proposition 1 (Evidence-unit invariance)**.** *Let $\mathcal{D}$ be a table in which each evidence class has a common referenced observation. Let A change the row order or the number of stored copies within existing evidence classes while preserving their complete keys and observations. Then*

$$\mathsf{Q}(A(\mathcal{D})) = \mathsf{Q}(\mathcal{D}).$$

Consequently, for deterministic fitted preprocessing P and typed risk $\mathcal{R}$,

$$\mathcal{R}(\theta; P(\mathsf{Q}(A(\mathcal{D})))) = \mathcal{R}(\theta; P(\mathsf{Q}(\mathcal{D}))),$$

with equal gradients wherever they exist.

Proof. The transformation preserves the evidence classes and their referenced observations, so it produces the same canonical records. Sorting those records gives identical compiler outputs, after which preprocessing and risk construction receive identical inputs. □

Applying Proposition 1 to the training and validation tables makes both objectives invariant to row materialization. Under matched initialization and deterministic execution, the optimization trajectory, selected checkpoint, fitted parameters, held-out probabilities, and decoded intervals are therefore identical.

For additive losses, retaining all stored rows with inverse within-class multiplicity gives the same objective as canonical compilation, $\mathcal{R}_{\mathrm{class}}(\theta; \mathcal{D}) = \mathcal{R}_{\mathrm{quot}}(\theta; \mathcal{D})$. Compilation is applied before preprocessing because row multiplicity can also enter scaler fitting, validation, batching, and other stages outside the additive training loss. Supplementary Section J gives the derivation and numerical comparison.

Supplementary Section I provides a matched-row comparison in which the numerical rows are identical but their statement identities differ. Five matching rows form one evidence class when they share statement identity and five evidence classes when they were generated independently.

4 EXPERIMENTAL DESIGN

Datasets and splits. We evaluate six public temporal datasets and one synthetic benchmark spanning smartphone activity recognition, freezing-of-gait detection, wearable activity recognition, and multimodal physiological monitoring Reyes-Ortiz et al. (2015); Banos et al. (2014); Roggen et al. (2010); Bächlin et al. (2010); Logacjov et al. (2021); Ustad et al. (2023); Reiss et al. (2019); Bach et al. (2022). Independent participants define the held groups for the public datasets, and independent generating units define them for the synthetic benchmark. Training, validation, and test groups are disjoint, and a separate model is fitted for each dataset. PPG-DaLiA was configured before its materialization response was examined and serves as the held-out dataset. Task definitions, group schedules, and evidence counts are given in Supplementary Section E.

Controlled materialization. Weak supervision consists of negative, positive-bag, timestamp, range, and observed-fragment statements. Each atomic statement receives statement identity before any joining, exporting, or metadata expansion. In the core intervention, every eligible range statement in the training and validation pools is represented by five rows sharing the same evidence identity. All other evidence units and held-out records are unchanged. Stored-row counts therefore increase by factors of $1.60\times$ to $3.00\times$ while the number of evidence units remains fixed. Separate interventions materialize the training and validation pools individually.

Models and evaluation. Ordinary learning processes rows as stored. Quotient learning first compiles them into evidence classes and then uses the same preprocessing, temporal convolutional network (TCN), typed losses, optimizer, validation objective, and decoder. We also evaluate a unique-acquisition scaler that fits preprocessing using each training acquisition once, and an approximately capacity-matched BiGRU.

Core training is full-batch over three seeds with validation-based checkpoint selection. Paired clean and materialized runs share the same split, initialization, optimization budget, losses, and decoder. Dense test labels are used for final evaluation.

The primary endpoint is segment F1 at an intersection-over-union threshold of 0.5. It is computed within each held-out acquisition, averaged across acquisitions and seeds within each held group, and then macro-averaged across groups. We also compare fitted probabilities and decoded intervals, including cases in which segment F1 is unchanged. Loss definitions appear in Supplementary Section D; interval matching, companion metrics, and optimization settings are given in Supplementary Section E.

Table materialization benchmark. We additionally apply six table transformations: a one-to-many metadata join, repeated-export concatenation, exploded list metadata, heterogeneous metadata fan-out, deterministic row reordering, and a mixed sequence. These transformations change the stored table while preserving the underlying weak statements, and every generated row inherits its parent statement's evidence identity. Row reordering changes only row order.

The benchmark evaluates the six transformations in three dataset–model configurations: HAPT–TCN, MHEALTH–TCN, and HAPT–BiGRU. Each configuration uses three held-group folds, and ordinary and quotient learning are compared throughout the fitted pipeline. Further details are given in Supplementary Section L.

5 RESULTS

5.1 CONTROLLED MATERIALIZATION ACROSS SEVEN DATASETS

The five-copy intervention increases stored-row counts by $1.60\times$ to $3.00\times$ while keeping the evidence units fixed.

Ordinary segment F1 changes on all seven datasets (Fig. 2A), ranging from $+0.004$ on HAR70+ to -0.228 on PPG-DaLiA. Quotient-learning responses are zero on all seven datasets. Figure 2B shows the held-group distributions underlying these dataset-level responses. Absolute F1 values and companion metrics are reported in Supplementary Section F.

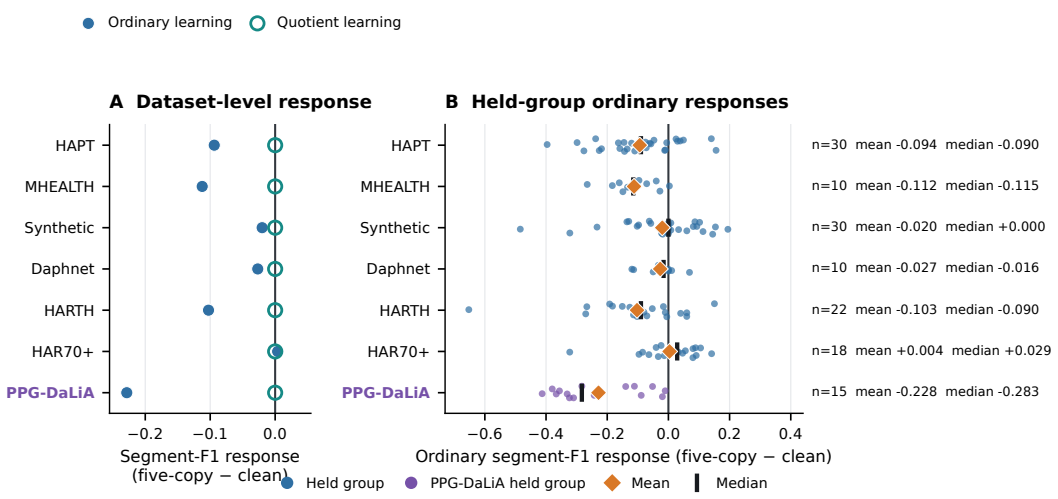

Figure 2: **Response to five-copy range materialization. A:** Dataset-level segment-F1 responses for ordinary and quotient learning. **B:** Ordinary responses across independent held groups after averaging acquisitions and seeds within group. Diamonds and black ticks denote group means and medians, and n is the number of held groups. PPG-DaLiA is highlighted in purple.

Aggregate segment F1 can remain unchanged while the fitted predictor changes. We therefore also compare fitted preprocessing, objectives, gradients, parameters, probabilities, and decoded outputs. Quotient learning gives identical fitted outputs across the corresponding clean and materialized conditions. Fixed-step and parameter-state comparisons are reported in Supplementary Section K.

5.2 Preprocessing, Training, and Validation Pathways

With the scaler fitted once per training acquisition, ordinary segment-F1 responses remain -0.129, -0.123, -0.039, and -0.229 for HAPT, MHEALTH, Synthetic, and PPG-DaLiA, respectively (Fig. 3A). Row weighting during training and validation therefore contributes independently of scaler fitting.

With unique-acquisition scaling fixed, training-only materialization changes segment F1 by -0.042, -0.054, and -0.152 for HAPT, MHEALTH, and PPG-DaLiA. Validation-only materialization changes segment F1 by -0.010, $+0.111$, and -0.004. These validation-only runs use identical training data, preprocessing, and training objectives; the changed validation objective can instead select a different checkpoint.

The prediction changes can be large even when segment F1 changes little (Fig. 3B). Validation-only materialization produces maximum absolute held-out probability changes of 0.641 on HAPT and 0.879 on PPG-DaLiA, while the corresponding segment-F1 changes are -0.010 and -0.004.

The approximately capacity-matched BiGRU is also sensitive to five-copy materialization. The same pattern appears when range, timestamp, bag, and negative statements are materialized at several multiplicities. Results are given in Supplementary Section H.

5.3 Common Table Transformations

We next apply six table transformations: a one-to-many metadata join, repeated export, exploded list metadata, heterogeneous metadata fan-out, deterministic row reordering, and a mixed sequence.

Across the six transformations and three dataset–model configurations, ordinary segment F1 changes in 12 of 18 combinations and held-out probabilities change in 13 (Fig. 4). Aggregate segment-F1 responses range from -0.138 to $+0.071$, and the largest absolute probability response is 0.587. Heterogeneous metadata fan-out produces some of the largest changes, whereas repeated export

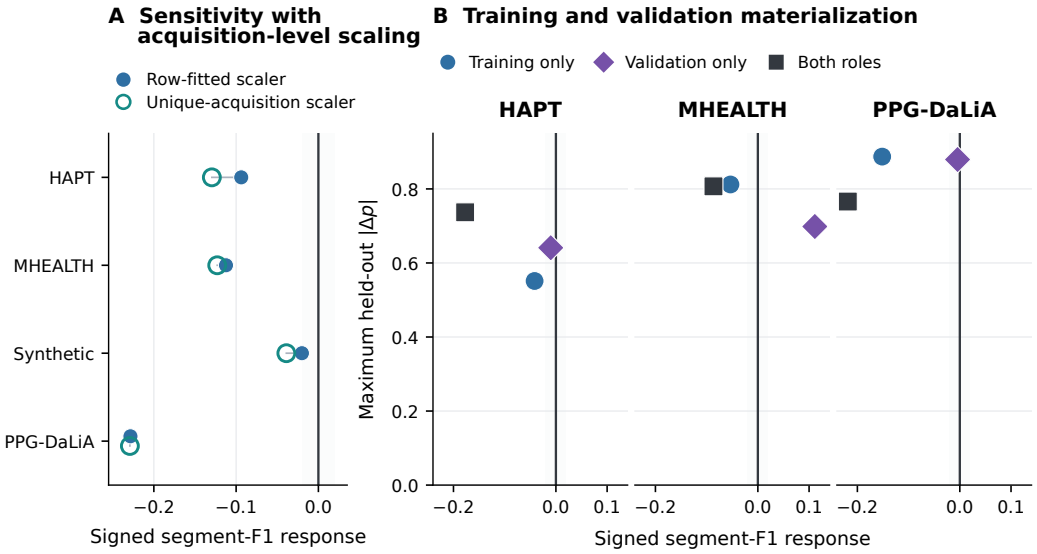

Figure 3: **Preprocessing, training, and validation pathways. A:** Ordinary segment-F1 responses under row-fitted and unique-acquisition scaling. **B:** Segment-F1 and maximum absolute held-out probability responses when training, validation, or both sets of statements are materialized.

and row reordering produce little or no change. Quotient-learning responses are zero across all 18 combinations.

