SCOPE: SELF-CALIBRATED ORDERED PRINCIPAL ENCODING FOR TABULAR FOUNDATION MODELS IN HIGH-DIMENSIONAL DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular Foundation Models (TFMs) remain difficult to apply in High-Dimensional, Low-Sample Size (HDLSS) regimes, where thousands of features can exceed practical input limits and dimensionality reduction often requires dataset-specific design or tuning. We introduce SCOPE (Self-Calibrated Ordered Principal Encoding), a deterministic, tuning-free feature compression framework for constructing compact representations of high-dimensional tabular data. SCOPE builds on a global feature ordering produced by LENS (Layout-Enhanced Neighborhood Sequencing), which combines efficient path construction and refinement with a self-calibrated ordering budget. Given this ordering, SCOPE estimates spectral complexity, analytically determines the output dimension, partitions the feature sequence through transition-aware segmentation, and performs local principal projections to obtain a compact representation. Across high-dimensional benchmarks and frozen TabPFN-style backbones, we evaluate the proposed approach on predictive performance, uncertainty-aware ranking, computational efficiency, and sustainability. Results show that SCOPE achieves a favorable compression, prediction, and compute trade-off by reducing representation size, inference cost, fit time, and memory usage while retaining competitive predictive performance on HDLSS benchmarks.

1 INTRODUCTION

High-Dimensional, Low-Sample Size (HDLSS) tabular prediction remains challenging because, when $m \gg n$ (where $m = \#\text{features}$, $n = \#\text{samples}$), a limited number of observations must support learning over an extremely large feature space, creating substantial statistical and computational burdens. This challenge is particularly relevant to Tabular Foundation Models (TFMs), whose practical scalability depends not only on the number of samples but also on the dimensionality of the input representation. Earlier TabPFN releases, including TabPFN-2.5 (Grinsztajn et al., 2025), were primarily designed and evaluated for datasets with approximately $2{,}000$ features or fewer. TabPFN-3 (Grinsztajn et al., 2026) substantially increases this capacity and supports feature counts up to $20{,}000$ within its sample-feature operating regime, but the dimensionality bottleneck remains important for genuinely HDLSS data. Increasing feature dimensionality increases memory and computational requirements and can restrict the number of samples that can be processed efficiently. This motivates dimensionality reduction and feature compression methods that reduce the effective representation size while preserving predictive information.

The first challenge is therefore the dimensionality bottleneck itself: existing TFMs handle high-dimensional inputs differently, with no common structure-aware compression mechanism. TabDPT (Ma et al., 2025), TabDPT-Turbo (Hosseinzadeh et al., 2026), and TabSwift (Liu & Ye, 2026) incorporate PCA-based dimensionality reduction in their pipelines, while Google's TabFM (Kong & Das, 2026) currently imposes a maximum feature cap of 500 in its scikit-style interface. In contrast, models such as TabPFN-3 (Grinsztajn et al., 2026), TabICL (Jingang et al., 2025), TabICLv2 (QU et al., 2026), and TabPFN-Wide (Kolberg et al., 2025) can operate directly on substantially higher-dimensional representations. This variation across TFMs highlights the lack of a common approach for handling high-dimensional feature spaces. Recently, TACO (Zabërgja et al., 2026) introduced end-to-end sample compression for TFMs, addressing the growing context induced by large sample

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.

sets. However, sample compression does not remove the challenging bottleneck created by thousands or tens of thousands of input features. A model-agnostic feature compression mechanism is therefore desirable for reducing dimensionality before inference while remaining compatible with TFM backbones. A model-agnostic feature compression mechanism is therefore desirable for reducing dimensionality before inference while remaining compatible with TFM backbones. Rather than treating dimensionality reduction independently of feature structure, feature ordering and permutation learning can organize related features into contiguous neighborhoods that support local compression (Habib et al., 2026b; Barthel et al., 2025; Wang et al., 2026).

A second challenge is that dimensionality reduction itself can introduce substantial configuration effort and computational overhead. GOTabPFN (Habib et al., 2026b) categorizes tabular models according to their tuning requirements. Most TFMs require little or no hyperparameter tuning, whereas conventional neural and tree-based baselines often depend on extensive optimization. GOTabPFN lies between these settings because its GO-LR ordering and its compression stages require dataset-specific configuration. Such optimization can become computationally expensive, particularly when repeated across HDLSS datasets and cross-validation folds. Extensive training and hyperparameter search consume substantial computational resources (Cai et al., 2020), increase energy use and CO_2 emissions (Patterson et al., 2021), and reduce sustainability from a Green-AI perspective (Khan et al., 2025). TabPack (Gorishniy et al., 2026) further demonstrates the importance of reducing tuning requirements even for non-foundation tabular models. Moreover, as recommended by ProtoGate (Jiang et al., 2024) and GOTabPFN, HDLSS evaluation commonly requires repeated cross-validation, further amplifying the computational cost of dataset-specific optimization.

Motivated by these limitations, we introduce **SCOPE**, **S**elf-**C**alibrated **O**rdered **P**rincipal **E**ncoding, a deterministic and tuning-free feature compression framework for HDLSS tabular data. SCOPE treats dimensionality reduction as a structured compression problem rather than applying a global projection directly to an unordered feature matrix. It first obtains a locality-preserving feature arrangement using LENS and then automatically determines the representation budget from the spectral complexity of the data. The ordered feature axis is partitioned using transition-aware segmentation, after which local principal projections produce a compact representation that can be passed directly to TFMs. This design provides a self-calibrated interface between high-dimensional tabular data and TFMs without requiring a manually specified output dimension. Our contributions are:

- We develop **LENS** (**L**ayout-**E**nhanced **N**eighborhood **S**equencing), an efficient feature ordering method that combines medoid-initialized path construction, candidate-pruned Travelling Salesman Problem (TSP), and self-budgeted Minimum Linear Arrangement (MinLA) to provide structured feature layout used by SCOPE with NP-Hardness proof.
- We introduce **SCOPE** (**S**elf-**C**alibrated **O**rdered **P**rincipal **E**ncoding), a deterministic and tuning-free dimensionality reduction framework that automatically determines its compression budget from spectral complexity and constructs compact representations through ordered, transition-aware local principal projections.
- We provide theoretical characterization of dataset regimes introduced by Habib et al. (2026e;b). We establish rank-based guarantees and model-dependent risk bounds for the original taxonomy, together with robustness and finite sample calibration guarantees for its Micro-Macro refinement under explicit assumptions.
- We extensively evaluate SCOPE as a model-agnostic compression layer for TFMs against classical, projection-based, manifold, supervised, and ordering-based dimensionality reduction methods. We further introduce **HDLSS-UCBT** (HDLSS **U**ncertainty-**C**alibrated **B**radley-**T**erry (BT)), an uncertainty-aware cross-dataset ranking framework that uses BT aggregation (Bradley & Terry, 1952) to assess predictive stability.

2 Related Work

Dimensionality reduction and TFM-focused compression methods span projection, factorization, learned, supervised, manifold, order-aware, and sample-compression approaches. Detailed review on related work and positioning of SCOPE are provided in Appendix A.

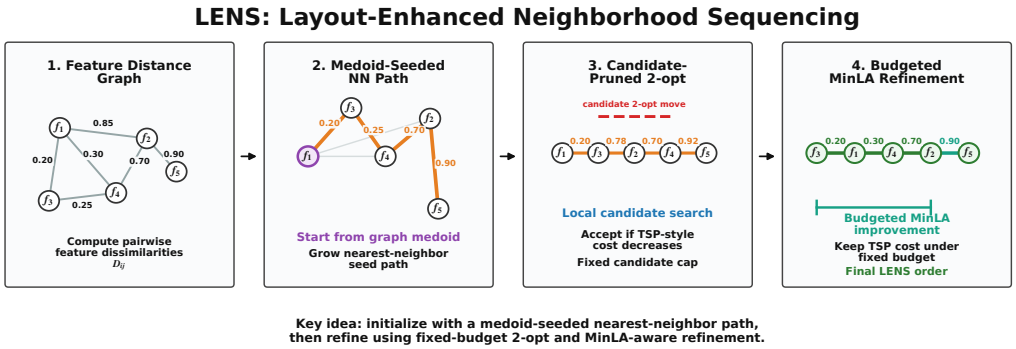


Figure 1: Overview of LENS feature ordering. Pairwise feature distances define the graph, a medoid-seeded nearest-neighbor path initializes ordering, candidate-pruned 2-opt reduces TSP cost, and budgeted MinLA-aware refinement produces the final order while preserving the TSP budget.

3 METHODOLOGY

Problem formulation. Let $X \in \mathbb{R}^{n\times m}$ denote a high-dimensional tabular dataset with n samples and m features, where typically $m \gg n$. Our objective is to construct a compact representation $Z \in \mathbb{R}^{n\times M^*}$ with $M^* \ll m$ while preserving informative structure in the original feature space, without using class labels, validation performance, downstream-model feedback, or dataset-specific hyperparameter search. We formulate the complete mapping as Eq. 1.

$$\Pi^* = \mathrm{LENS}(X), \qquad Z = \Phi_{\mathrm{SCOPE}}\left(X; \Pi^*\right) \in \mathbb{R}^{n\times M^*}, \qquad M^* \le m \tag{1}$$

where LENS constructs a deterministic global feature ordering Π^* and SCOPE uses this ordering to automatically determine the representation budget, partition the ordered feature sequence, and construct the final compressed representation. Thus, both the structural organization and the output dimensionality are determined directly from the input data.

3.1 LENS: LAYOUT-ENHANCED NEIGHBORHOOD SEQUENCING

Feature ordering can be formulated as a graph-layout problem that seeks to preserve local feature relationships while maintaining a coherent global arrangement. In high-dimensional settings, however, practical ordering procedures should achieve this balance without relying on dataset-specific solver configurations, clustering choices, refinement budgets, or downstream predictive feedback. We introduce **LENS**, a deterministic and tuning-free feature ordering algorithm that constructs a global feature sequence through a fixed TSP-first pipeline followed by self-budgeted MinLA refinement. LENS does not use validation-based hyperparameter search or downstream accuracy to select its ordering. Instead, its implementation consists of four stages: raw feature-distance graph construction using Euclidean distance by default; global medoid nearest-neighbor path initialization; distance-ordered candidate-pruned 2-opt refinement; and MinLA-budgeted 2-opt refinement. The resulting sequence combines adjacent feature smoothness with global layout quality through a deterministic optimization procedure whose refinement effort is controlled internally rather than through dataset-specific tuning. The nearest-neighbor and 2-opt components follow classical TSP heuristics (Rosenkrantz et al., 1977; Croes, 1958; Chandra et al., 1999). Let $X \in \mathbb{R}^{n\times m}$ denote a tabular dataset with n samples and m features, where $n \ll m$. We first standardize each feature column and form the transposed feature representation as $Z = X^\top \in \mathbb{R}^{m\times n}$, where each row $z_j \in \mathbb{R}^n$ represents the response profile of feature j across samples. Thus, LENS treats features as objects and constructs an ordering directly over the m feature profiles. Fig. 1 illustrates the LENS feature ordering pipeline, and Fig. 2a visualizes the resulting graph-based feature organization.