The transformations can first appear at different points in the fitted pipeline, including scaler fitting, training and validation objectives, gradients, checkpoint selection, and decoded predictions. The complete stage-level results and numerical values are given in Supplementary Section L.

6 Discussion

Row count is part of the learning problem whenever a pipeline assigns statistical weight to stored rows. Our experiments show that ordinary table operations can therefore alter fitted preprocessing, training weights, checkpoint selection, and held-out predictions without changing the weak statements themselves. The response varies across datasets and transformations: some changes improve segment F1, others reduce it, and some substantially alter predicted probabilities while leaving segment F1 nearly unchanged.

The central distinction is between row identity and evidence identity. Repeated examples are already known to affect fitted models Schofield et al. (2017); Lee et al. (2022); Kandpal et al. (2022); Mohammed et al. (2025). Multiplicity can also carry intentional statistical weight: Fang et al. show that repeated exposure to selected data and explicit changes to individual-document counts can improve learning Fang et al. (2025). Relational learning systems also encounter multiplicity mechanically when joins produce duplicate tuples Xu et al. (2026). Statement identity supplies the information needed to assign empirical weight, determining whether repeated rows represent additional evidence. Independently generated statements can have identical numerical contents, while several stored rows can originate from one statement. The matched-row experiment in Supplementary Section I shows that this distinction changes the learning objective before optimization and can produce different fitted models.

Evidence identity must enter the pipeline before the first operation that depends on row multiplicity. Fitting the scaler once per acquisition removes the preprocessing contribution, while training and validation weighting still produce substantial responses. Likewise, inverse within-class weighting recovers the same additive objective as canonical compilation when preprocessing is already

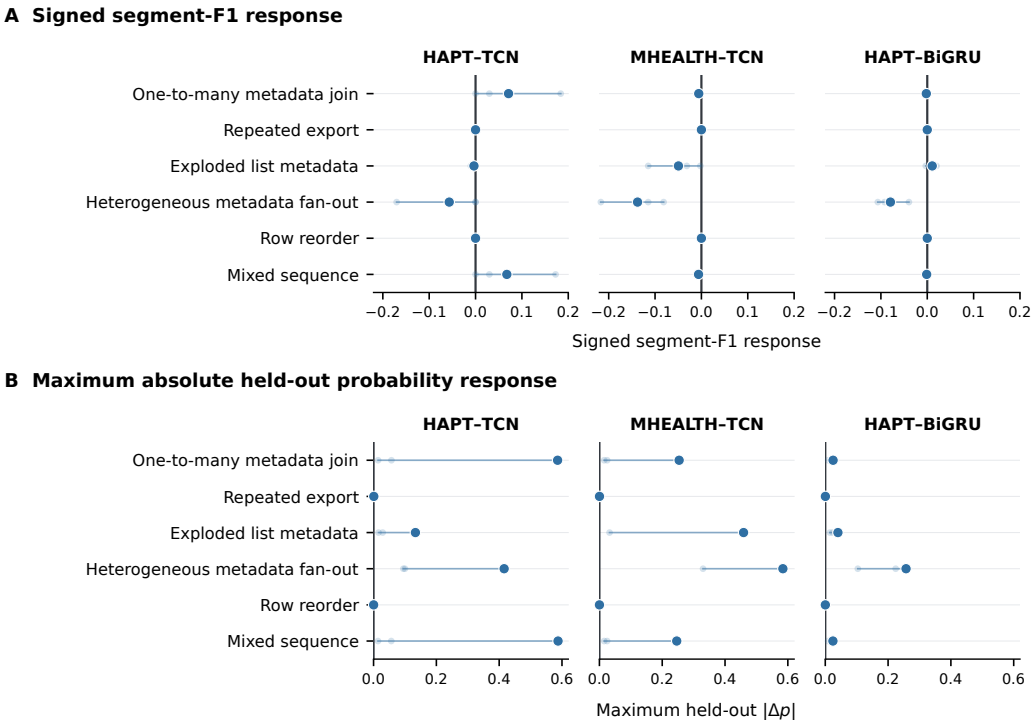

Figure 4: **Responses to six table transformations.** Columns show HAPT–TCN, MHEALTH–TCN, and HAPT–BiGRU; rows show the six transformations. **A:** Signed segment-F1 response. **B:** Maximum absolute held-out probability response. Pale points show fold-level values, dark points show aggregate responses, and horizontal segments span the fold-level range. Quotient-learning responses are zero.

multiplicity-invariant. Compilation provides the more general implementation because preprocessing, validation, batching, and other row-dependent stages all receive the canonical evidence classes.

The issue extends beyond the particular weak temporal tasks studied here. Structured annotations are often flattened into relational tables before they reach a learning algorithm. In such settings, statement identity is part of the statistical specification of the data. Joins, exports, metadata expansions, and row reordering also provide useful invariance tests: when they preserve the underlying evidence units, the compiled learner should remain unchanged Buneman et al. (2001); Green et al. (2007); Cheney et al. (2009); Segura et al. (2016); Murphy et al. (2008); Xie et al. (2011).

7 Limitations

Evidence-unit compilation depends on correctly represented statement identity. Missing statement identity can leave repeated materializations as separate evidence units, while an incorrect shared identity can combine independently generated statements. Canonical temporal supports, annotation content, and the referenced observations must therefore be represented consistently.

The experiments span seven temporal datasets built with a common weak-statement construction framework, and the prevalence of row-materialization effects in deployed systems remains unknown. The BiGRU results extend the finding to a second architecture within the same implementation. Exact numerical equality was measured under matched initialization and deterministic execution; small floating-point differences may arise in other numerical environments. Annotation quality, source disagreement, and annotator reliability are separate problems.

8 Conclusion

A table row and a unit of supervision are not necessarily the same thing. When one weak statement is materialized as several rows, ordinary learning can change through preprocessing, training weights, and validation-based model selection. Across seven temporal datasets and six common table transformations, these changes reach both evaluation metrics and held-out predictions.

Evidence-unit invariance makes the statistical unit explicit. Quotient learning enforces it by compiling stored rows into evidence classes before row-dependent stages of the pipeline. This ties statistical weight to evidence identity across alternative table representations.

Reproducibility Statement

The main paper specifies the evidence-class construction, controlled materializations, experimental design, and primary comparisons. The appendix provides the compiler algorithm and proof of Proposition 1, typed weak losses, dataset construction and held-group schedules, optimization and interval-matching details, complete numerical results, and the six table transformations (Sections B–L). An anonymized code and results package accompanies the submission with the compiler, executable tests, experiment configurations, machine-readable results, and figure and table generators.

AI Use Statement

Generative AI tools were used for manuscript editing, literature search, limited feedback on experimental design and theoretical presentation, and limited code-development assistance. AI-assisted suggestions were checked against the source literature, implementation, and experimental results. The authors reviewed the final analyses, claims, code, and manuscript and take responsibility for the content of this work.

References

Amro Kamal Mohamed Abbas, Kushal Tirumala, Daniel Simig, Surya Ganguli, and Ari S. Morcos. SemDeDup: Data-efficient learning at web-scale through semantic deduplication. In *ICLR 2023 Workshop on Multimodal Representation Learning: Perks and Pitfalls*, 2023. URL https://openreview.net/forum?id=u96ZBg_Shna.

Roy Adams and Benjamin Marlin. Learning time series detection models from temporally imprecise labels. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 157–165. PMLR, 2017.

Kerstin Bach, Atle Kongsvold, Hilde Bårdstu, Ellen Marie Bardal, Håkon S. Kjærnli, Sverre Herland, Aleksej Logacjov, and Paul Jarle Mork. A machine learning classifier for detection of physical activity types and postures during free-living. *Journal for the Measurement of Physical Behaviour*, 5(1):24–31, 2022. doi: 10.1123/jmpb.2021-0015. URL https://doi.org/10.1123/jmpb.2021-0015.

Marc Bächlin, Meir Plotnik, Daniel Roggen, Inbal Maidan, Jeffrey M. Hausdorff, Nir Giladi, and Gerhard Tröster. Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):436–446, 2010. doi: 10.1109/TITB.2009.2036165.

Oresti Banos, Rafael Garcia, and Alejandro Saez. Mhealth. UCI Machine Learning Repository, 2014.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2.

Kay H. Brodersen, Cheng Soon Ong, Klaas E. Stephan, and Joachim M. Buhmann. The balanced accuracy and its posterior distribution. In *20th International Conference on Pattern Recognition*, pp. 3121–3124, 2010.

Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Why and where: A characterization of data provenance. In *Database Theory—ICDT 2001*, volume 1973 of *Lecture Notes in Computer Science*, pp. 316–330. Springer, 2001. doi: 10.1007/3-540-44503-X_20.

Vivien Cabannes, Alessandro Rudi, and Francis Bach. Structured prediction with partial labelling through the infimum loss. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1230–1239. PMLR, 2020. URL https://proceedings.mlr.press/v119/cabannes20a.html.

Xiwen Chen, Peijie Qiu, Wenhui Zhu, Huayu Li, Hao Wang, Aristeidis Sotiras, Yalin Wang, and Abolfazl Razi. TimeMIL: Advancing multivariate time series classification via a time-aware multiple instance learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7190–7206. PMLR, 2024. URL https://proceedings.mlr.press/v235/chen24af.html.

James Cheney, Laura Chiticariu, and Wang-Chiew Tan. Provenance in databases: Why, how, and where. *Foundations and Trends in Databases*, 1(4):379–474, 2009. doi: 10.1561/1900000006.

A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979. doi: 10.2307/2346806.

Umeshwar Dayal, Nathan Goodman, and Randy H. Katz. An extended relational algebra with control over duplicate elimination. In *Proceedings of the First ACM SIGACT-SIGMOD Symposium on Principles of Database Systems*, pp. 117–123. ACM, 1982. doi: 10.1145/588111.588132.