Raw Euclidean feature graph. LENS constructs a raw Euclidean feature-distance graph by Eq. 2.

$$D_{ij} = \|z_i - z_j\|_2, \qquad 1 \le i, j \le m \tag{2}$$

with $D_{ii} = 0$. The same distance matrix is used for both the TSP-path cost and the MinLA objective. The TSP objective measures adjacent feature dissimilarity by $C_{\mathrm{TSP}}(\Pi)$ in Eq. 3, where $\Pi = (\Pi_1, \ldots, \Pi_m)$ is a permutation of the features. The distance-weighted MinLA objective follows the classical linear arrangement formulation (Harper, 1964; Garey et al., 1974) as $C_{\mathrm{MinLA}}(\Pi)$

in Eq. 3, where $\mathrm{pos}_\Pi(i)$ denotes the position of feature i in ordering Π.

$$C_{\mathrm{TSP}}(\Pi) = \sum_{t=1}^{m-1} D_{\Pi_t,\Pi_{t+1}}, \qquad C_{\mathrm{MinLA}}(\Pi) = \sum_{1\le i<j\le m} D_{ij}\left|\mathrm{pos}_\Pi(i) - \mathrm{pos}_\Pi(j)\right| \tag{3}$$

Global medoid nearest-neighbor initialization. LENS constructs a global TSP-style Hamiltonian path over all features. The starting feature z_{s^*} is defined as medoid of feature distance graph, with s^* denoting its index. Starting from s^*, path repeatedly appends the nearest unvisited feature according to Eq. 4. This produces an initial path $\Pi^{(0)}$ without clustering features into predefined blocks.

$$z_{s^*} = \mathrm{medoid}\{z_i\}_{i=1}^m, \quad s^* = \arg\min_{1\le i\le m} \sum_{j=1}^m D_{ij}, \quad \Pi_1 = s^*, \quad \Pi_{t+1} = \arg\min_{j\notin\{\Pi_1,\ldots,\Pi_t\}} D_{\Pi_t,j} \tag{4}$$

Distance-ordered candidate-pruned 2-opt. The nearest-neighbor path is refined using a deterministic candidate-pruned 2-opt procedure. In general, k-opt TSP local search removes k edges and reconnects the resulting path components (Croes, 1958; Chandra et al., 1999). For each feature i, LENS stores the b_{TSP} nearest candidate neighbors according to D. For an ordering Π, consider two edges $(a, b) = (\Pi_i, \Pi_{i+1})$ and $(c, d) = (\Pi_j, \Pi_{j+1})$ with $j > i + 1$. A 2-opt reversal replaces these edges by (a, c) and (b, d) and reverses the segment $\Pi_{i+1:j}$. The corresponding TSP change is:

$$\Delta_{\mathrm{TSP}} = D_{a,c} + D_{b,d} - D_{a,b} - D_{c,d} \tag{5}$$

The reversal is accepted during the TSP phase only if $\Delta_{\mathrm{TSP}} < 0$. Candidates are traversed in increasing distance order, so the search prioritizes nearby feature connections rather than arbitrary feature-index order. Let the resulting ordering be denoted by Π^{TSP}.

Self-budgeted MinLA refinement. Although the TSP stage directly optimizes adjacent feature dissimilarity, it does not explicitly optimize the global MinLA objective in Eq. 3. LENS therefore performs a second refinement stage that reduces the MinLA objective while keeping the TSP cost within a self-calibrated budget. Baseline TSP cost is defined as $B_0 = C_{\mathrm{TSP}}\left(\Pi^{\mathrm{TSP}}\right)$. The median adjacent edge length in the TSP-refined ordering and the admissible TSP budget are given in Eq. 6.

$$\mu_{\mathrm{edge}} = \mathrm{median}\left\{D_{\Pi_t^{\mathrm{TSP}},\Pi_{t+1}^{\mathrm{TSP}}}\right\}_{t=1}^{m-1}, \qquad B_{\mathrm{TSP}} = B_0 + \eta\mu_{\mathrm{edge}} \tag{6}$$

where $\eta = 1$ in the selected implementation. This budget is computed directly from ordering and does not use validation accuracy / downstream model performance. During MinLA polishing, a candidate ordering Π' is accepted only if it satifies Eq. 7. LENS permits changes to TSP structure only within a data-adaptive one median-edge budget while explicitly reducing global MinLA objective.

$$C_{\mathrm{MinLA}}(\Pi') < C_{\mathrm{MinLA}}(\Pi), \qquad C_{\mathrm{TSP}}(\Pi') \le B_{\mathrm{TSP}} \tag{7}$$

Tuning-free configuration. LENS uses fixed constants $b_{\mathrm{TSP}} = 30$, $b_{\mathrm{MinLA}} = 4$, $\eta = 1$, and $R = 20$. The larger TSP budget reflects its cheaper local updates, while the smaller MinLA budget and finite refinement cap bound the more expensive global search; $\eta = 1$ allows one median-edge unit of TSP slack. Their bounded-search and scale-equivariant rationale is given in Appendix B.5. All values remain fixed across datasets, without validation or downstream model-based selection. Thus, LENS is tuning-free with respect to downstream predictive performance.

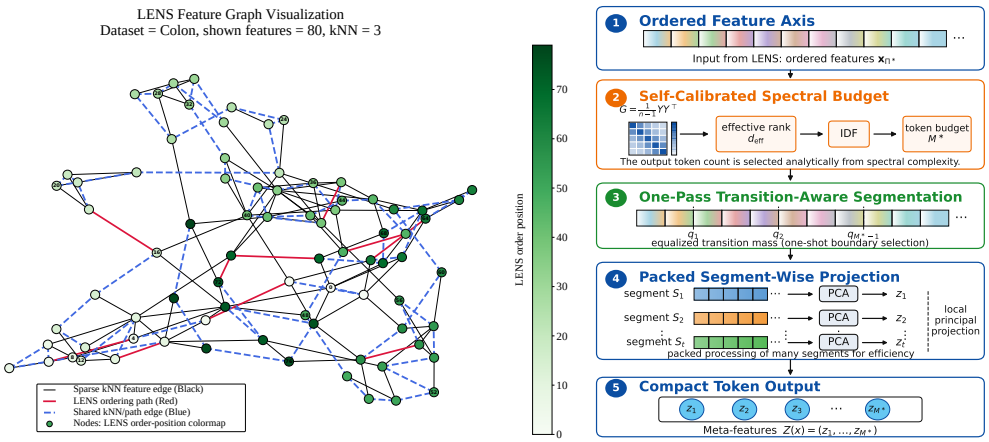
Hardness and approximation perspective. LENS is an efficient approximation rather than an exact solver for the underlying combinatorial ordering problem.

Theorem 1 (Computational hardness)**.** *The exact LENS ordering objective is NP-hard in the number of features m.*

The proof is in Appendix B.7. Additional theoretical properties, exact prefix-accelerated MinLA updates, complexity analysis, and ordering-solver evaluation protocol are provided in Appendix B.

3.2 SCOPE: Self-Calibrated Ordered Principal Encoding

High-dimensional tabular data require compact representations that reduce feature count while preserving informative structural relationships among features. This problem is particularly challenging in HDLSS regimes, where the appropriate representation size and grouping structure can vary substantially across datasets, making manually specified compression budgets or dataset-specific tuning undesirable. We introduce **SCOPE**, a deterministic and tuning-free framework that constructs compact representations directly from the geometry and spectral complexity of the training data. SCOPE



(a) (b)

Figure 2: **(a)** LENS feature graph on Colon: black edges denote sparse k-NN connections, red edges the learned ordering path, and blue dashed edges shared structure. The optimization trajectory appears in Appendix Fig. F.1, with the full animation in the supplementary material. **(b)** SCOPE pipeline: spectral complexity determines M^*, transition-aware segmentation partitions the LENS-ordered feature axis, and packed local principal projections produce $Z(x) = (z_1, \ldots, z_{M^*})$.

automatically determines both the representation budget and the structural partition of the ordered feature space, eliminating the need for validation-based configuration or downstream-model feedback. Specifically, SCOPE consists of four stages: global spectral-complexity estimation, analytical representation-budget selection, transition-aware one-shot partitioning, and segment-wise principal projection. Together, these stages produce a data-adaptive compressed representation whose dimensionality and partition structure are determined directly from the input geometry. No class labels, validation accuracy, downstream predictions, or hyperparameter search are used to configure the compression procedure. Fig. 2b presents the integrated SCOPE pipeline.

Ordered feature representation. Let $X_{\mathrm{tr}} \in \mathbb{R}^{n \times m}$ denote the training matrix, with n = number of training samples and m = number of input features. Feature-wise means and standard deviations are estimated using only the training fold. The standardized training matrix is defined as $\widetilde{X}_{\mathrm{tr}} = \mathrm{Standardize}(X_{\mathrm{tr}}) \in \mathbb{R}^{n \times m}$. Let $\Pi^* = (\Pi_1^*, \ldots, \Pi_m^*)$ be the global feature permutation produced by LENS. Denote the ordered training matrix by Eq. 8. Then, nearby columns of Y tend to correspond to features with similar response profiles under the feature geometry used by LENS.

$$Y = \left[\widetilde{X}_{\mathrm{tr},:,\Pi_1^*}, \ldots, \widetilde{X}_{\mathrm{tr},:,\Pi_m^*}\right] \in \mathbb{R}^{n \times m} \tag{8}$$

Self-calibrated spectral complexity. SCOPE estimates the intrinsic spectral complexity of the training data without requiring a manually selected cumulative explained-variance threshold. In the HDLSS regime, where $m \gg n$, the nonzero spectrum can be computed through the sample-space Gram matrix rather than the full $m \times m$ covariance matrix (Halko et al., 2011). We define $G = \frac{1}{n-1} YY^\top \in \mathbb{R}^{n \times n}$. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > 0$, where $r = \mathrm{rank}(G) \leq \min\{n-1, m\}$, denote the positive eigenvalues of G. Their normalized spectral masses are denoted by p_i (Eq. 9). Following entropy-based effective-rank estimation (Roy & Vetterli, 2007; Halko et al., 2011), we define Eq. 9. Since p_i in Eq. 9 is defined only over positive eigenvalues, $p_i > 0$ for all $i = 1, \ldots, r$. Therefore, Eq. 9 is well-defined without introducing a manually selected cumulative explained-variance threshold. The effective rank satisfies $1 \leq d_{\mathrm{eff}} \leq r \leq m$. We define the intrinsic-dimension fraction as $\mathrm{IDF} = \frac{d_{\mathrm{eff}}}{m}$. The quantity $\mathrm{IDF} \in (0, 1]$ measures the ratio between the effective spectral dimension and the ambient feature dimension. A smaller value indicates greater spectral redundancy and permits stronger compression.

$$p_i = \frac{\lambda_i}{\sum_{j=1}^{r} \lambda_j}, \quad i = 1, \ldots, r; \qquad d_{\mathrm{eff}} = \exp\left(-\sum_{i=1}^{r} p_i \log p_i\right) \tag{9}$$

One-shot analytical token budget. SCOPE determines the number of output tokens by balancing normalized representation size and average intrinsic load assigned to each token. For a candidate token budget M, we define $J(M)$ in Eq. 10. First term in $J(M)$ (Eq. 10) penalizes a large com-

pressed representation, whereas the second term penalizes excessive intrinsic complexity per token. Both terms in $J(M)$ are dimensionless and require no dataset-specific weighting. The objective is strictly convex for $M > 0$, yielding the unique continuous minimizer as $M_0 = \sqrt{m\, d_{\text{eff}}} = m\sqrt{\text{IDF}}$. Let B denote maximum number of input features supported by the downstream backbone. When no architectural limit is imposed, we set $B = m$. Feasible continuous budget is defined as $\widetilde{M} = \min\{M_0, m, B\}$. Final integer-valued token budget is selected by M^* in Eq. 10.

$$J(M) = \frac{M}{m} + \frac{d_{\text{eff}}}{M}, \qquad M^* = \underset{M \in \mathcal{C}(\widetilde{M})}{\arg\min} J(M) \tag{10}$$

where $\mathcal{C}(\widetilde{M})$ contains the feasible integer candidates adjacent to $\widetilde{M}$. Ties are resolved deterministically in favor of the smaller token budget. Complete integer construction and proof of optimality are provided in Appendix C.1.

Transition-aware one-shot segmentation. Let $D \in \mathbb{R}^{m \times m}$ denote the feature-distance matrix used by LENS. We reuse the same feature geometry to define the transition dissimilarity between consecutive ordered features by δ_j (Eq. 11). SCOPE therefore does not require a separate segmentation metric that may be inconsistent with the ordering objective. The cumulative transition mass is defined by A_j (Eq. 11). Let $A_{\text{tot}} = A_{m-1}$. We define the first and final boundaries as $q_0 = 0$ and $q_{M^*} = m$. When $A_{\text{tot}} > 0$, the preliminary t-th internal boundary is selected by Eq. 12.

$$\delta_j = D_{\Pi^*_j, \Pi^*_{j+1}}, \qquad A_j = \sum_{\ell=1}^{j} \delta_\ell, \qquad j = 1, \ldots, m-1 \tag{11}$$

$$\widetilde{q}_t = \min\left\{ j \in \{1, \ldots, m-1\} : A_j \geq \frac{t}{M^*} A_{\text{tot}} \right\}, \qquad t = 1, \ldots, M^* - 1 \tag{12}$$

Thus, each segment receives approximately the same proportion of the total transition mass. Repeated or highly concentrated transition masses can cause multiple preliminary boundaries to occur at the same feature position. We therefore apply the deterministic feasibility correction in Eq. 13. The final one-shot partition is defined by Eq. 14. All boundaries are constructed from a single cumulative scan over the ordered transition sequence. Therefore, no recursive splitting, intermediate segment PCA, or repeated fidelity evaluation is required. The degenerate transition case and proof that Eq. 13 produces a valid partition are provided in Appendix C.2.

$$q_t = \min\{m - M^* + t, \max\{q_{t-1} + 1, \widetilde{q}_t\}\} \tag{13}$$

$$S_t = \{q_{t-1} + 1, \ldots, q_t\}, \quad t = 1, \ldots, M^*; \qquad \mathcal{P}^* = \{S_1, \ldots, S_{M^*}\} \tag{14}$$

Segment-wise principal-subspace projection. Let $\mathcal{P}^* = \{S_1, \ldots, S_{M^*}\}$ denote the final one-shot partition. For each segment S_t, let $Y_{S_t} = Y_{:,S_t} \in \mathbb{R}^{n \times |S_t|}$ denote the corresponding ordered feature block. Its feature-wise mean vector is represented as $\mu_t = \frac{1}{n} Y_{S_t}^\top \mathbf{1}_n \in \mathbb{R}^{|S_t|}$. The centered segment matrix is defined as $\overline{Y}_{S_t} = Y_{S_t} - \mathbf{1}_n \mu_t^\top$. Let $\overline{Y}_{S_t} = U_t \Sigma_t V_t^\top$ be the singular value decomposition of the centered segment matrix. The first right singular vector is denoted by v_t. For an input sample $x \in \mathbb{R}^m$, let $\widetilde{x} = \text{Standardize}_{\text{tr}}(x)$ denote standardization using the training-fold statistics. Its ordered representation is $x^\Pi = \left(\widetilde{x}_{\Pi^*_1}, \ldots, \widetilde{x}_{\Pi^*_m}\right)$. The t-th compressed meta-feature is defined as $z_t(x) = \left(x^\Pi_{S_t} - \mu_t\right)^\top v_t$. The complete SCOPE representation is defined by Eq. 15. Each contiguous ordered segment contributes one principal meta-feature to the final representation. The principal-direction sign is fixed deterministically, and the segment projections are implemented using packed heterogeneous linear algebra for efficient execution. Implementation details, computational complexity, tuning-free properties, and complete pseudocode are provided in Appendix C.

$$\Phi_{\text{SCOPE}}(x) = (z_1(x), \ldots, z_{M^*}(x)) \in \mathbb{R}^{M^*} \tag{15}$$

Theorem 2 (Structural validity of SCOPE). *For a fixed LENS ordering Π^*, SCOPE selects an optimal feasible integer budget M^* under $J(M)$ and constructs exactly M^* nonempty, disjoint, and contiguous segments covering the complete ordered feature axis. Consequently, $\Phi_{\text{SCOPE}}(x) \in \mathbb{R}^{M^*}$ contains exactly one principal meta-feature from each segment.*

The proof is provided in Appendix C.3.

3.3 Theoretical Characterization of Dataset Regimes

Fig. D.1 in Appendix D summarizes our theoretical analysis of the dataset taxonomy introduced in DynaTab (Habib et al., 2026e) and GOTabPFN (Habib et al., 2026b). The feature to sample

ratio $\rho = m/n$ provides a distribution-free rank deficit certificate, with $\rho > 2$ implying that more than half of the feature space lies in the null space of the centered sample covariance. Under a Gaussian linear model, the admissible feature count instead depends on both sample size and a prescribed excess-risk tolerance. We establish deterministic stability properties for the Micro-Macro refinement and conditional finite sample calibration guarantees using Dvoretzky-Kiefer-Wolfowitz (DKW) inequality (Massart, 1990). Detailed proofs, and calibration analysis are in Appendix D.

3.4 HDLSS-UCBT: Uncertainty-Calibrated Bradley-Terry Ranking

To account for the substantial fold-level variability that can arise in HDLSS evaluation, we introduce **HDLSS Uncertainty-Calibrated Bradley-Terry (HDLSS-UCBT)**, an uncertainty-aware cross-dataset ranking framework. Rather than treating each pairwise performance difference as a deterministic win or loss, HDLSS-UCBT uses matched repeated cross-validation differences, corrected cross-validation uncertainty, and a Region of Practical Equivalence (RoPE) to construct fractional pairwise outcomes, which are jointly aggregated through a regularized Bradley-Terry model (Bradley & Terry, 1952; Nadeau & Bengio, 2003; Benavoli et al., 2017). Dataset-level bootstrap resampling further quantifies ranking stability. Complete formulation, implementation details, and numerical results are provided in Appendix E. We additionally report the **Uncertainty-Calibrated Pairwise Win score (UCPW)**, a dataset-balanced average of the fractional pairwise outcomes.

4 Experimental Results

Datasets and methods. We evaluate on 30 high-dimensional datasets with $m \gg n$, spanning biological HDLSS, face images represented as tabular features, bag of words text, and extreme dimensional omics. Most datasets have been adopted from prior HDLSS-oriented studies by Jiang et al. (2024); Yang et al. (2022); Yamada et al. (2020); Habib et al. (2026b;e); Kolberg et al. (2025); complete dataset sources are provided in App. J. Our comparative baselines span classical linear dimensionality reduction, such as PCA (Pearson, 1901), to recent nonlinear methods e.g. Param-Repulsor (Huang et al., 2024) and LocalMAP (Wang et al., 2025b), with further details provided in App. A. We primarily use TabPFN-3 (Grinsztajn et al., 2026) as downstream TFM and additionally evaluate TabICLv2 (QU et al., 2026) to assess generalization across TFM backbones.

Evaluation protocol. We use 5×5 repeated cross-validation, yielding 25 evaluations per dataset and method, following the HDLSS evaluation protocol used by Jiang et al. (2024); Habib et al. (2026b). Following Zabërgja et al. (2026), we evaluate predictive and computational performance using mean ROC-AUC, inference time, fit time, and memory consumption relative to the Raw TFM representation. We additionally report predictive retention (%) and introduce the uncertainty-aware HDLSS-UCBT rating to account for the substantial fold-level variability that can arise in the $m \gg n$ regime, building on the pairwise rating principle underlying Elo-style evaluation in TabArena (Erickson et al., 2025). Following Cai et al. (2020), we also measure energy consumption and estimated CO_2 emissions to assess sustainability and deployment efficiency. We also compare Raw and SCOPE + TabPFN-3 on 6 BeyondArena (Purucker et al., 2026) datasets with $m > n$ using pairwise Elo; Fig. L.2 shows the resulting efficiency-performance trade-off. All experiments were conducted on a Vulcan GPU cluster with 8×49 GB NVIDIA RTX A6000 GPUs, $2\times$ Intel Xeon Gold 5320 CPUs, and 503 GB RAM. SCOPE requires no hyper-parameter tuning at any stage.

Ordering quality comparison. Table 1 compares LENS with standard metaheuristics and tabular feature ordering methods on Colon using runtime, raw Euclidean TSP cost, raw Euclidean MinLA cost, and the proposed min-max normalized ordering score (Eq. 28). LENS achieves the highest overall score and the best TSP and MinLA costs. Relative to Simulated Annealing (Kirkpatrick et al., 1983), Genetic Algorithm (Larranaga et al., 1999), Ant Colony Optimization (Dorigo & Gambardella, 2002), and Christofides (Christofides, 2022), it attains better ordering quality at substantially lower computational cost. It also outperforms DynaTab-DFO (Habib et al., 2026e), TabSeq-FO (Habib et al., 2024), CausalTAD-LOP (Wang et al., 2026), and ROTATOR-LLM (Wang et al., 2025a), supporting the effectiveness of the TSP-first, MinLA-budgeted design.

Cross-domain results. Table 2 demonstrates that SCOPE transfers consistently across markedly different high-dimensional data domains when paired with TabPFN-3 (Grinsztajn et al., 2026). Among all dimensionality reduction methods, SCOPE achieves the highest average ROC-AUC and retention in each of the four domains, covering biological HDLSS, tabularized face images, high-dimensional text, and extreme dimensional omics. It retains 99.32% of Raw performance on bi-

Table 1: **Comparison of ordering solvers on Colon.** Lower values are better for runtime, TSP cost, and MinLA cost, while higher values are better for the ordering score. The score is the average min-max normalized reward from Eqs. 27 and 28. For each metric, gold, silver, and bronze denote the 1st, 2nd, & 3rd best methods. Ties at the reported precision are broken by lower runtime.

#	Solver	Time (s)↓	TSP Cost↓	MinLA Cost↓	Score↑	#	Solver	Time (s)↓	TSP Cost↓	MinLA Cost↓	Score↑
1	**LENS**	4.61	**11700.95**	1.4725×10^{10}	**0.999**	6	GO-LR	10.07	21958.78	1.4743×10^{10}	0.646
2	Sim. Annealing	15.01	11712.75	1.4803×10^{10}	0.915	7	DynaTab-DFO	0.05	21670.27	1.4764×10^{10}	0.635
3	Genetic Alg.	206.59	11712.75	1.4803×10^{10}	0.887	8	Christofides	1424.83	11715.06	1.4994×10^{10}	0.505
4	ROTATOR-LLM	45.01	13165.17	1.4781×10^{10}	0.886	9	TabSeq-FO	**0.01**	20712.33	1.5040×10^{10}	0.374
5	Ant Colony	1501.44	11792.08	1.4760×10^{10}	0.739	10	CausalTAD-LOP	2268.65	21283.64	1.4933×10^{10}	0.135

Table 2: **Cross-domain dimensionality reduction performance with TabPFN-3.** Results cover 12 biological HDLSS, 7 face image, 5 text, and 6 extreme dimensional omics datasets. **Avg. ROC-AUC** reports the mean ± sample standard deviation across dataset-level means, and **Retention** the mean dataset-wise performance relative to Raw. Dataset counts and dimension ranges appear below the headers. Raw is the uncompressed reference and is excluded from ranking. , , and denote the top three distinct compression results; † marks supervised dimensionality reduction. Per-dataset and TabICLv2 results are provided in Appendix Tables G.1 through G.4 and Appendix H.