Livia Del Gaudio, Vittorio Cuculo, and Rita Cucchiara. HyperMIL: Hypergraph-based channel reasoning for multiple instance learning on multivariate time series. In *Pattern Recognition. ICPR 2026*, volume 16814 of *Lecture Notes in Computer Science*, pp. 208–222. Springer, 2027. doi: 10.1007/978-3-032-31654-7_15. URL https://doi.org/10.1007/978-3-032-31654-7_15.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997. doi: 10.1016/S0004-3702(96)00034-3.

Joseph Early, Gavin K. C. Cheung, Kurt Cutajar, Hanting Xie, Jas Kandola, and Niall Twomey. Inherently interpretable time series classification via multiple instance learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=xriGRsoAza.

Alex Fang, Hadi Pouransari, Matt Jordan, Alexander Toshev, Vaishaal Shankar, Ludwig Schmidt, and Tom Gunter. Datasets, documents, and repetitions: The practicalities of unequal data quality. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0999. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2aba6ec20299931d46ddeefd5ddcb442-Abstract-Conference.html.

Daniel Fu, Mayee Chen, Frederic Sala, Sarah Hooper, Kayvon Fatahalian, and Christopher Ré. Fast and three-rious: Speeding up weak supervision with triplet methods. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3280–3291. PMLR, 2020. URL https://proceedings.mlr.press/v119/fu20a.html.

Stefan Grafberger, Shubha Guha, Julia Stoyanovich, and Sebastian Schelter. MLINSPECT: A data distribution debugger for machine learning pipelines. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 2736–2739, 2021. doi: 10.1145/3448016.3452759.

Stefan Grafberger, Shubha Guha, Paul Groth, and Sebastian Schelter. mlwhatif: What if you could stop re-implementing your machine learning pipeline analyses over and over? *Proceedings of the VLDB Endowment*, 16(12):4002–4005, 2023. doi: 10.14778/3611540.3611606.

Todd J. Green, Gregory Karvounarakis, and Val Tannen. Provenance semirings. In *Proceedings of the Twenty-Sixth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 31–40, 2007. doi: 10.1145/1265530.1265535.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330, 2017.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10697–10707. PMLR, 2022. URL https://proceedings.mlr.press/v162/kandpal22a.html.

Bojan Karlaš, David Dao, Matteo Interlandi, Sebastian Schelter, Wentao Wu, and Ce Zhang. Data debugging with shapley importance over machine learning pipelines. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=qxGXjWxabq.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 8424–8445, 2022.

Zhe Li, Yazan Abu Farha, and Juergen Gall. Temporal action segmentation from timestamp supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8365–8374, 2021.

Mengnan Liu, Le Wang, Sanping Zhou, Kun Xia, Xiaolong Sun, and Gang Hua. Boosting point-supervised temporal action localization through integrating query reformation and optimal transport. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13865–13875, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Liu_Boosting_Point-Supervised_Temporal_Action_Localization_through_Integrating_Query_Reformation_and_CVPR_2025_paper.html.

Aleksej Logacjov, Kerstin Bach, Atle Kongsvold, Hilde Bremseth Bårdstu, and Paul Jarle Mork. HARTH: A human activity recognition dataset for machine learning. *Sensors*, 21(23):7853, 2021. doi: 10.3390/s21237853. URL https://doi.org/10.3390/s21237853.

Sedir Mohammed, Lukas Budach, Moritz Feuerpfeil, Nina Ihde, Andrea Nathansen, Nele Noack, Hendrik Patzlaff, Felix Naumann, and Hazar Harmouch. The effects of data quality on machine learning performance on tabular data. *Information Systems*, 132:102549, 2025. doi: 10.1016/j.is.2025.102549. URL https://doi.org/10.1016/j.is.2025.102549.

Davide Moltisanti, Sanja Fidler, and Dima Damen. Action recognition from single timestamp supervision in untrimmed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9915–9924, 2019.

Christian Murphy, Gail E. Kaiser, Lifeng Hu, and Leon Wu. Properties of machine learning applications for use in metamorphic testing. In *Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering*, pp. 867–872, 2008.

Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6752–6761, 2018.

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: Rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11(3):269–282, 2017. doi: 10.14778/3157794.3157797.

Alexander J. Ratner, Christopher M. De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3567–3575, 2016.

Attila Reiss, Ina Indlekofer, Philip Schmidt, and Kristof Van Laerhoven. Deep PPG: Large-scale heart rate estimation with convolutional neural networks. *Sensors*, 19(14):3079, 2019. doi: 10.3390/s19143079.

Jorge-L. Reyes-Ortiz, Davide Anguita, Luca Oneto, and Xavier Parra. Smartphone-based recognition of human activities and postural transitions. UCI Machine Learning Repository, 2015. URL https://doi.org/10.24432/C54G7M.

Daniel Roggen, Meir Plotnik, and Jeffrey M. Hausdorff. Daphnet freezing of gait. UCI Machine Learning Repository, 2010. URL https://doi.org/10.24432/C56K78.

Sebastian Schelter, Shubha Guha, and Stefan Grafberger. Automated provenance-based screening of ML data preparation pipelines. *Datenbank-Spektrum*, 24(3):187–196, 2024. doi: 10.1007/s13222-024-00483-4.

Alexandra Schofield, Laure Thompson, and David Mimno. Quantifying the effects of text duplication on semantic models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2737–2747, 2017. doi: 10.18653/v1/D17-1290.

Sergio Segura, Gordon Fraser, Ana B. Sanchez, and Antonio Ruiz-Cortés. A survey on metamorphic testing. *IEEE Transactions on Software Engineering*, 42(9):805–824, 2016. doi: 10.1109/TSE.2016.2532875.

Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. AutoLoc: Weakly-supervised temporal action localization in untrimmed videos. In *Proceedings of the European Conference on Computer Vision*, pp. 162–179, 2018.

Yuta Tsuboi, Hisashi Kashima, Shinsuke Mori, Hiroki Oda, and Yuji Matsumoto. Training conditional random fields using incomplete annotations. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pp. 897–904, 2008. URL https://aclanthology.org/C08-1113/.

Astrid Ustad, Aleksej Logacjov, Stine Øverengen Trollebø, Pernille Thingstad, Beatrix Vereijken, Kerstin Bach, and Nina Skjæret Maroni. Validation of an activity type recognition model classifying daily physical behavior in older adults: The HAR70+ model. *Sensors*, 23(5):2368, 2023. doi: 10.3390/s23052368. URL https://doi.org/10.3390/s23052368.

Paroma Varma, Frederic Sala, Shiori Sagawa, Jason Fries, Daniel Fu, Saelig Khattar, Ashwini Ramamoorthy, Ke Xiao, Kayvon Fatahalian, James Priest, and Christopher Ré. Multi-resolution weak supervision for sequential data. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Kaifu Wang, Efthymia Tsamoura, and Dan Roth. On learning latent models with multi-instance weak supervision. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0424. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1e83498c3eafe109a44b12979c2c73db-Abstract-Conference.html.

Peter Welinder, Steve Branson, Pietro Perona, and Serge Belongie. The multidimensional wisdom of crowds. In John Lafferty, Christopher Williams, John Shawe-Taylor, Richard Zemel, and Aron Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/0f9cafd014db7a619ddb4276af0d692c-Paper.pdf.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Yoshua Bengio, Dale Schuurmans, John Lafferty, Christopher Williams, and Aron Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper_files/paper/2009/file/f899139df5e1059396431415e770c6dd-Paper.pdf.

Ziying Xia, Jian Cheng, Siyu Liu, Yongxiang Hu, Shiguang Wang, Yijie Zhang, and Liwan Dang. Realigning confidence with temporal saliency information for point-level weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18440–18450, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Xia_Realigning_Confidence_with_Temporal_Saliency_Information_for_Point-Level_Weakly-Supervised_Temporal_CVPR_2024_paper.html.

Xiaoyuan Xie, Joshua W. K. Ho, Christian Murphy, Gail E. Kaiser, Baowen Xu, and Tsong Yueh Chen. Testing and validating machine learning classifiers by metamorphic testing. *Journal of Systems and Software*, 84(4):544–558, 2011. doi: 10.1016/j.jss.2010.11.920.

Lijie Xu, Chulin Xie, Yiran Guo, Haiyang Lu, Gustavo Alonso, Bo Li, Guoliang Li, Wei Wang, Wentao Wu, and Ce Zhang. TablePuppet: Towards a generic framework for learning over relational tables. *The VLDB Journal*, 35(1), 2026. doi: 10.1007/s00778-025-00956-2. URL https://doi.org/10.1007/s00778-025-00956-2.

Yuan Yin, Yifei Huang, Ryosuke Furuta, and Yoichi Sato. Proposal-based temporal action localization with point-level supervision. In *Proceedings of the British Machine Vision Conference*, 2023. URL https://papers.bmvc2023.org/0820.pdf.

Peilin Yu, Tiffany Ding, and Stephen H. Bach. Learning from multiple noisy partial labelers. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 11072–11095, 2022. URL https://proceedings.mlr.press/v151/yu22c.html.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

APPENDIX

A EXTENDED RELATED WORK

Weak supervision and annotation sources. Programmatic weak supervision combines outputs from noisy labeling functions and may estimate their accuracies or dependencies Ratner et al. (2016; 2017); Fu et al. (2020); Yu et al. (2022); Varma et al. (2019). Crowd-label models likewise retain labels from different annotators as separate observations Dawid & Skene (1979); Whitehill et al. (2009); Welinder et al. (2010). In both settings, multiplicity corresponds to genuinely separate annotation statements. Our setting concerns a different source of multiplicity: one weak statement can appear in several stored rows after table operations. Evidence compilation keeps independent statements separate while mapping repeated materializations of the same statement to one evidence unit. Recent theory extends this setting to weak signals generated jointly from several latent instance labels. Wang et al. (2023) formulate multi-instance partial-label learning and derive learnability conditions and generalization bounds.

Weak temporal and multiple-instance learning. Multiple-instance learning defines the bag as the supervision unit Dietterich et al. (1997). Weak temporal methods extend this idea to video-level labels, sparse activations, ranges, boundaries, timestamps, points, negative statements, and incomplete temporal annotations Nguyen et al. (2018); Shou et al. (2018); Moltisanti et al. (2019); Li et al. (2021); Yin et al. (2023); Adams & Marlin (2017); Tsuboi et al. (2008); Cabannes et al. (2020); Varma et al. (2019). These methods specify how a weak statement constrains a latent temporal target. The question studied here is one step earlier: whether several stored records correspond to several weak statements or to several materializations of one statement.