Method	Biological HDLSS		Face Image		Text		Extreme Dimensional Omics	
	Avg. ROC-AUC	Retention (%)	Avg. ROC-AUC	Retention (%)	Avg. ROC-AUC	Retention (%)	Avg. ROC-AUC	Retention (%)
# Datasets	**12**		**7**		**5**		**6**	
Samples (n)	50 - 203		100 - 1,440		64 - 1,993		244 - 1,067	
Features (m)	2,000 - 22,283		1,024 - 10,304		3,289 - 8,266		42,728 - 67,420	
Raw	94.48 ± 6.65	100.00	99.21 ± 1.61	100.00	96.69 ± 3.44	100.00	96.76 ± 1.80	100.00
SCOPE (ours)	93.87 ± 7.07	99.32	99.30 ± 1.47	100.10	95.64 ± 2.30	98.98	97.23 ± 1.66	100.50
PCA	92.48 ± 7.03	97.89	99.22 ± 1.60	100.01	95.63 ± 2.58	98.93	93.61 ± 2.82	96.77
GRP	90.88 ± 7.51	96.12	99.05 ± 1.84	99.84	91.33 ± 4.62	94.46	94.23 ± 3.04	97.38
AE	92.31 ± 7.76	97.65	99.11 ± 1.67	99.90	92.47 ± 6.82	95.72	91.66 ± 4.08	94.72
UMAP	86.44 ± 8.00	91.45	94.20 ± 5.97	94.94	73.32 ± 14.15	76.03	85.18 ± 7.63	88.00
PaCMAP	82.46 ± 8.07	87.32	89.63 ± 8.06	90.37	60.41 ± 8.83	62.75	79.22 ± 12.19	81.86
LocalMAP	80.08 ± 7.73	84.81	86.69 ± 8.04	87.41	60.79 ± 9.16	63.14	77.66 ± 12.85	80.24
ParamRepulsor	82.46 ± 8.11	87.28	90.55 ± 10.60	91.29	84.91 ± 7.16	88.02	81.92 ± 11.75	84.62
Kernel PCA	90.95 ± 7.78	96.19	98.60 ± 1.93	99.38	94.61 ± 2.40	97.89	92.05 ± 3.32	95.13
PLS†	93.73 ± 6.79	99.20	98.85 ± 2.83	99.62	93.01 ± 4.60	96.17	93.03 ± 3.44	96.14
NSC-pSP	91.63 ± 7.88	96.91	99.20 ± 1.42	100.00	95.60 ± 3.40	98.88	94.44 ± 3.81	97.58

Retention is computed per dataset relative to Raw and then averaged by domain. Sample and feature ranges report the domain-wise minima and maxima.

ological data and 98.98% on text, while slightly exceeding the Raw representation on both face image (100.10% retention) and extreme dimensional omics (100.50%) benchmarks. This consistency is notable given that the datasets span from 1,024 to 67,420 features and differ substantially in modality, sample size, and statistical structure. Overall, the results position SCOPE as a robust model-agnostic compression interface for TabPFN-3 (Grinsztajn et al., 2026), preserving or improving predictive performance without relying on domain-specific dimensionality reduction design. Detailed per-dataset results for the four domains are provided in Appendix G - biological (Table G.4), face image (Table G.3), text (Table G.2), and extreme dimensional (Table G.1). We further validate backbone generalization with TabICLv2 (QU et al., 2026) in Appendix H.

Uncertainty-aware ranking. We evaluate 10 unsupervised compressors with HDLSS-UCBT across 30 high-dimensional datasets. As shown in Fig. 3, SCOPE attains the highest HDLSS-UCBT rating. Complete formulation, uncertainty analysis, UCPW, statistical analysis, bootstrap stability results, and detailed rankings are provided in Appendix E.

Computational efficiency. Fig. 4 shows that SCOPE achieves a favorable balance between predictive fidelity and efficiency with TabPFN-3. SCOPE retains 99.46% of Raw ROC-AUC while providing a 2.97× inference speedup, and retains 99.71% while achieving a 1.39× fit speedup. Importantly, SCOPE lies in the high-fidelity, efficiency-gain region in both panels, whereas many competing compressors either incur larger predictive degradation or provide less consistent acceleration across fitting and inference. These results indicate that SCOPE reduces downstream computational cost without materially sacrificing TabPFN-3 predictive performance. Additional TabPFN-3 Pareto and sustainability analyses are provided in Appendix G (Table G.5, Fig. G.1), with corresponding TabICLv2 fidelity-efficiency results in Appendix H (Figs. H.1-H.2,Table H.3).

Ablation studies. We further examine SCOPE through controlled block-model experiments and component-wise ablations. On the synthetic HDLSS block benchmark, SCOPE remains competi-

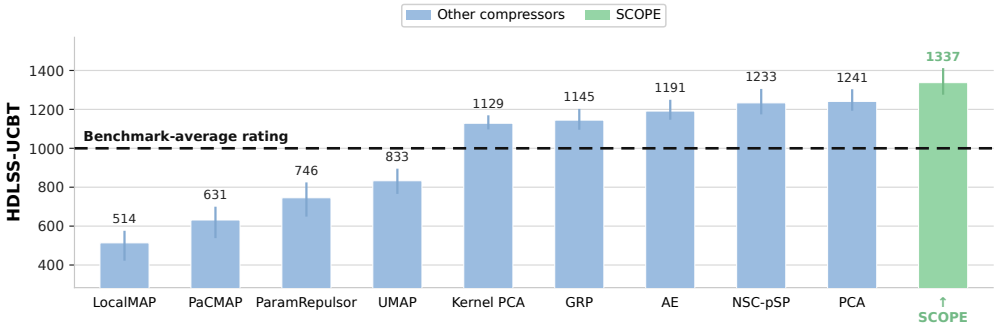
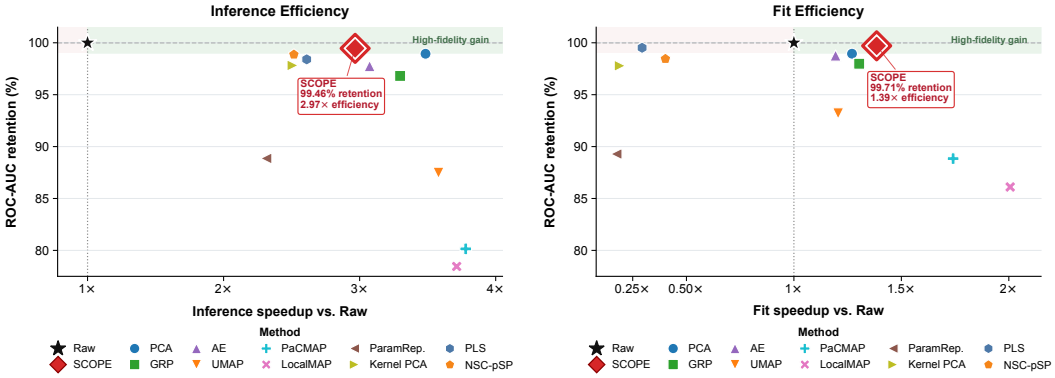


Figure 3: **TabArena-style HDLSS-UCBT leaderboard.** Ratings of 10 unsupervised compressors across 30 HDLSS datasets with TabPFN-3; error bars show 95% bootstrap intervals and the dashed line marks the benchmark-average rating. Detailed uncertainty statistics are provided in Fig. E.1.



(a) **Inference efficiency.** Predictive retention versus Raw-relative inference speedup.

(b) **Fit efficiency.** Predictive retention versus Raw-relative fit speedup.

Figure 4: **Predictive fidelity versus computational efficiency w/ TabPFN-3.** SCOPE remains in the high-fidelity efficiency region, preserving near-Raw ROC-AUC while reducing computational cost. Dashed horizontal and dotted vertical lines mark the Raw predictive and efficiency references. The shaded upper-right region denotes $\geq 99\%$ ROC-AUC retention with efficiency above $1\times$ Raw.

tive across predictive and representation-quality metrics under a fixed compression dimension, while the SMK dataset ablations show that individual component removals cause only modest ROC-AUC changes but can substantially affect representation efficiency. LENS stage analysis further shows that TSP refinement improves local path cost, whereas complete pipeline achieves the lowest MinLA objective under its self-calibrated TSP budget. Full synthetic and statistical analyses are provided in App. K, with detailed SCOPE and LENS ablations in App. L, and TabPFN stress test in App. I.

Limitations. The current formulation of SCOPE is specifically designed and validated for the HDLSS regime, where the number of features substantially exceeds the number of samples. In particular, our present analysis focuses on datasets with feature to sample ratio $\rho = m/n > 2$, where rank deficiency and the associated Gram-matrix structure provide the primary motivation for structured feature compression. Consequently, the present guarantees and empirical conclusions should not be assumed to extend directly to regimes with $\rho \leq 2$ or $n \gg m$. Extending SCOPE and its self-calibration mechanism to these sample-rich regimes remains for future work.

5 Conclusions

We present **SCOPE**, a tuning-free dimensionality reduction method for TabPFN-style TFMs in HDLSS regimes. LENS reorganizes the feature space to expose coherent local neighborhoods, allowing SCOPE to "see" through this structure and create scope for reliable TFM operation when $m \gg n$. By exploiting these ordered neighborhoods through self-calibrated compression, SCOPE provides a compact, structure-preserving interface that maintains near-Raw predictive performance while improving computational efficiency across diverse domains.

AI USE STATEMENT

As authors of the submitted manuscript, we affirm that the core research concept, research questions, and methodological formulations were developed by the human authors. The initial manuscript draft was also written by the authors. Large Language Models (LLMs) were used primarily for language polishing, grammatical correction, and general LaTeX formatting. For implementation, the core model package and its primary structure were initially designed and written by the authors, after which AI-assisted coding tools were used for code refinement and cleanup. AI tools were also used to edit or streamline some repetitive experimental notebook code. All scientific decisions, methodological contributions, experimental design, analysis, and final verification of the manuscript and code were performed by the authors.

ETHICS STATEMENT

This work does not raise specific ethical concerns. All datasets used in the experiments are publicly available, and their original sources are appropriately cited. The study does not involve the collection of new human-subject data, private information, or personally identifiable information. The authors are not aware of any conflicts of interest related to this work.

REPRODUCIBILITY STATEMENT

All reported results, including baseline results, are reproducible. We provide the model packages together with a cleaned subset of the experimental notebooks after removing identifying information. The released `.ipynb` notebooks retain their displayed outputs to facilitate direct verification of the reported experimental results. Some notebooks cannot be released in their current form because portions of their execution records contain identifying information. All datasets used in this study are publicly available, and we provide citations to their original sources (Appendix J) as well as prior HDLSS-oriented studies that used these datasets. The methodological visualizations in Figs. 1, 2a, 2b, and F.1 were generated using Python, and the corresponding visualization code is also provided. To further support Appendix Fig. F.1, we include both video and GIF animations of the feature ordering procedure (generated using NetworkX (Hagberg et al., 2008) in Python) in the supplementary material, together with the code used to generate these animations.

REFERENCES

Kai Uwe Barthel, Florian Tim Barthel, and Peter Eisert. Permutation Learning with Only N Parameters: From SoftSort to Self-Organizing Gaussians. In *2025 33rd European Signal Processing Conference (EUSIPCO)*, pp. 1892–1896. IEEE, 2025. doi: 10.23919/EUSIPCO63237.2025.11226796.

Alessio Benavoli, Giorgio Corani, Janez Demsar, and Marco Zaffalon. Time for a Change: A Tutorial for Comparing Multiple Classifiers Through Bayesian Analysis. *Journal of Machine Learning Research*, 18(77):1–36, 2017. URL https://jmlr.org/papers/v18/16-305.html.

Hervé Bourlard and Yves Kamp. Auto-Association by Multilayer Perceptrons and Singular Value Decomposition. *Biological Cybernetics*, 59(4):291–294, 1988. doi: 10.1007/BF00332918.

Ralph Allan Bradley and Milton E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4):324–345, 1952. doi: 10.2307/2334029.

Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-All: Train One Network and Specialize it for Efficient Deployment. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HylxE1HKwS.

Barun Chandra, Howard Karloff, and Craig Tovey. New Results on the Old k-Opt Algorithm for the Traveling Salesman Problem. *SIAM Journal on Computing*, 28(6):1998–2029, 1999. doi: 10.1137/S0097539793251244.

Nicos Christofides. Worst-Case Analysis of A New Heuristic for the Travelling Salesman Problem. In *Operations Research Forum*, volume 3, pp. 20. Springer, 2022. doi: 10.1007/s43069-021-00101-z.

Benoît Courty, Victor Schmidt, Goyal-Kamal, inimaz, Marion Coutarel, Luis Blanche, Boris Feld, Jérémy Lecourt, Liam Connell, Amine Saboni, Patrick Lloret, SabAmine, supatomic, Mathilde Léval, cianc, Alexis Cruveiller, ouminasara, Franklin Zhao, Christian Bauer, Jerry Laruba Festus, Aditya Joshi, Alexis Bogroff, Hugues de Lavoreille, Niko Laskaris, Alexandre Phiev, Edoardo Abati, Douglas Blank, and rosekelly6400. mlco2/codecarbon: v3.2.6, March 2026. URL https://doi.org/10.5281/zenodo.19334697.

Georges A. Croes. A Method for Solving Traveling-Salesman Problems. *Operations Research*, 6 (6):791–812, 1958. doi: 10.1287/opre.6.6.791.

Sanjoy Dasgupta and Anupam Gupta. An Elementary Proof of A Theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003. doi: 10.1002/rsa.10073.

Janez Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7:1–30, 2006. URL https://www.jmlr.org/papers/volume7/demsar06a/demsar06a.pdf.

Marco Dorigo and Luca Maria Gambardella. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, 1 (1):53–66, 2002. doi: 10.1109/4235.585892.

Bradley Efron. Bootstrap Methods: Another Look at the Jackknife. In *Breakthroughs in Statistics: Methodology and Distribution*, pp. 569–593. Springer, 1992. doi: 10.1007/978-1-4612-4380-9_41.

Ran Eisenberg and Ofir Lindenbaum. Learning Permutation from Structure Without Supervision. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=H8ozFYOmk8.

Ran Eisenberg, Jonathan Svirsky, and Ofir Lindenbaum. COPER: Correlation-based Permutations for Multi-View Clustering. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=5ZEbpBYGwH.

Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Desai, David Salinas, and Frank Hutter. TabArena: A Living Benchmark for Machine Learning on Tabular Data. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://openreview.net/forum?id=jZqCqpCLdU.

Michele Filannino. DBWorld E-Mail Classification Using A Very Small Corpus. *The University of Manchester*, 86, 2011a. URL http://www.michelefilannino.com/public/papers/dbworld_email_classification.pdf.

Michele Filannino. DBWorld e-mails. UCI Machine Learning Repository, 2011b. URL https://doi.org/10.24432/C5589M.

Milton Friedman. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937. doi: 10.1080/01621459.1937.10503522.

Michael R Garey, David S Johnson, and Larry Stockmeyer. Some Simplified NP-Complete Problems. In *Proceedings of the Sixth Annual ACM Symposium on Theory of Computing*, pp. 47–63. ACM, 1974. doi: 10.1145/800119.803884.

Yury Gorishniy, Akim Kotelnikov, Ivan Rubachev, and Artem Babenko. TabPack: Efficient Hyperparameter Ensembles for Tabular Deep Learning. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=8qHjrpxrQn.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, et al. TabPFN-2.5: Advancing the State of the Art in Tabular Foundation Models. *arXiv preprint arXiv:2511.08667*, 2025. doi: 10.48550/arXiv.2511.08667.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Mihir Manium, Shi Bin Hoo, Magnus Bühler, Anurag Garg, Dominik Safaric, Jake Robertson, Benjamin Jäger, Simone Alessi, Adrian Hayler, Vladyslav Moroshan, Lennart Purucker, Philipp Singer, Alan Arazi, Julien Siems, Jan Hendrik Metzen, Georg Grab, Nick Erickson, Siyuan Guo, Eliott Kalfon, Simon Bing, David Salinas, Clara Cornu, Lilly Charlotte Wehrhahn, Diana Kriuchkova, Kursat Kaya, Lydia Sidhoum, Marie Salmon, Jerry Chen, Madelon Hulsebos, Yann LeCun, Samuel Müller, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. TabPFN-3: Technical Report. *arXiv preprint arXiv:2605.13986*, 2026. doi: 10.48550/arXiv.2605.13986.

Al Zadid Sultan Bin Habib, Kesheng Wang, Mary-Anne Hartley, Gianfranco Doretto, and Donald A. Adjeroh. TabSeq: A Framework for Deep Learning on Tabular Data via Sequential Ordering. In *International Conference on Pattern Recognition (ICPR)*, pp. 418–434. Springer, 2024. doi: 10.1007/978-3-031-78128-5_27.

Al Zadid Sultan Bin Habib, Md Younus Ahamed, Prashnna Kumar Gyawali, Gianfranco Doretto, and Donald A. Adjeroh. BSTabDiff: Block-Subunit Diffusion Priors for High-Dimensional Tabular Data Generation. In *ICLR 2026 2nd Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*, 2026a. URL https://openreview.net/forum?id=RKNDy0KhGT.

Al Zadid Sultan Bin Habib, Md Younus Ahamed, Prashnna Kumar Gyawali, Gianfranco Doretto, and Donald A. Adjeroh. GOTabPFN: From Feature Ordering to Compact Tokenization for Tabular Foundation Models on High-Dimensional Data. *Forty-Third International Conference on Machine Learning (ICML)*, 2026b. URL https://openreview.net/forum?id=fpqfV3lCIB.

Al Zadid Sultan Bin Habib, Md Younus Ahamed, Prashnna Kumar Gyawali, Gianfranco Doretto, and Donald A Adjeroh. iStructTab: Structured Feature Sequencing for Multimodal Learning of Image and Tabular Data. In *International Conference on Pattern Recognition (ICPR)*, pp. 650–665. Springer, 2026c. doi: 10.1007/978-3-032-31404-8_43.

Al Zadid Sultan Bin Habib, Md Younus Ahamed, Prashnna Kumar Gyawali, Gianfranco Doretto, and Donald A Adjeroh. iSyncTab: Learning Cross-Modal Feature Sequencing for Image-Tabular Data via Neural Synchrony. In *European Conference on Computer Vision (ECCV)*, pp. 337–359. Springer, 2026d. doi: 10.1007/978-3-032-37035-8_19.

Al Zadid Sultan Bin Habib, Gianfranco Doretto, and Donald A. Adjeroh. DynaTab: Dynamic Feature Ordering as Neural Rewiring for High-Dimensional Tabular Data. In *Proceedings of the First Workshop on NeuroAI Multimodal Intelligence @ AAAI 2026*, volume 308 of *Proceedings of Machine Learning Research*, pp. 27–57. PMLR, 2026e. URL https://proceedings.mlr.press/v308/habib26a.html.

Aric A Hagberg, Daniel A Schult, and Pieter J Swart. Exploring Network Structure, Dynamics, and Function Using NetworkX. In *Proceedings of the Python in Science Conference*, pp. 11–15. SciPy, 2008. doi: 10.25080/tcwv9851.

Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Review*, 53(2):217–288, 2011. doi: 10.1137/090771806.

Lawrence Hueston Harper. Optimal Assignments of Numbers to Vertices. *Journal of the Society for Industrial and Applied Mathematics*, 12(1):131–135, 1964. doi: 10.1137/0112012.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in High-Dimensional Ridgeless Least Squares Interpolation. *Annals of Statistics*, 50(2):949, 2022. doi: 10.1214/21-aos2133.

Aura Hernández-Sabat, Antonio M. López, and Katerine Diaz-Chito. DrivFace. UCI Machine Learning Repository, 2016. URL https://doi.org/10.24432/C5XC7Q.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate Predictions on Small Data with a Tabular Foundation Model. *Nature*, 637(8045):319–326, 2025. doi: 10.1038/s41586-024-08328-6.

Rasa Hosseinzadeh, Alex Labach, Zexin Xue, Shuyi Han, Valentin Thomas, and Anthony L Caterini. TabDPT-Turbo: Efficient In-Context Learning for Tabular Prediction. In *2nd ICML Workshop on Foundation Models for Structured Data*, 2026. URL https://openreview.net/forum?id=Y00pwFyrHR.

Haiyang Huang, Yingfan Wang, and Cynthia Rudin. Navigating the Effect of Parametrization for Dimensionality Reduction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=eYNYnYle41.

Benjamin Jäger, Nick Erickson, Léo Grinsztajn, Felix Birkel, Klemens Flöge, Oscar Key, Kürşat Kaya, Jonas Kübler, Adèle Frankel, Tobias Schröder, Anurag Garg, Jan Hendrik Metzen, David Salinas, Simon Bing, Kristina Collins, Tuana Çelik, Vahid Balazadeh, Lydia Sidhoum, Tomás Pereda, Brendan Roof, Andrej Tschalzev, Siyuan Guo, Philipp Singer, Lennart Purucker, Jake Robertson, Marie Salmon, Philipp Jund, Jerry Chen, Diana Kriuchkova, Arthur Cahu, Eliott Kalfon, Adrian Hayler, Georg Grab, Vitor Monteiro, Lilly Wehrhahn, Dominik Safaric, Clara Cornu, Alan Arazi, Rylee Grace, Simone Alessi, Mihir Manium, Bernhard Schölkopf, Yann LeCun, Madelon Hulsebos, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. TabPFN-3.5: Technical Report. *arXiv preprint arXiv:2609.17895*, 2026. doi: 10.48550/arXiv.2609.17895.

Xiangjian Jiang, Andrei Margeloiu, Nikola Simidjievski, and Mateja Jamnik. ProtoGate: Prototype-based Neural Networks with Global-to-local Feature Selection for Tabular Biomedical Data. In *International Conference on Machine Learning (ICML)*, pp. 21844–21878. PMLR, 2024. URL https://proceedings.mlr.press/v235/jiang24c.html.

QU Jingang, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A Tabular Foundation Model for In-Context Learning on Large Data. In *International Conference on Machine Learning (ICML)*. PMLR, 2025.