Multiple-instance learning for time series. Recent work applies MIL directly to time-series data. MILLET treats a time series as a bag of temporal instances and uses instance-level contributions to produce local explanations for time-series classification Early et al. (2024). TimeMIL explicitly reformulates multivariate time-series classification as a weakly supervised MIL problem and incorporates temporal ordering and dependence into its aggregation Chen et al. (2024). HyperMIL further models high-order interactions among channels within a multivariate time-series MIL framework Del Gaudio et al. (2027). These methods determine how instances within a time series contribute to bag-level supervision. Evidence-unit invariance concerns how many statistical supervision units reach such a learner after the annotations have been represented in tables.

Duplicate data and implicit weighting. Repeated training examples have been studied in relation to computational efficiency, memorization, privacy leakage, and content redundancy Schofield et al. (2017); Lee et al. (2022); Kandpal et al. (2022); Abbas et al. (2023). Repetition can also change the effective weight assigned to an example during learning. In our setting, the annotation process determines that weight. Independently generated statements may have identical contents, while several rows derived from one statement may carry different auxiliary metadata. Evidence identity therefore requires information beyond content equality.

Sets, multisets, and database lineage. Deep Sets formalizes permutation-invariant learning, but repeated elements still change a multiset Zaheer et al. (2017). Database theory similarly distinguishes set and bag semantics and uses lineage to describe how output records were produced Dayal et al. (1982); Buneman et al. (2001); Green et al. (2007); Cheney et al. (2009). We apply this distinction before preprocessing and risk construction. Multiplicity is removed among rows derived from one declared statement and retained across independently generated statements, annotation sources, types, supports, annotation contents, and lineages.

Recent relational-learning work makes this multiplicity concrete. TablePuppet studies learning over joins and unions and includes explicit handling of duplicate tuples introduced by joins Xu et al. (2026). It preserves learning with respect to the materialized relational-query result. Evidence compilation determines the statistical units represented by that result when several tuples originate from one weak statement.

ML pipeline inspection and metamorphic testing. ML inspection systems track how data-preparation operations produce training records and how those operations affect downstream models Grafberger et al. (2021; 2023); Schelter et al. (2024). Metamorphic testing specifies transformations under which a system should preserve its behavior Segura et al. (2016); Murphy et al. (2008); Xie et al. (2011). Evidence-unit invariance applies this idea to weak temporal learning: changing row order or adding stored copies of an existing evidence unit should leave the compiled learning problem unchanged.

B Evidence Identity, Lineage, and Compilation

B.1 Statements, sessions, and materializations

We distinguish annotation sessions, annotation acts, atomic weak statements, and their stored materializations. One annotation act may emit several weak statements, each with its own identity. Table joins, exports, and metadata expansions can instead produce several rows from a single statement. The represented statement therefore determines the statistical unit used by the learner.

Numerically identical rows may come from independently generated statements and remain separate evidence units. Conversely, several rows derived from one statement may differ in storage metadata. The number of evidence units therefore depends on statement identity as well as numerical content.

B.2 Lineage, keys, and canonical records

In the controlled datasets, each atomic weak statement receives a unique `emission_id` before copying, joining, exporting, or other table materialization. Rows derived from the same statement inherit this token, while independently generated statements receive distinct tokens. The implementation uses `emission_id` as the lineage component of the complete evidence key.

Some schemas may lack statement-level lineage. Let $z_{\text{sess}} = 1$ when the schema defines one annotation session as one evidence unit, and let $z_{\text{sess}} = 0$ otherwise. For each stored row r,

$$\lambda(r) = \begin{cases} (\text{statement}, e_r), & e_r \neq \varnothing, \\ (\text{session}, q_r), & e_r = \varnothing,\ q_r \neq \varnothing,\ z_{\text{sess}} = 1, \\ (\text{row}, u_r), & \text{otherwise.} \end{cases}$$

Here, e_r is statement-level lineage, q_r is an annotation-session identifier, and u_r is a unique row token. Session identity contributes to evidence identity when the schema defines the session as one evidence unit. Otherwise, rows without statement-level lineage retain separate row identities.

The complete evidence-class key is

$$\kappa(r) = \big(g,\ a,\ s,\ \tau,\ \sigma_\tau(\omega),\ \pi_\tau(\omega),\ \lambda(r)\big),$$

where g is the independent group, a the acquisition, s the annotation source, τ the annotation type, $\sigma_\tau(\omega)$ the canonical temporal support, and $\pi_\tau(\omega)$ the remaining canonical annotation content. The referenced observation is represented separately as

$$\chi(r) = (x_a, v_a),$$

where x_a is the signal and v_a its validity mask.

For each evidence class C, let κ_C and χ_C denote its common key and observed payload. Its canonical record is

$$q_C = (\kappa_C, \chi_C).$$

When rows sharing a key also share the same observed payload, compilation returns

$$\mathsf{Q}(\mathcal{D}) = \text{sort}_\kappa\big\{q_C : C \in \mathcal{C}(\mathcal{D})\big\}.$$

Storage-row identifiers, filenames, paths, and other storage fields are discarded. If rows share a complete key but contain different signals or validity masks, compilation raises a conflicting-observation error.

Two rows belong to the same evidence unit when

$$\kappa(r) = \kappa(r') \quad \text{and} \quad \chi(r) = \chi(r').$$

Algorithm 1 Compilation of weak evidence

Require: Stored weak rows $\mathcal{D}$
Ensure: Canonically ordered evidence records $\mathsf{Q}(\mathcal{D})$
Initialize an empty map M
for each stored row $r \in \mathcal{D}$ **do**
  Resolve $\lambda(r)$ from statement lineage, declared session-level identity, or a unique row token
  Construct $\kappa(r) = (g, a, s, \tau, \sigma_\tau(\omega), \pi_\tau(\omega), \lambda(r))$
  Construct $\chi(r) = (x_a, v_a)$
  if $\kappa(r) \notin M$ **then**
    Store $q = (\kappa(r), \chi(r))$ in $M[\kappa(r)]$
  else if $\chi(r) \neq M[\kappa(r)].\mathrm{observed_payload}$ **then**
    raise conflicting-observation error
  else
    Associate r with the existing class $M[\kappa(r)]$
  end if
end for
return the values of M sorted by their complete keys

Evidence identity is therefore determined by the full key together with the referenced observation. Differences in group, acquisition, source, type, support, annotation content, or lineage define different evidence units. Storage-row identifiers, filenames, paths, and evaluation truth play no role in this identity.

Signal and validity-mask comparisons are elementwise and missingness-aware. Missing values agree when they occur at the same positions, and finite values must agree elementwise. Thus, rows satisfying

$$\kappa(r) = \kappa(r') \quad \text{and} \quad \chi(r) \neq \chi(r')$$

trigger a conflicting-observation error.

Canonical supports and annotation contents are defined by the source schema before evidence classes are constructed. Units, interval conventions, timestamps, categorical encodings, and numerical representations are normalized before compilation.

B.3 Compiler algorithm

For n stored rows and total inspected payload size B, hash-based compilation requires expected time $O(n + B)$ and storage linear in the number of distinct evidence classes.

B.4 Executable compiler tests

The executable suite contains 32 cases. Twenty span the five native annotation types at total row multiplicities 2, 3, 5, and 10. The remaining 12 exercise row order, evidence-key boundaries, missing values, conflicting observations, and statement- versus session-level identity. Across these cases, repeated materializations map to one evidence unit, independently generated statements remain separate, and conflicting observations raise an error.

The machine-readable results record each constructed input, expected and observed evidence-class counts, initial loss and gradient differences, error status where applicable, and test outcome.

C Proof and Implications of Proposition 1

Proof of Proposition 1. Let A change only row order or the number of stored copies within existing evidence classes while preserving their complete keys and observed payloads. The set of evidence-class keys is unchanged, and each key remains associated with the same observed payload. The canonical record q_C is therefore the same for every evidence class before and after the transformation. Sorting these records by their complete keys gives

$$\mathsf{Q}(A(\mathcal{D})) = \mathsf{Q}(\mathcal{D}).$$

A. Type-by-multiplicity tests (20 cases)

Annotation type	$\times 2$	$\times 3$	$\times 5$	$\times 10$
Negative	✓	✓	✓	✓
Positive bag	✓	✓	✓	✓
Timestamp	✓	✓	✓	✓
Range	✓	✓	✓	✓
Dense fragment	✓	✓	✓	✓

B. Additional compiler tests (12 cases)

Test	Setup	Result
Row reordering	Five annotation types	$5/5$ exact
Different evidence-key fields	Seven evidence units	$7/7$ retained
Shared lineage, different support	Distinct complete keys	Retained separately
Missing-aware equality	Matching key and observation	Combined correctly
Conflicting signal	Matching complete key	Error raised
Conflicting validity mask	Matching complete key	Error raised
Same session, independent statements	No statement lineage; session identity not used as the evidence unit	Retained separately
Session-level evidence	No statement lineage; session defined as one evidence unit	Combined correctly

Table 1: **Executable compiler tests.** Panel A covers the five annotation types at four row multiplicities. Panel B summarizes additional tests of evidence identity, missing values, session identity, and conflicting observations. Complete results for all 32 cases are included in machine-readable form.

Deterministic fitted preprocessing receives identical compiled inputs in the two conditions. The resulting risks are therefore identical functions of θ,

$$\mathcal{R}(\theta; P(\mathsf{Q}(A(\mathcal{D})))) = \mathcal{R}(\theta; P(\mathsf{Q}(\mathcal{D}))),$$

with equal gradients wherever the risk is differentiable.

Applying the argument separately to training and validation gives identical training and checkpoint-selection objectives. Under matched initialization and deterministic execution, the optimization trajectory, selected checkpoint, held-out probabilities, and decoded intervals are consequently identical. □

The proposition concerns transformations that preserve complete evidence keys and referenced observations. Independently generated statements retain distinct keys, while rows without statement-level lineage follow the row-identity or declared session-identity rules defined in Section B.

D Typed Weak Losses and Evidence Weighting

For acquisition a, let V_a denote its valid temporal bins and $p_t \in (0, 1)$ the model probability at bin t. Probabilities are clipped to a fixed numerical epsilon before evaluating logarithms.

For a negative statement,

$$\ell_{\text{neg}} = -\frac{1}{|V_a|} \sum_{t \in V_a} \log(1 - p_t).$$

For a positive bag with support B,

$$\ell_{\text{bag}}(B) = -\log \left[1 - \prod_{t \in B \cap V_a} (1 - p_t) \right].$$

This loss requires positive probability somewhere within the valid support without specifying its location.