William B Johnson and Joram Lindenstrauss. Extensions of Lipschitz Mappings into A Hilbert Space. *Contemporary Mathematics*, 26(189-206):1, 1984.

Mateusz Jurewicz and Leon Derczynski. Set Interdependence Transformer: Set-to-Sequence Neural Networks for Permutation Learning and Structure Prediction. *In Proceedings on the International Joint Conferences on Artificial Intelligence (IJCAI-22)*, 2022. doi: 10.24963/ijcai.2022/434.

Richard M Karp. Reducibility Among Combinatorial Problems. In *50 Years of Integer Programming 1958-2008: from the Early Years to the State-of-the-Art*, pp. 219–241. Springer, 2009. doi: 10.1007/978-3-540-68279-0_8.

Sunawar Khan, Naila Sammar Naz, Tehseen Mazhar, Muhammad Usman Tariq, Tariq Shahzad, Sghaier Guizani, and Habib Hamam. Green AI Techniques for Reducing Energy Consumption in AI Systems. *Array*, pp. 100652, 2025.

Scott Kirkpatrick, C Daniel Gelatt Jr, and Mario P Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983. doi: 10.1126/science.220.4598.671.

Christopher Kolberg, Katharina Eggensperger, and Nico Pfeifer. TabPFN-Wide: Continued Pre-Training for Extreme Feature Counts. In *EurIPS 2025 Workshop: AI for Tabular Data*, 2025. URL https://openreview.net/forum?id=iT5ldbTl3q.

Weihao Kong and Abhimanyu Das. Introducing TabFM: A Zero-Shot Foundation Model for Tabular Data. Google Research Blog, June 2026. URL https://research.google/blog/introducing-tabfm-a-zero-shot-foundation-model-for-tabular-data/. Accessed: 2026-09-12.

Mark A Kramer. Nonlinear Principal Component Analysis Using Autoassociative Neural Networks. *AIChE Journal*, 37(2):233–243, 1991. doi: 10.1002/aic.690370209.

Pedro Larranaga, Cindy M. H. Kuijpers, Roberto H. Murga, Inaki Inza, and Sejla Dizdarevic. Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators. *Artificial Intelligence Review*, 13(2):129–170, 1999. doi: 10.1023/A:1006529012972.

Daniel D Lee and H Sebastian Seung. Learning the Parts of Objects by Non-Negative Matrix Factorization. *nature*, 401(6755):788–791, 1999. doi: 10.1038/44565.

Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature Selection: A Data Perspective. *ACM Computing Surveys (CSUR)*, 50(6):94, 2018. doi: 10.1145/3136625.

Jundong Li et al. Datasets (scikit-feature / Feature Selection @ ASU). Website, 2026. URL https://jundongl.github.io/scikit-feature/datasets. Accessed: 2026-05-06.

Si-Yang Liu and Han-Jia Ye. TabSwift: An Efficient Tabular Foundation Model with Row-Wise Attention. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=ottLtChHp2.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Jesse C Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L Caterini, and Maksims Volkovs. TabDPT: Scaling Tabular Foundation Models on Real Data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=pIZxEOZCId.

Diana Mahdessian, Anthony J Cesnik, Christian Gnann, Frida Danielsson, Lovisa Stenström, Muhammad Arif, Cheng Zhang, Trang Le, Fredric Johansson, Rutger Schutten, et al. Spatiotemporal Dissection of the Cell Cycle with Single-Cell Proteogenomics. *Nature*, 590(7847):649–654, 2021. doi: 10.1038/s41586-021-03232-9.

Pascal Massart. The Tight Constant in the Dvoretzky-Kiefer-Wolfowitz Inequality. *The Annals of Probability*, pp. 1269–1283, 1990. URL https://www.jstor.org/stable/2244426.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint arXiv:1802.03426*, 2018. doi: 10.48550/arXiv.1802.03426.

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

NCBI. GEO Accession Viewer: GSE146773, National Center for Biotechnology Information. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE146773, 2021. Accessed: 2026-05-24.

David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. Carbon Emissions and Large Neural Network Training. *arXiv preprint arXiv:2104.10350*, 2021.

Karl Pearson. LIII. On Lines and Planes of Closest Fit to Systems of Points in Space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. doi: 10.1080/14786440109462720.

Lennart Purucker, Andrej Tschalzev, Nick Erickson, Gioia Blayer, David Holzmüller, Alan Arazi, Alexander Pfefferle, Mustafa Tajjar, Gaël Varoquaux, and Frank Hutter. Beyond IID: How General Are Tabular Foundation Models, Really?, 2026. URL https://arxiv.org/abs/2606.30410.

Jingang QU, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A Better, Faster, Scalable, and Open Tabular Foundation Model. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=SxsyLjIfWB.

Tasnia Rahman and Sathish AP Kumar. SIC-Autonomous Ensemble: Large Language Model Hallucination Mitigation Framework Using Self Iterative Contradiction and Optimal Ensembling. *IEEE Transactions on Artificial Intelligence*, 2026. doi: 10.1109/TAI.2026.3721061.

Daniel J. Rosenkrantz, Richard E. Stearns, and Philip M. Lewis. An Analysis of Several Heuristics for the Traveling Salesman Problem. *SIAM Journal on Computing*, 6(3):563–581, 1977. doi: 10.1137/0206041.

Olivier Roy and Martin Vetterli. The Effective Rank: A Measure of Effective Dimensionality. In *2007 15th European Signal Processing Conference*, pp. 606–610. IEEE, 2007.

Preeti Sah and Ernest Fokoué. What Do Asian Religions Have in Common? An Unsupervised Text Analytics Exploration. *arXiv preprint arXiv:1912.10847*, 2019. doi: 10.48550/arXiv.1912.10847.

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear Component Analysis as A Kernel Eigenvalue Problem. *Neural Computation*, 10(5):1299–1319, 1998. doi: 10.1162/089976698300017467.

Anton Frederik Thielmann, Manish Kumar, Christoph Weisser, Arik Reuter, Benjamin Säfken, and Soheila Samiee. Mambular: A Sequential Model for Tabular Deep Learning. *arXiv preprint arXiv:2408.06291*, 2024. doi: 10.48550/arXiv.2408.06291.

UCI. A study of Asian Religious and Biblical Texts. UCI Machine Learning Repository, 2019. URL https://doi.org/10.24432/C55S4W.

Guanchu Wang, Yuzhong Chen, Huiyuan Chen, Xiran Fan, Junpeng Wang, Xiaoting Li, Mingzhi Hu, Chia-Yuan Chang, and Xia Hu. Advancing Table Understanding of Large Language Models via Feature Re-ordering. *ACM SIGKDD Explorations Newsletter*, 27(1):112–123, 2025a. doi: 10.1145/3748239.3748248.

Ruiqi Wang, Ruikang Liu, Runyu Chen, Haoxiang Suo, Zhiyi Peng, Zhuo Tang, and Changjian Chen. CausalTAD: Injecting Causal Knowledge into Large Language Models for Tabular Anomaly Detection. *arXiv preprint arXiv:2602.07798*, 2026. doi: 10.48550/arXiv.2602.07798.

Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization. *Journal of Machine Learning Research*, 22(201):1–73, 2021. URL https://jmlr.org/papers/v22/20-1061.html.

Yingfan Wang, Yiyang Sun, Haiyang Huang, and Cynthia Rudin. Dimension Reduction with Locally Adjusted Graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 21357–21365, 2025b. doi: 10.1609/aaai.v39i20.35436.

Svante Wold, Arnold Ruhe, Herman Wold, and William J Dunn, III. The Collinearity Problem in Linear Regression. The Partial Least Squares (PLS) Approach to Generalized Inverses. *SIAM Journal on Scientific and Statistical Computing*, 5(3):735–743, 1984. doi: 10.1137/0905052.

Yutaro Yamada, Ofir Lindenbaum, Sahand Negahban, and Yuval Kluger. Feature Selection Using Stochastic Gates. In *International Conference on Machine Learning (ICML)*, pp. 10648–10659. PMLR, 2020. URL https://proceedings.mlr.press/v119/yamada20a.html.

Junchen Yang, Ofir Lindenbaum, and Yuval Kluger. Locally Sparse Neural Networks for Tabular Biomedical Data. In *International Conference on Machine Learning (ICML)*, pp. 25123–25153. PMLR, 2022. URL https://proceedings.mlr.press/v162/yang22i.html.

Ziwei Yang, Rikuto Kotoge, Xihao Piao, Zheng Chen, Lingwei Zhu, Peng Gao, Yasuko Matsubara, Yasushi Sakurai, and Jimeng Sun. MLOmics: Cancer Multi-Omics Database for Machine Learning. *Scientific Data*, 12(1):913, 2025. doi: 10.1038/s41597-025-05235-x.

Guri Zabërgja, Rafiq Kamel, Arlind Kadra, Christian Frey, and Josif Grabocka. End-to-End Compression for Tabular Foundation Models. In *Forty-third International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=84mfkGDxYh.

Fuzhen Zhang (ed.). *The Schur Complement and Its Applications*, volume 4 of *Numerical Methods and Algorithms*. Springer, New York, 2005. doi: 10.1007/b105056.

Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse Principal Component Analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006. doi: 10.1198/106186006X113430.

Supplementary Material for

SCOPE: Self-Calibrated Ordered Principal Encoding for Tabular Foundation Models on High-Dimensional Data

This supplementary document accompanies our main paper *SCOPE: Self-Calibrated Ordered Principal Encoding for Tabular Foundation Models in High-Dimensional Data*, submitted to the Fifteenth International Conference on Learning Representations (ICLR 2027). The supplementary material is organized into the following appendices:

Appendix A Detailed Related Work

Appendix B Additional Details for LENS

Appendix C Additional Details for SCOPE

Appendix D Theoretical Properties and Calibration of the Dataset Regime Taxonomy

Appendix E HDLSS-UCBT: Uncertainty-Calibrated Pairwise Ranking

Appendix F Visualization of LENS Ordering Dynamics

Appendix G Detailed Experimental Results

Appendix H Experiments with TabICLv2

Appendix I TabPFN Override Stress Test

Appendix J Dataset Sources

Appendix K HDLSS Block Model Analysis and Ablation for Dimensionality Reduction of SCOPE

Appendix L Detailed Ablation Analysis

A Detailed Related Work

Dimensionality reduction (DR) seeks to replace a high-dimensional representation $X \in \mathbb{R}^{n\times m}$ with a lower-dimensional representation $Z \in \mathbb{R}^{n\times M}$, where $M \ll m$, while preserving some notion of information or structure in the original data. Existing methods differ substantially in what they preserve: variance, pairwise distances, sparsity, reconstruction fidelity, neighborhood geometry, or target-associated covariance. These objectives are useful in general-purpose representation learning, but they do not directly optimize the requirements that arise when DR is used with a TFM. In particular, HDLSS data introduce an additional challenge: the representation must be estimated from relatively few observations while potentially compressing thousands or tens of thousands of features. Our work studies DR from this *feature compression* perspective, with particular emphasis on compression stability, predictive-performance retention, and downstream computational efficiency.

A.1 Linear, Projection-Based, and Factorization Methods

Principal Component Analysis. Principal Component Analysis (PCA) is one of the canonical approaches to linear dimensionality reduction (Pearson, 1901). PCA constructs mutually orthogonal directions that sequentially maximize the variance of the projected data, providing a compact global representation when most of the variance is concentrated in a small principal subspace. Its simplicity, deterministic formulation, and efficient implementations make PCA a natural DR baseline and have also led to its adoption inside recent TFM pipelines. However, its objective is global variance preservation rather than preservation of downstream predictive information. This distinction becomes especially important in HDLSS regimes. For a centered matrix with $m \gg n$, the empirical covariance matrix has rank at most $n-1$, so the estimated global eigenspace is necessarily supported by a small number of observations relative to the ambient dimension. Consequently, global principal directions can be sensitive to sampling variation even when they explain substantial empirical variance. Moreover, low-variance directions discarded by PCA may still contain information useful to a downstream predictor.

Gaussian Random Projection. Random projection provides a different approach in which the projection matrix is generated independently of the observed covariance structure. The Johnson-Lindenstrauss result establishes that a collection of points can be embedded into a substantially lower-dimensional Euclidean space while approximately preserving pairwise distances (Johnson & Lindenstrauss, 1984); subsequent work provides elementary constructions and analyses of this phenomenon (Dasgupta & Gupta, 2003). Dense Gaussian Random Projection (GRP) is therefore an attractive computational baseline because it avoids covariance estimation and expensive eigendecomposition. At the same time, its data-independent nature means that the projection is not adapted to localized feature dependencies or task-relevant structures. Distance preservation over samples therefore does not necessarily imply preservation of the feature interactions required by a downstream TFM.

Kernel and Sparse PCA. Kernel PCA generalizes PCA by performing principal-component analysis in an implicitly defined nonlinear feature space (Schölkopf et al., 1998). This allows nonlinear structure to be represented through an appropriate kernel, but introduces additional design choices such as the kernel family and its parameters. It also shifts the computation to the sample-space kernel matrix, and the resulting representation remains driven by a global variance criterion in the induced feature space. Sparse PCA instead modifies the PCA objective to obtain principal directions with sparse loadings (Zou et al., 2006), improving interpretability and potentially suppressing irrelevant features. This sparsity comes with additional regularization choices, however, and the resulting representation depends on the estimated global covariance structure. Both methods therefore provide important comparisons for determining whether structured local feature compression offers advantages beyond global linear or nonlinear principal-subspace estimation.

Non-negative Matrix Factorization. Non-negative Matrix Factorization (NMF) constructs a low-rank representation under non-negativity constraints and is widely used to obtain parts-based representations (Lee & Seung, 1999). Unlike an orthogonal PCA decomposition, NMF learns additive latent components whose interpretation can be useful when the data admit a naturally non-negative representation. However, the method requires compatible input preprocessing, is generally solved through iterative optimization, and its factorization objective is reconstruction-oriented rather than explicitly designed to preserve the behavior of a downstream TFM.

Partial Least Squares. Partial Least Squares (PLS) constructs latent variables using both predictors and the response, seeking directions associated with predictive covariance rather than variance alone (Wold et al., 1984). PLS is therefore fundamentally different from the unsupervised compressors considered above: it is a supervised dimensionality-reduction approach. This can make PLS competitive when the available labels identify informative directions that would receive little weight under an unsupervised variance criterion. At the same time, it requires label access during representation construction and must be fitted strictly within each training fold to avoid information leakage. We include PLS as a supervised reference point, whereas SCOPE constructs its representation without class labels, validation performance, or downstream-model feedback.

A.2 Learned and Nonlinear Dimensionality Reduction

Autoencoders. Autoassociative neural networks provide a learned alternative to spectral projection by reconstructing an input through a lower-dimensional bottleneck. Classical analyses connected autoassociative multilayer perceptrons to dimensionality reduction and principal-subspace learning (Bourlard & Kamp, 1988), while nonlinear autoassociative networks extended this idea beyond linear PCA-like representations (Kramer, 1991). In our setting, a shallow nonlinear autoencoder provides a natural learned-compression baseline: the encoder maps the original features to a bottleneck of the target dimension and the decoder is trained to reconstruct the original input. Autoencoders can capture nonlinear dependencies that global linear projections cannot, but their representation depends on optimization, architecture, initialization, and the amount of available training data. These considerations are particularly relevant when $n \ll m$, where a high-capacity reconstruction network is learned from relatively few observations. In contrast, SCOPE does not train a neural compressor and requires no gradient-based optimization.

Manifold and neighborhood-preserving methods. A complementary family of DR algorithms focuses on neighborhood and manifold structure in the sample space. UMAP constructs a neighborhood graph and derives a low-dimensional embedding using a topological and geometric formulation, providing a scalable general-purpose manifold-learning method (McInnes et al., 2018). PaCMAP explicitly balances nearby, mid-near, and far pairs in order to improve the preservation of both local and global structure (Wang et al., 2021). These approaches were developed in large part to produce informative low-dimensional embeddings, and the geometry they preserve can differ substantially from the geometry required for downstream prediction. Recent work has further examined shortcomings of graph-based and parametric dimensionality reduction. ParamRepulsor was introduced to address the observation that parametric DR models can retain global organization while losing important local detail; it combines hard-negative mining with stronger repulsive forces to improve local-structure preservation while retaining an explicit parametric mapping (Huang et al., 2024). LocalMAP instead adapts the neighborhood graph during optimization, motivated by the observation that similarity graphs constructed from high-dimensional distances can contain unreliable edges and omit important separation information (Wang et al., 2025b). These advances make them valuable modern baselines. Nevertheless, UMAP, PaCMAP, ParamRepulsor, and LocalMAP primarily optimize relations among *samples*. SCOPE instead organizes and compresses the *feature axis*: related features are first brought into local neighborhoods and these ordered neighborhoods are subsequently summarized. The two formulations consequently operate on different structural objects. More broadly, manifold preservation, reconstruction fidelity, explained variance, or pairwise-distance preservation are not equivalent to preserving predictive performance under a frozen downstream model. This mismatch motivates evaluating a feature compressor not only by geometric reconstruction criteria but also by its stability across resampled training sets and by the fraction of raw downstream predictive performance retained after compression.

A.3 Dimensionality Reduction Inside Tabular Foundation Models

The growth of TFMs makes feature dimensionality a systems issue in addition to a statistical one. In-context tabular predictors must process a representation whose cost depends on both the number of training examples and the number of input features. Accordingly, several recent TFMs already incorporate explicit mechanisms for controlling feature dimensionality. TabDPT applies PCA when the number of features exceeds its supported feature budget (Ma et al., 2025). The more recent TabDPT-Turbo similarly uses a fixed-width row representation and requires feature reduction, such as PCA or column subsampling, when the original number of variables exceeds that representation size (Hosseinzadeh et al., 2026). TabSwift also maps high-dimensional datasets to a fixed feature budget through PCA before row-wise TFM inference (Liu & Ye, 2026). Google's TabFM exposes a practical maximum-feature budget through its current estimator interface (Kong & Das, 2026). Other TFMs avoid an explicit PCA front end but still operate within characteristic feature regimes. TabPFN has progressively expanded its supported dimensionality, with newer checkpoints permitting substantially higher feature counts than earlier releases (Grinsztajn et al., 2026). TabICL and TabICLv2 similarly generalize beyond the dimensionalities encountered during pretraining, although their training and evaluation regimes remain concentrated in substantially lower-dimensional tables (Jingang et al., 2025; QU et al., 2026). Thus, feature dimensionality remains a relevant constraint even as the nominal feature capacity of TFMs increases. These developments suggest an important distinction between "model-specific dimensionality handling" and a "model-agnostic feature compressor". PCA-based preprocessing solves the immediate fixed-width input problem, but does not exploit structure among neighboring or related variables and requires an externally chosen target dimensionality. Likewise, increasing a TFM's maximum supported width reduces the need for hard truncation but does not remove the memory and computational cost associated with representing thousands of features. Our objective is therefore not simply to enable inputs that would otherwise exceed a hard model limit; it is to construct a smaller, stable representation that can improve the computational operating point of a TFM while retaining as much of its raw predictive behavior as possible.

Reliability perspective. Reliability provides an additional motivation for controlling the representation supplied to foundation models. In Large Language Models (LLMs), hallucination remains a fundamental concern: even dedicated self-refinement and ensembling frameworks can substantially mitigate unsupported or internally inconsistent generations without eliminating the underlying failure mode (Rahman & Kumar, 2026). Standard TFM classification and regression differ funda-

mentally from free-form language generation because predictions are produced over mathematically defined tabular inputs and constrained output spaces rather than generated as unconstrained text. Nevertheless, TFMs are not immune to reliability failures: in HDLSS regimes, redundant, noisy, or weakly estimated feature directions can perturb the representation presented to the predictor and lead to unstable or poorly calibrated predictions. From this perspective, appropriate feature compression can serve as a complementary representation-level safeguard: rather than correcting an erroneous output after inference, it reduces the dimensional complexity presented to the TFM while seeking to retain the predictive structure of the original data. SCOPE is designed around this objective, with its effectiveness evaluated through predictive-performance retention and cross-fold stability relative to inference on the uncompressed representation.

A.4 Order-Aware Feature Compression

GOTabPFN (Habib et al., 2026b) applies dimensionality reduction to an reordered feature matrix, it first constructs a global feature ordering through Graph-guided Ordering with Local Refinement (GO-LR). Features that are close under the learned feature geometry are placed near one another, after which Neuro-Inspired Subunit Compression (NSC) pools contiguous neighborhoods into compact meta-features. This established the key principle that feature ordering and feature compression should be considered jointly in HDLSS tabular learning. GOTabPFN studies four NSC variants: NSC, NSC-P, NSC-SP, and NSC-pSP. The variants separate two design choices: how the meta-feature budget or intrinsic dimensionality is determined, and how each ordered segment is summarized. NSC provides the base structured pooling formulation; NSC-P incorporates a PCA-inspired intrinsic dimensionality rule; NSC-SP studies segment-wise principal component pooling; and NSC-pSP combines the PCA-derived intrinsic dimension mechanism with segment-wise PCA compression. GOTabPFN adopts NSC-pSP in its principal experiments (Habib et al., 2026b). Collectively, these variants show that exploiting ordered local feature structure can provide a useful alternative to applying a single global dimensionality-reduction map. SCOPE builds on this structural insight but changes the role of the compressor. Rather than treating compression as a configured component of a specific GOTabPFN pipeline, SCOPE is formulated as an independent, model-agnostic feature compression layer. Its representation size is determined directly from the spectral complexity of the training data rather than from a manually specified output dimension or downstream validation performance. It then uses the ordered transition structure to construct a one-shot partition and applies local principal subspace projection inside the resulting segments. Thus, the method preserves the central motivation of order-aware local compression while removing validation-based selection of the compression budget and decoupling the compressor from a particular TFM checkpoint. This distinction is particularly relevant in HDLSS settings. Global dimensionality reduction methods estimate a single transformation over the full feature space, whereas SCOPE first reorganizes that space into locally coherent regions and only then estimates low-dimensional projections within those regions. The resulting hypothesis is that localizing the compression problem can make the representation less sensitive to the severe global rank deficiency that arises when $m \gg n$. Our experiments test this hypothesis through repeated folds and evaluate not only mean predictive performance but also its variability after compression.