A timestamp at location u with tolerance h defines

$$W(u,h) = [u-h, u+h] \cap V_a.$$

We use the dense local-window loss

$$\ell_{\text{time}} = -\frac{1}{|W(u,h)|} \sum_{t \in W(u,h)} \log p_t.$$

The timestamp generator requires at least one positive valid bin in $W(u,h)$, and the loss encourages high probability across the window. The same loss is used in all ordinary- and quotient-learning conditions.

For a positive range R with a nonempty valid complement, the loss adds an existential positive term inside the range and a negative mean outside it:

$$\ell_{\text{range}} = \ell_{\text{bag}}(R) - \frac{1}{|V_a \setminus R|} \sum_{t \in V_a \setminus R} \log(1-p_t).$$

When $V_a \setminus R$ is empty,

$$\ell_{\text{range}} = \ell_{\text{bag}}(R).$$

For an observed dense fragment with support F and binary labels y_t,

$$\ell_{\text{fragment}} = -\frac{1}{|F \cap V_a|} \sum_{t \in F \cap V_a} \left[y_t \log p_t + (1-y_t) \log(1-p_t) \right].$$

Statements with empty valid positive support are rejected before optimization. This includes empty bags, timestamp windows, positive ranges, and observed fragments.

Let D_a be the multiset of stored rows associated with acquisition a. Training is full-batch and assigns equal mass to independent groups, acquisitions within each group, and stored rows within each acquisition:

$$w_r = \frac{1}{|G|\,|A_g|\,|D_a|}, \qquad r \in D_a,$$

with

$$\mathcal{R}_{\text{row}}(\theta; D) = \sum_{g \in G} \sum_{a \in A_g} \sum_{r \in D_a} w_r \ell_r(\theta).$$

Validation uses the same normalization. Copying one row preserves the total weight of its acquisition but redistributes that weight among its weak statements.

Let $\mathcal{C}_a$ denote the evidence classes in acquisition a. Compilation emits one canonical record q_C for each class, giving

$$\mathcal{R}_{\text{quot}}(\theta; D) = \frac{1}{|G|} \sum_{g \in G} \frac{1}{|A_g|} \sum_{a \in A_g} \frac{1}{|\mathcal{C}_a|} \sum_{C \in \mathcal{C}_a} \ell_{q_C}(\theta).$$

Quotient learning therefore assigns risk weight to evidence classes while retaining the same typed weak losses.

Section J derives the equivalent inverse-within-class weighting formulation.

E Datasets, Splits, Models, and Evaluation

E.1 Dataset tasks and temporal representations

We evaluate six public continuous-time datasets and one synthetic benchmark. The public datasets span smartphone activity recognition, freezing-of-gait detection, wearable activity recognition, and multimodal physiological monitoring Reyes-Ortiz et al. (2015); Banos et al. (2014); Roggen et al. (2010); Bächlin et al. (2010); Logacjov et al. (2021); Ustad et al. (2023); Reiss et al. (2019); Bach et al. (2022). The synthetic benchmark provides known temporal intervals under controlled signal generation. A separate model is trained for each dataset.

Dataset	Positive target	Features and temporal resolution	Construction and evaluation
HAPT	Activity codes 7–12, corresponding to transition activities	Eight moment and dynamics summaries from six accelerometer and gyroscope channels; 48 features per nonoverlapping 0.5 s bin	Weak training uses 128-bin clips with stride 64; evaluation uses 61 complete experiments from held-out participants
MHEALTH	Official activity label > 0 versus label 0	Mean and standard deviation from 23 channels; 46 features per 2 s bin	Weak training uses 128-bin positive clips and deterministic negative clips; evaluation includes 142 acquisitions from ten held-out participants
Synthetic	Generated binary interval state	Six generated channels per 1 s bin	Known continuous reference intervals; 206 acquisitions across 30 independent held groups
Daphnet	Label 2, freezing, versus label 1 within experimental labels 1 and 2	Mean and standard deviation from nine inertial channels; 18 features per 1 s bin	Complete recordings are evaluated; bins containing fewer than 32 raw samples are marked invalid
HARTH/ HAR70+	Labels 1, 3, 4, and 5, representing locomotor activity	Mean and standard deviation from six back and thigh accelerometer channels; 12 features per 1 s bin	Weak training uses fixed 120 s windows containing at least 60 valid seconds; complete held-out participant records are scored
PPG-DaLiA	Activity ID 2, stairs, or ID 7, walking	Twelve fixed multimodal summaries per 5 s bin	Complete held-out participant records are evaluated; the dataset was configured before its materialization response was examined

Table 2: **Dataset task definitions.** A separate model is fitted for each dataset. Metrics are first computed within acquisitions and then aggregated within each held group. The held group is the independent evaluation unit.

Independent participants define the evaluation groups for the public datasets, and independent generating units define them for the synthetic benchmark. PPG-DaLiA was configured before its materialization response was examined and serves as the held-out dataset.

HAPT uses 0.5 s bins. Temporal errors are converted from bins to seconds by multiplying onset, offset, and duration values by 0.5. False-positive interval counts are converted to hourly rates using the corresponding factor of two.

Compilation removes 804, 300, 304, 276, 2,104, 1,296, and 120 additional materializations from the five-copy HAPT, MHEALTH, Synthetic, Daphnet, HARTH, HAR70+, and PPG-DaLiA tables, respectively. The corresponding tables contain 1,344, 284, 412, 286, 2,102, 754, and 60 evidence classes.

Across the seven clean datasets, all 5,242 stored rows map one-to-one to 5,242 evidence classes. Numerically matching statements remain separate when they have distinct statement identities.

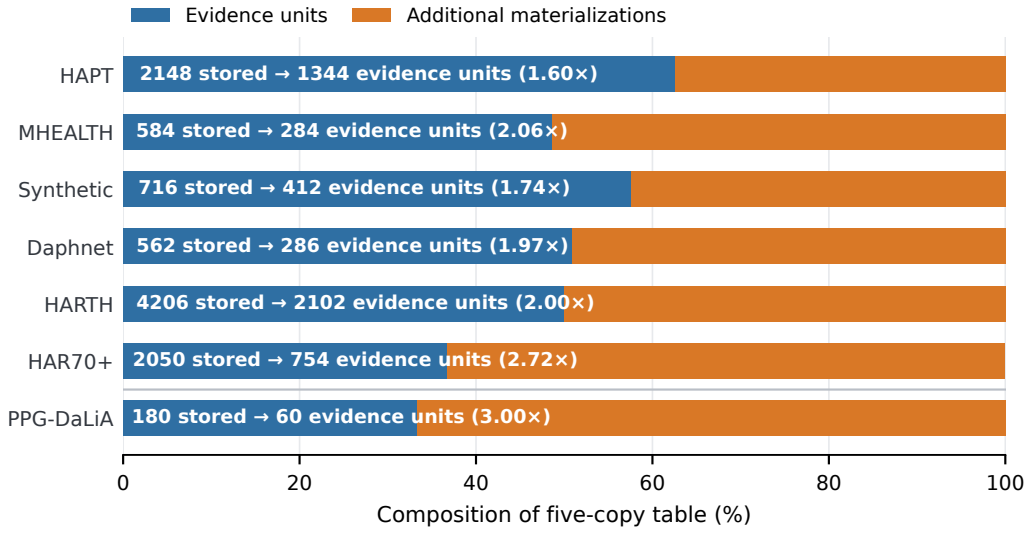

Figure 5: **Composition of the controlled five-copy tables.** Each clean row represents one evidence class. The five-copy intervention adds four materializations of every eligible range statement while keeping the number of evidence units fixed. Bars separate evidence units from the additional stored rows, and labels give the resulting row-to-evidence-unit ratio.

E.2 Weak-statement construction and controlled materialization

Weak statements are constructed from training labels as negative, positive-bag, timestamp, range, and observed-fragment annotations. Each atomic statement receives statement-level lineage before any joining, exporting, or metadata expansion. Independently generated statements retain distinct identities even when their numerical contents agree, while rows derived from the same statement inherit its identity. Section B gives the full lineage rules.

Each clean table contains one row per evidence class, so compilation leaves it unchanged. In the core intervention, every eligible range statement in the training and validation pools is represented by five stored rows: the original row and four additional materializations with the same evidence identity. All other evidence units remain unchanged. Thus,

$$|\mathcal{D}^{(5)}| = |\mathcal{D}^{(1)}| + 4N_{\text{range}},$$

where N_{range} is the number of eligible range statements in the clean table.

Held-out evaluation sequences and reference intervals are identical across the clean and five-copy conditions. The core experiment materializes both training and validation statements. Separate experiments materialize the two roles individually to isolate effects arising during training and model selection.

E.3 Held-group schedules

Training, validation, and test groups are disjoint in every schedule. Each acquisition belongs to one independent group and one role, and evidence classes never cross roles.

For datasets using five-fold schedules, the fold following the outer test fold under cyclic indexing is used for validation and the remaining folds for training. For subject-wise schedules, training, validation, and test participants are specified directly.

PPG-DaLiA was selected and configured before its materialization response was examined and is used as the held-out dataset.

Dataset	Groups	Schedules	Clean evidence units	Eligible range statements	Five-copy rows	Rows / evidence unit
HAPT	30	5	1344	201	2148	1.60×
MHEALTH	10	5	284	75	584	2.06×
Synthetic	30	5	412	76	716	1.74×
Daphnet	10	5	286	69	562	1.97×
HARTH	22	22	2102	526	4206	2.00×
HAR70+	18	18	754	324	2050	2.72×
PPG-DaLiA	15	15	60	30	180	3.00×

Table 3: **Dataset sizes and five-copy materialization.** Each clean row represents one evidence unit. The five-copy intervention adds four stored rows for every eligible range statement while keeping the number of evidence units fixed. Group counts define the evaluation denominator.

E.4 Models and optimization

The primary model is a fixed-capacity temporal convolutional network (TCN). Ordinary learning processes weak rows as stored. Quotient learning first compiles them into evidence units and then uses the same preprocessing, TCN, typed losses, optimizer, validation objective, and temporal decoder.

The unique-acquisition scaler fits the global preprocessor using each training acquisition once while leaving the ordinary row-wise training and validation objectives unchanged. This separates the contribution of row-weighted preprocessing from the effects of statement weighting during training and validation.

We also use an approximately capacity-matched bidirectional gated recurrent unit model (BiGRU). The TCN and BiGRU contain 11,617 and 11,953 trainable parameters for HAPT, 11,377 and 11,665 for MHEALTH, and 6,577 and 5,905 for Synthetic, respectively. Layer dimensions and preprocessing details are provided in the repository.

Training is full-batch. Core fits use a learning rate of 0.01, at most 25 epochs, patience 7, and seeds 1103, 2207, and 3301. Checkpoints are selected using the validation objective. The decoder uses a probability threshold of 0.5, minimum duration of one bin, and merge gap zero.

Paired clean and materialized runs share the same architecture, data split, seed, initialization, optimizer, training budget, losses, and decoder. Materialization can change the validation trajectory, selected checkpoint, and stopping time.

E.5 Evaluation metrics and interval matching

The primary endpoint is segment F1 at an intersection-over-union threshold of 0.5. Metrics are computed separately within each held-out acquisition, averaged across acquisitions and seeds within each held group, and then macro-averaged across groups. Segment matching is performed separately within each acquisition.

Companion endpoints are frame-level balanced accuracy Brodersen et al. (2010), Brier score Brier (1950), 10-bin expected calibration error Guo et al. (2017), onset error, duration error, and false-positive intervals per non-event hour. Probability metrics use valid held-out bins. Onset and duration errors are computed from matched prediction–reference intervals.

Within each acquisition, prediction–reference pairs with IoU below 0.5 are excluded. Among the remaining pairs, matching first maximizes the number of matches and then uses total IoU to break ties. The corresponding empty-set, unmatched-event, and temporal-error conventions are given in Section K.2.

Alongside endpoint metrics, we compare fitted preprocessing, initial objectives, gradients, selected checkpoints, parameters, probabilities, and decoded outputs.

Dataset	Clean F1	Five-copy F1	ΔF1
HAPT	0.831	0.737	−0.094
MHEALTH	0.264	0.152	−0.112
Synthetic	0.263	0.243	−0.020
Daphnet	0.039	0.013	−0.027
HARTH	0.347	0.245	−0.103
HAR70+	0.479	0.482	+0.004
PPG-DaLiA	0.368	0.139	−0.228

Table 4: **Ordinary segment-F1 response to five-copy range materialization.** Segment F1 is computed within each acquisition at IoU 0.5, averaged across acquisitions and seeds within each held group, and then macro-averaged across groups. ΔF1 is five-copy minus clean. Quotient-learning responses are zero and are omitted.

Dataset	ΔBA	ΔBrier	ΔECE	Δonset (s)	Δduration (s)	ΔFP/h
HAPT	−0.0605	+0.00575	+0.0072	−0.174	−0.551	+1.71
MHEALTH	−0.0356	+0.0035	+0.0074	−0.955	−6.33	−19.3
Synthetic	−0.0109	−0.00475	−0.0123	+0.11	−2.2	−10.8
Daphnet	−0.0845	−0.0771	−0.112	−1.55	−1.49	−35.4
HARTH	−0.115	+0.0121	+0.0208	+0.736	+0.938	−1.34
HAR70+	−0.0188	+0.00714	+0.0119	−0.86	−1.38	−4.74
PPG-DaLiA	−0.134	−0.0136	−0.0193	+15.6	+26.8	+1.21

Table 5: **Companion metric responses under five-copy materialization.** Values are ordinary five-copy minus clean. Lower values are preferable for Brier score, ECE, temporal errors, and false-positive intervals per hour. Quotient-learning responses are zero and are omitted.

F Full Core and Companion Results

The clean tables contain one stored row per evidence class, so ordinary and quotient learning begin from the same compiled data, preprocessing input, training objective, and validation objective.

The five-copy intervention increases stored-row counts by factors from $1.60\times$ to $3.00\times$ while keeping the number of evidence units fixed. Representing every eligible range statement by five rows changes ordinary segment F1 on all seven datasets. Responses range from $+0.004$ on HAR70+ to -0.228 on PPG-DaLiA, with six of seven exceeding 0.02 in absolute value. Quotient learning gives zero response on every dataset.

The sign varies across datasets: HAR70+ changes by $+0.004$, while the other six dataset-level responses are negative.

G Transmission Pathways

G.1 Unique-acquisition scaling

In ordinary preprocessing, the fitted scaler receives one copy of an acquisition for every stored weak row that references it. The unique-acquisition variant instead fits the scaler using each training acquisition once, while retaining the ordinary row-wise training and validation objectives.

Segment-F1 responses remain -0.129 for HAPT, -0.123 for MHEALTH, -0.039 for Synthetic, and -0.229 for PPG-DaLiA. Row multiplicity therefore continues to affect learning after its contribution to scaler fitting is removed.

G.2 Training and validation materialization

With unique-acquisition scaling fixed, we materialize the training pool, validation pool, or both. Training-only materialization changes segment F1 by -0.042, -0.054, and -0.152 for HAPT, MHEALTH, and PPG-DaLiA, respectively.

Dataset	References	Clean predictions	Clean matches	Five-copy predictions	Five-copy matches	Δ matches
HAPT	1074	1390	1000	1305	886	-114
MHEALTH	816	1104	262	602	147	-115
Synthetic	390	1567	270	1285	243	-27
Daphnet	708	638	33	178	7	-26
HARTH	6312	3796	1636	3174	1237	-399
HAR70+	3777	2065	1377	1892	1377	0
PPG-DaLiA	90	396	78	491	39	-39

Table 6: **Segment counts across acquisition–seed evaluations.** Matching uses the IoU-qualified cardinality-first rule. Δ matches is ordinary five-copy minus clean. Quotient-learning counts match the clean condition and are omitted.

Dataset	Clean F1	Five-copy F1	ΔF1
HAPT	0.831	0.701	-0.129
MHEALTH	0.264	0.141	-0.123
Synthetic	0.263	0.224	-0.039
PPG-DaLiA	0.368	0.138	-0.229

Table 7: **Ordinary response with unique-acquisition scaling.** The scaler is fitted using each training acquisition once. ΔF1 is five-copy minus clean.

Validation-only responses are -0.010, $+0.111$, and -0.004. In these runs, the training data, fitted scaler, and training objective are unchanged; the altered validation objective can instead lead to a different selected checkpoint and stopping time.

The prediction response can be much larger than the segment-F1 response. Validation-only materialization changes the maximum absolute held-out probability by 0.641 on HAPT and 0.879 on PPG-DaLiA while changing segment F1 by only -0.010 and -0.004, respectively.

H Architecture and Annotation-Type Breadth

The approximately capacity-matched BiGRU also responds to five-copy materialization, with segment-F1 changes of -0.003 on HAPT, -0.106 on MHEALTH, and -0.012 on Synthetic. Quotient learning gives zero response in all three cases.

We next materialize range, timestamp, positive-bag, and negative statements at total multiplicities 2, 5, and 10 on HAPT, MHEALTH, and Synthetic. Ordinary responses vary in sign, magnitude, and monotonicity across annotation types and datasets.

Two HAPT negative-statement fits are non-finite at multiplicities 5 and 10; these entries are reported as non-finite. Quotient-learning responses are zero for all finite paired conditions.

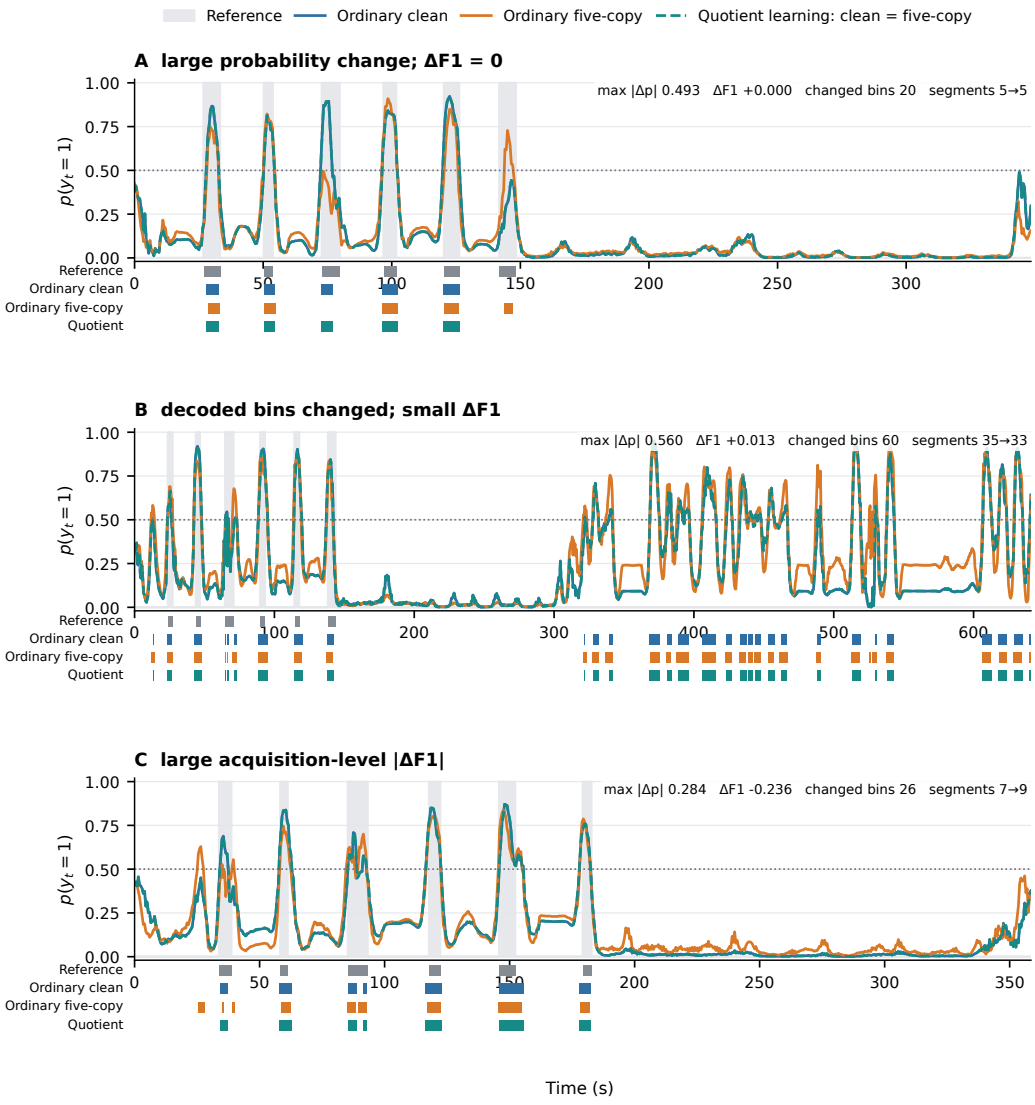

Figure 6: **Held-out examples of materialization sensitivity. A:** Large probability changes with unchanged acquisition-level segment F1. **B:** Changed decoded bins with a small segment-F1 response. **C:** A large acquisition-level segment-F1 response. Reference intervals and ordinary clean, ordinary five-copy, and quotient-learning probability trajectories are shown together with decoded intervals. Clean and five-copy quotient-learning outputs coincide. The dotted line marks the decoder threshold of 0.5.

Dataset	Materialized role	Selected epoch median [range]	Updates median [range]	ΔF1	$\max \lvert\Delta p\rvert$	Pairs
HAPT	Neither	23 [16, 25]	25 [23, 25]	$+0.000$	0.000	5
HAPT	Training only	23 [19, 25]	25 [25, 25]	-0.042	0.551	5
HAPT	Validation only	23 [13, 25]	25 [20, 25]	-0.010	0.641	5
HAPT	Both roles	23 [3, 25]	25 [10, 25]	-0.177	0.737	5
MHEALTH	Neither	4 [1, 7]	11 [8, 14]	$+0.000$	0.000	5
MHEALTH	Training only	5 [1, 12]	12 [8, 19]	-0.054	0.812	5
MHEALTH	Validation only	5 [2, 7]	12 [9, 14]	$+0.111$	0.698	5
MHEALTH	Both roles	5 [1, 13]	12 [8, 20]	-0.087	0.807	5
PPG-DaLiA	Neither	24 [11, 25]	25 [18, 25]	$+0.000$	0.000	15
PPG-DaLiA	Training only	9 [6, 25]	16 [13, 25]	-0.152	0.887	15
PPG-DaLiA	Validation only	8 [6, 25]	15 [13, 25]	-0.004	0.879	15
PPG-DaLiA	Both roles	16 [6, 25]	23 [13, 25]	-0.219	0.766	15

Table 8: **Training-versus-validation decomposition under unique-acquisition scaling.** Responses are relative to the condition in which neither role is materialized. The probability column reports the maximum absolute held-out response.

Dataset	Clean F1	Five-copy F1	ΔF1
HAPT	0.704	0.701	-0.003
MHEALTH	0.271	0.166	-0.106
Synthetic	0.231	0.219	-0.012

Table 9: **BiGRU response to five-copy materialization.** Values are ordinary five-copy minus clean. The approximately capacity-matched BiGRU uses the same evidence construction, splits, typed losses, optimization settings, and interval matching as the TCN. Quotient-learning responses are zero and are omitted.

Type	Rows	Mean ΔF1	Range across finite runs	Non-finite runs
Bag	2	$+0.065$	$[+0.003, +0.175]$	0
Bag	5	$+0.042$	$[+0.016, +0.080]$	0
Bag	10	$+0.024$	$[-0.010, +0.079]$	0
Negative	2	-0.004	$[-0.041, +0.032]$	0
Negative	5	-0.012	$[-0.013, -0.011]$	1
Negative	10	-0.041	$[-0.050, -0.031]$	1
Range	2	-0.010	$[-0.036, +0.005]$	0
Range	5	-0.037	$[-0.085, +0.012]$	0
Range	10	-0.071	$[-0.108, -0.014]$	0
Timestamp	2	$+0.015$	$[-0.063, +0.104]$	0
Timestamp	5	-0.011	$[-0.128, +0.104]$	0
Timestamp	10	-0.042	$[-0.195, +0.077]$	0

Table 10: **Ordinary responses across annotation types and row multiplicities.** Values summarize HAPT, MHEALTH, and Synthetic. Means and ranges are computed over finite runs. Quotient-learning responses are zero in all finite paired runs.

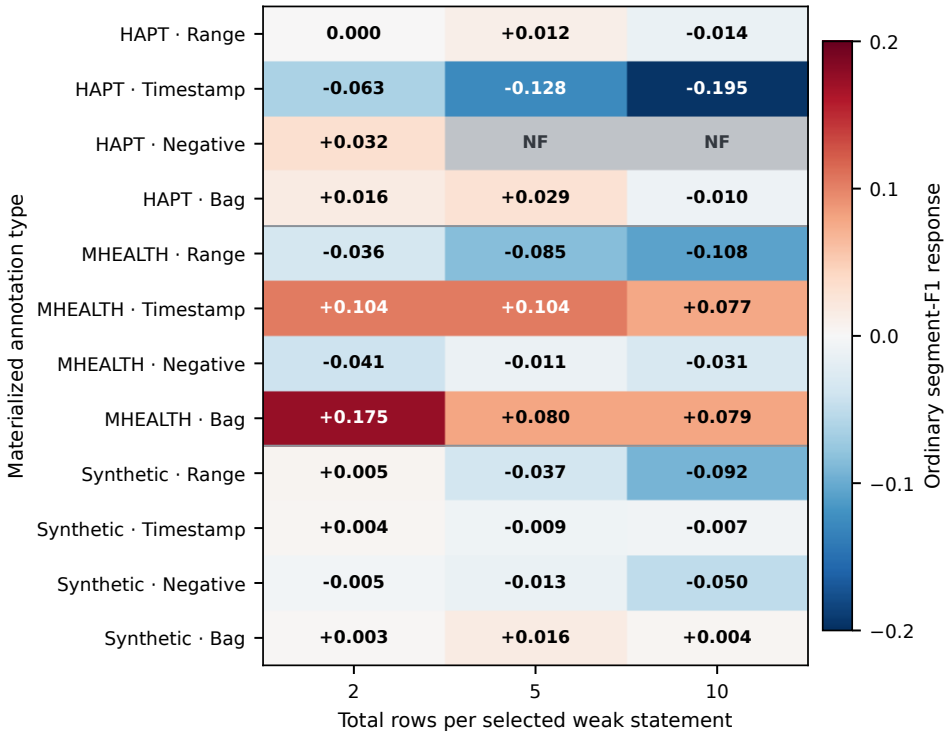

Figure 7: **Annotation-type-by-multiplicity stress matrix.** Cells show ordinary materialized-minus-clean segment-F1 responses for 2, 5, and 10 stored rows per selected weak statement. Responses vary in sign and magnitude across annotation types and datasets. Grey cells denote non-finite ordinary fits. Quotient-learning responses are zero in all finite paired conditions.

I Lineage Selectivity and Conflicting Observations

The shared-lineage and independent-lineage cases contain the same seven loss-bearing numerical rows. In the shared-lineage case, five matching rows belong to one evidence class. In the independent-lineage case, the same five rows carry distinct statement identities and remain five evidence classes.

Ordinary learning receives the same seven numerical rows in both cases and therefore produces the same initial objective, gradient, fitted parameters, selected checkpoint, update count, held-out probabilities, and decoded bins.

Compilation instead produces three evidence classes under shared lineage and seven under independent lineage. At the common initialization θ_0, the initial losses are 0.6818069 and 0.6608495, respectively, a difference of -0.0209574. This difference appears before optimization because one repeated statement and five independent statements receive different statistical weights.

After fitting, the two conditions select checkpoints 25 and 20 while both execute 25 updates. Their maximum absolute parameter difference is 0.2708, the parameter ℓ_2 difference is 4.568, and the maximum absolute held-out probability difference is 0.237. The decoded bins are unchanged in this example.

Rows sharing a complete evidence key but containing different signals or validity masks raise a conflicting-observation error. Rows that share lineage but differ in annotation type, support, or annotation content remain separate. Matching rows created within one annotation session also remain separate unless the schema defines that session as one evidence unit.

J Class-Normalized Weighting Equivalence

Let $\mathcal{C}_a$ denote the evidence classes in acquisition a, and let $m_C = |C|$ be the stored multiplicity of class C. Retaining all stored rows with inverse within-class multiplicity gives

$$\mathcal{R}_{\mathrm{class}}(\theta; D) = \frac{1}{|G|} \sum_{g \in G} \frac{1}{|A_g|} \sum_{a \in A_g} \frac{1}{|\mathcal{C}_a|} \sum_{C \in \mathcal{C}_a} \frac{1}{m_C} \sum_{r \in C} \ell_r(\theta).$$

All rows in evidence class C have the same typed loss, so

$$\frac{1}{m_C} \sum_{r \in C} \ell_r(\theta) = \ell_{q_C}(\theta),$$

and hence

$$\mathcal{R}_{\mathrm{class}}(\theta; D) = \mathcal{R}_{\mathrm{quot}}(\theta; D)$$

in real arithmetic.

The small numerical differences in loss, gradient, balanced accuracy, Brier score, and expected calibration error arise from floating-point accumulation order.

For the end-to-end comparison, both implementations fit the scaler once per training acquisition. Inverse class weighting inside the loss alone would leave row multiplicity in fitted preprocessing.

Method	Classes shared identity	Classes independent identity	Initial ΔL_0 independent − shared	$\max \lvert\Delta\theta^\star\rvert$	$\lVert\Delta\theta^\star\rVert_2$	$\max \lvert\Delta p^\star\rvert$
Ordinary learning	7	7	0	0	0	0
Quotient learning	3	7	-0.0209574	0.2708	4.568	0.2370

Table 11: **Effect of statement identity under matched numerical rows.** Both cases contain the same seven loss-bearing numerical rows. $\Delta L_0 = L_{\text{independent}}(\theta_0) - L_{\text{shared}}(\theta_0)$ is measured before optimization. Parameter and probability differences are measured at the selected checkpoints. The shared- and independent-identity quotient-learning runs select checkpoints 25 and 20, respectively.

Materialization	Rows	$\lvert\Delta L\rvert$	$\max \lvert\Delta\nabla\rvert$
All statements $\times 2$	8	5.96×10^{-8}	2.24×10^{-8}
All statements $\times 5$	20	0	3.73×10^{-8}
All statements $\times 10$	40	0	1.49×10^{-8}
One-to-many metadata join	10	0	7.45×10^{-9}
Repeated export	12	0	6.05×10^{-9}
Exploded list metadata	11	5.96×10^{-8}	7.45×10^{-9}
Heterogeneous metadata fan-out	12	0	6.05×10^{-9}
Mixed sequence	30	0	4.47×10^{-8}
Maximum	–	5.96×10^{-8}	4.47×10^{-8}

Table 12: **Canonical compilation and inverse-class weighting agree to floating-point precision.** Every condition contains four evidence classes. Clean input and deterministic row reordering give zero differences and are omitted.

Dataset	ΔF1	ΔBA	ΔBrier	ΔECE
HAPT	0	4.21×10^{-6}	-1.18×10^{-7}	2.09×10^{-7}
MHEALTH	0	0	-6.39×10^{-7}	3.04×10^{-6}
Synthetic	0	3.00×10^{-7}	-2.07×10^{-7}	-1.42×10^{-7}

Table 13: **Inverse-class weighting compared with canonical compilation.** Values are inverse-class weighting minus canonical compilation in the five-copy condition. Both implementations fit preprocessing once per training acquisition. Clean comparisons are zero and are omitted.

K Exact Invariance Comparisons and Interval Matching

K.1 Fixed-step and parameter-state comparisons

Across the 15 fixed-step pairs, parameter values, held-out probabilities, decoded segments, and companion metrics are identical at stored precision. The fitted scalers, objectives, selected checkpoints, and predicted segment sets also coincide.

Across the 180 additional pairs, serialized parameter tensors are identical.

K.2 IoU-qualified segment matching

Let M_{ij} denote the intersection over union between predicted interval i and reference interval j, and define

$$E = \{(i, j) : M_{ij} \geq 0.5\}.$$

Let $\mathfrak{M}(E)$ be the set of one-to-one matchings whose edges belong to E. We select

$$\max_{\mathcal{M} \in \mathfrak{M}(E)} \left(|\mathcal{M}|, \sum_{(i,j) \in \mathcal{M}} M_{ij} \right)$$

lexicographically: matching first maximizes the number of pairs and then total IoU among solutions with the same cardinality.

The implementation expresses this criterion as a linear-assignment problem with a cardinality term larger than any possible difference in total IoU. Segment F1, onset error, offset error, duration error, and their denominators all use this matching.

L Table Materialization Benchmark

L.1 Table transformations

L.2 End-to-end responses

Across the six transformations applied to three dataset–model configurations, ordinary segment F1 changes in 12 of 18 combinations and held-out probabilities change in 13. Aggregate segment-F1 responses range from -0.138 to $+0.071$, and the largest absolute held-out probability response is 0.587.

Row reordering leaves segment F1 unchanged in all three configurations, with probability responses below 10^{-3}. Repeated-export concatenation also leaves segment F1 unchanged in all three configurations; its probability response is 0.001 for HAPT–TCN and zero at the displayed precision for MHEALTH–TCN and HAPT–BiGRU. Quotient-learning responses are zero in each case.

Across the 18 ordinary combinations, the largest absolute changes in the initial training objective, validation objective, maximum-coordinate gradient, and fitted scaler are 0.0287, 0.0287, 0.0664, and 2.075, respectively. Quotient learning gives zero response at these stages.

Companion metrics also change in several combinations. For HAPT–TCN, the one-to-many metadata join changes balanced accuracy by $+0.0314$, Brier score by $+0.00679$, expected calibration error by $+0.00612$, and false-positive intervals by $+7.54$ per non-event hour. For HAPT–BiGRU, heterogeneous fan-out changes balanced accuracy by -0.0476 and false-positive intervals by -6.05 per non-event hour.

Comparison	Pairs	Updates per pair	Quantities compared
Fixed-step fitted outputs	15	25	Scaler, objectives, checkpoint, parameters, probabilities, decoded segments, and companion metrics
Parameter states	180	Varied	Serialized parameter tensors

Table 14: **Exact invariance comparisons.** The 15 fixed-step pairs compare fitted state and held-out outputs after 25 deterministic updates. The 180 additional pairs compare serialized parameter states across a broader set of runs.

Transformation	Table operation
One-to-many metadata join	Join weak statements to multiple auxiliary records, producing several stored rows from the same statement.
Repeated-export concatenation	Append additional exports of the same normalized weak-supervision table.
Exploded list metadata	Expand list-valued metadata into multiple stored rows.
Heterogeneous metadata fan-out	Apply different metadata expansion counts to different weak statements.
Deterministic row reordering	Permute stored rows while preserving their contents, evidence keys, and observed payloads.
Mixed sequence	Compose joining, repeated export, metadata expansion, and deterministic row ordering.

Table 15: **Table operations that preserve the weak statements.** Rows produced by these operations inherit the evidence identity of their parent statement.

Dataset/model	Table transformation	Ordinary ΔF1	Ordinary $\max \lvert\Delta p\rvert$	Checkpoint changes
HAPT–TCN	One-to-many metadata join	$+0.071$	0.586	1/3
HAPT–TCN	Repeated export	$+0.000$	0.001	0/3
HAPT–TCN	Exploded list metadata	-0.003	0.133	0/3
HAPT–TCN	Heterogeneous metadata fan-out	-0.057	0.416	0/3
HAPT–TCN	Row reorder	$+0.000$	0.000	0/3
HAPT–TCN	Mixed sequence	$+0.067$	0.587	1/3
MHEALTH–TCN	One-to-many metadata join	-0.006	0.254	0/3
MHEALTH–TCN	Repeated export	$+0.000$	0.000	0/3
MHEALTH–TCN	Exploded list metadata	-0.050	0.459	2/3
MHEALTH–TCN	Heterogeneous metadata fan-out	-0.138	0.584	2/3
MHEALTH–TCN	Row reorder	$+0.000$	0.000	0/3
MHEALTH–TCN	Mixed sequence	-0.006	0.246	0/3
HAPT–BiGRU	One-to-many metadata join	-0.002	0.025	0/3
HAPT–BiGRU	Repeated export	$+0.000$	0.000	0/3
HAPT–BiGRU	Exploded list metadata	$+0.011$	0.040	0/3
HAPT–BiGRU	Heterogeneous metadata fan-out	-0.079	0.257	0/3
HAPT–BiGRU	Row reorder	$+0.000$	0.000	0/3
HAPT–BiGRU	Mixed sequence	-0.001	0.024	0/3

Table 16: **Responses to six table transformations.** The six transformations are applied to three dataset–model configurations, giving 18 combinations. ΔF1 and Δp compare each transformed table with its corresponding normalized-table baseline. Ordinary segment F1 changes in 12 combinations and held-out probabilities in 13. Checkpoint counts are out of three folds. Quotient-learning responses are zero and are omitted.

L.3 PIPELINE RESPONSES

Figure 8 shows where the response first appears along the ordinary learning pipeline.

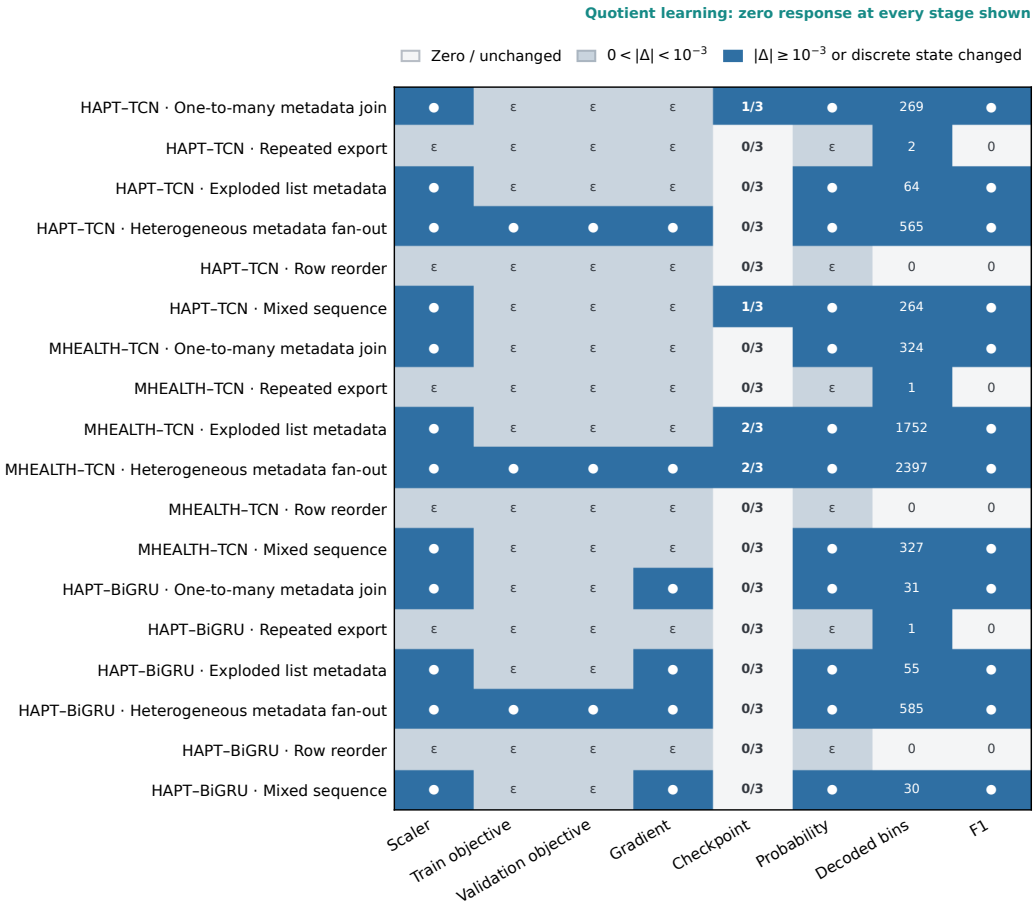

Figure 8: **Propagation of materialization responses through the ordinary pipeline.** Rows are dataset–model–transformation combinations and columns are pipeline stages. White cells denote zero response or no discrete change; ε denotes $0 < |\Delta| < 10^{-3}$; dark cells denote $|\Delta| \geq 10^{-3}$ or a discrete state change. Checkpoint entries give the number of changed folds out of three, and decoded-bin entries give the total number of changed temporal bins. Quotient-learning responses are zero at every stage shown.

M REPRODUCIBILITY MATERIALS

An anonymized code and results package accompanies the submission. It includes:

- the compiler, evidence-class construction, lineage rules, typed weak losses, and missing-value handling;
- executable tests covering multiplicity, row order, evidence-key boundaries, session identity, and inconsistent observations;
- normalized relational tables and implementations of the six table materializations;
- lineage-comparison, canonical-compilation, and inverse-class weighting implementations;
- annotation-type-by-multiplicity experiments, including the non-finite runs;

- held-group schedules, model configurations, optimization settings, evidence-class assignments, and the PPG-DaLiA held-out configuration;
- machine-readable results and the scripts used to generate all figures and tables; and
- the software-environment specification.

The public datasets are not redistributed. The compiler and its executable tests run without the raw datasets; full training uses the cited datasets under their original licenses.