A.5 Sample Compression for TFMs and TACO

Our broader motivation is also closely related to TACO (Zabërgja et al., 2026), which demonstrates that compression itself can be an effective mechanism for making TFM inference practical. TACO introduces an end-to-end learned compressor that maps a large training context containing N rows to a much smaller latent context containing $K \ll N$ rows. The downstream predictor therefore attends to the compressed context rather than the complete training set. This substantially reduces inference-time computation and memory and shows that aggressively reducing the representation supplied to a TFM need not necessarily cause a correspondingly large loss in predictive performance. TACO and SCOPE, however, address orthogonal dimensions of the tabular matrix. TACO performs *sample or context compression*, $N \longrightarrow K$, to reduce the number of training instances presented to the in-context predictor. SCOPE performs *feature compression*, $m \longrightarrow M^*$, to reduce the number of features describing each instance. Moreover, TACO learns its compressor and predictor jointly through end-to-end pretraining, whereas SCOPE is designed to operate in front of an already-pretrained, frozen TFM and requires no end-to-end retraining. TACO therefore motivates our systems-level view of compression while addressing a complementary scaling bottleneck.

In principle, the two directions are composable: row compression can reduce context length while feature compression simultaneously reduces representation width.

A.6 Feature Ordering in Tabular Learning

Feature ordering provides a promising structural basis for such compression. Recent studies increasingly show that feature order is a consequential design choice for tabular learning, especially in high-dimensional regimes. GOTabPFN (Habib et al., 2026b) uses graph-guided ordering to place locally related features together and compresses these neighborhoods into meta-features, enabling TabPFN inference on HDLSS data under limited computational budgets. DynaTab (Habib et al., 2026e) first investigated *when* feature ordering is beneficial, while GOTabPFN formalized *when and why* it matters in high-dimensional tabular learning and established the NP-hardness of the underlying ordering problem. CausalTAD (Wang et al., 2026) and ROTATOR-LLM (Wang et al., 2025a) developed feature ordering methods for tabular LLM interfaces. Concurrently, Mambular (Thielmann et al., 2024) showed that random feature permutations can alter predictive performance, whereas TabSeq (Habib et al., 2024) introduced an explicit feature ordering algorithm; subsequent work (Eisenberg & Lindenbaum, 2026) has further confirmed the importance of permutation learning. More generally, permutation learning optimizes the arrangement of a finite set for a downstream objective, often through differentiable relaxations of discrete permutations (Barthel et al., 2025; Jurewicz & Derczynski, 2022). This is particularly relevant to tabular data, which lack the inherent spatial or sequential structure that provides useful inductive bias. COPER (Eisenberg et al., 2025) incorporates a permutation-based objective for image-tabular multimodal learning, while iStructTab (Habib et al., 2026c) and iSyncTab (Habib et al., 2026d) further explore structured and cross-modal feature ordering for multimodal tabular representations. while TabICL (Jingang et al., 2025) and TabSwift (Liu & Ye, 2026) ensemble predictions across feature permutations. Nevertheless, learning cross-feature dependencies without overfitting remains difficult in HDLSS regimes, where even MLPs or Lasso can outperform more sophisticated approaches (Jiang et al., 2024). Rather than treating dimensionality reduction independently of feature structure, ordering can organize related features into contiguous neighborhoods that can be summarized locally.

A.7 Positioning of SCOPE

The preceding literature exposes a gap between general purpose dimensionality reduction and the requirements of high-dimensional TFM inference. Classical methods such as PCA, GRP, Kernel PCA, Sparse PCA, NMF, PLS, and autoencoders provide powerful generic compression mechanisms (Pearson, 1901; Johnson & Lindenstrauss, 1984; Dasgupta & Gupta, 2003; Schölkopf et al., 1998; Zou et al., 2006; Lee & Seung, 1999; Wold et al., 1984; Bourlard & Kamp, 1988; Kramer, 1991), while UMAP, PaCMAP, ParamRepulsor, and LocalMAP provide increasingly sophisticated mechanisms for preserving sample geometry (McInnes et al., 2018; Wang et al., 2021; Huang et al., 2024; Wang et al., 2025b). However, these methods were not designed around the combined objective of (i) operating reliably when $m \gg n$, (ii) automatically selecting a TFM-compatible representation size, (iii) exploiting relationships along the feature axis, and (iv) preserving the predictive behavior of a downstream TFM under substantial compression. GOTabPFN (Habib et al., 2026b) provides the closest feature-side precedent by demonstrating the value of order-aware local compression, while TACO (Zabërgja et al., 2026) establishes the complementary benefit of learned sample-side compression for TFM scalability. SCOPE connects these directions by treating feature reduction as a first-class compression problem for pretrained TFMs: it constructs a deterministic ordered representation, derives its compression budget directly from the input data, and performs localized principal projection without labels, validation feedback, or end-to-end TFM training. Our evaluation correspondingly goes beyond post-compression point estimates by measuring predictive retention relative to the raw TFM, inference and fitting time, memory consumption, and energy/CO_2 cost, while introducing **HDLSS-UCBT** to provide uncertainty-aware cross-dataset ranking from paired repeated-cross-validation outcomes. This separates the proposed representation method from its statistical evaluation and directly tests whether dimensionality reduction can serve as a reliable computational interface between high-dimensional data and TFMs rather than merely as a visualization or reconstruction tool.

B ADDITIONAL DETAILS FOR LENS

This section provides additional theoretical and implementation details for LENS, including properties of the initialization and refinement procedures, the prefix-accelerated MinLA update, computational complexity, the complete optimization algorithm, and the protocol used to compare ordering solvers. The main LENS methodology is defined in Section 3.1.

B.1 EVALUATION CONSISTENCY

The MinLA objective in Eq. 3 uses the raw Euclidean feature-distance matrix to remain consistent with the direct comparison protocol used for GO-LR (Habib et al., 2026b) and the metaheuristic ordering baselines. Rank-normalized and affinity-based variants can still be used as internal locality diagnostics, but the primary ordering comparison uses raw Euclidean TSP and raw Euclidean MinLA. The same LENS methodology can also be instantiated with other admissible feature-distance choices, including Manhattan distance, cosine distance, symmetric KL divergence, and correlation distance $1 - |r_{ij}|$, by replacing the Euclidean distance matrix with the corresponding metric-induced matrix.

B.2 PROPERTIES OF THE LENS INITIALIZATION AND REFINEMENT

Hamiltonian path construction. The global medoid nearest-neighbor initialization described in Section 3.1 returns a valid path containing every feature exactly once.

Lemma 1 (Hamiltonian path construction)**.** *The medoid nearest-neighbor construction returns a Hamiltonian path over the m features.*

Proof sketch. The algorithm starts from one feature and appends exactly one previously unvisited feature at each step. Since no feature is revisited and the process terminates only after all m features have been appended, the resulting sequence contains each feature exactly once. Hence, it is a Hamiltonian path over the complete feature set. □

Monotone TSP refinement. The first 2-opt refinement stage accepts only candidate reversals whose TSP change in Eq. 5 is negative.

Proposition 1 (Monotone TSP improvement)**.** *During the TSP 2-opt refinement phase, the TSP-path cost is monotonically non-increasing.*

Proof sketch. Every accepted 2-opt move satisfies $\Delta_{\mathrm{TSP}} < 0$ Rejected moves leave the ordering unchanged. Therefore, the TSP-path cost can only decrease or remain unchanged throughout the TSP refinement phase. □

Budget-feasible MinLA refinement. The second refinement stage explicitly reduces the MinLA objective while enforcing the self-calibrated TSP budget defined in Eq. 6.

Theorem 3 (Budget-feasible MinLA improvement)**.** *During the MinLA-budgeted refinement stage, every accepted move strictly decreases the MinLA objective and keeps the TSP cost within the budget B_{TSP}.*

Proof. A candidate move is accepted only if it satisfies Eqs. 16 and 17.

$$C_{\mathrm{MinLA}}(\Pi') < C_{\mathrm{MinLA}}(\Pi) \tag{16}$$

$$C_{\mathrm{TSP}}(\Pi') \leq B_{\mathrm{TSP}} \tag{17}$$

Therefore, every accepted move strictly decreases C_{MinLA} and yields an ordering whose TSP cost is no larger than B_{TSP}. Rejected moves leave the ordering unchanged. □

Corollary 1 (Local budgeted Pareto stability)**.** *When LENS terminates, no candidate move in the selected 2-opt neighborhood can reduce the MinLA objective while remaining within the TSP budget. Hence, the final ordering is locally stable under the budgeted TSP and MinLA criterion.*

Proof sketch. The refinement terminates only when no admissible candidate reversal satisfies both conditions in Eq. 7. Therefore, within the explored candidate neighborhood, no additional move can improve the MinLA objective without violating the TSP budget. □

B.3 Prefix-Accelerated MinLA Update

A naive recomputation of the complete MinLA objective after every candidate 2-opt reversal is expensive. LENS therefore uses an exact prefix-sum formulation to compute the MinLA change induced by a contiguous segment reversal. Let $W = D$ denote the MinLA weight matrix and let W^{Π} denote the matrix reordered according to the current ordering Π in Eq. 18. For every row u, define the prefix sums by Eq. 19.

$$W^{\Pi}_{uv} = W_{\Pi_u,\Pi_v} \tag{18}$$

$$P_{u,t} = \sum_{v=1}^{t} W^{\Pi}_{u,v}, \qquad T_u = P_{u,m} \tag{19}$$

Consider reversing the contiguous segment $[l, r]$ in the current ordering. Pairwise distances among two features both inside the segment are unchanged by the reversal, and pairwise distances among two features both outside the segment are also unchanged. Therefore, only feature pairs with one feature inside $[l, r]$ and the other outside the segment can change their MinLA contribution. The exact MinLA change induced by the reversal is defined by Eq. 20.

$$\Delta_{\mathrm{MinLA}}(l, r) = \sum_{p=l}^{r} \left[(l + r - 2p) \sum_{q<l} W^{\Pi}_{p,q} + (2p - l - r) \sum_{q>r} W^{\Pi}_{p,q} \right] \tag{20}$$

Using the prefix quantities in Eq. 19, the two sums required for each p can be recovered without repeatedly scanning the full feature set. Specifically, we get Eqs. 21 and 22.

$$\sum_{q<l} W^{\Pi}_{p,q} = P_{p,l-1} \tag{21}$$

$$\sum_{q>r} W^{\Pi}_{p,q} = T_p - P_{p,r} \tag{22}$$

Thus, after construction of the prefix matrix, the contribution of each position p in Eq. 20 is obtained in constant time. This provides an exact MinLA reversal delta while avoiding complete objective recomputation for every candidate move.

B.4 Tuning-Free Property

The selected LENS configuration uses fixed algorithmic constants $b_{\mathrm{TSP}} = 30$, $b_{\mathrm{MinLA}} = 4$, $\eta = 1$, and $R = 20$. These values are not selected using validation accuracy, validation loss, test performance, Optuna search, or downstream model feedback. The only data-adaptive quantity controlling the second refinement stage is the TSP budget in Eq. 6, which is computed from the median adjacent edge length of the TSP-refined ordering.

Proposition 2 (No validation-based tuning)**.** *LENS is tuning free with respect to downstream model performance: its ordering is determined by the input feature matrix, the feature-distance graph, and fixed algorithmic constants.*

Proof sketch. The feature graph D is computed directly from the standardized input matrix. The medoid start, nearest-neighbor path, distance-ordered 2-opt candidates, and MinLA-budgeted refinement are deterministic functions of D and the fixed algorithmic constants. The algorithm does not use validation accuracy, validation loss, test labels, or downstream predictive feedback. Hence, no validation-based tuning is performed. □

B.5 Rationale for Fixed LENS Search Budgets

The constants b_{TSP}, b_{MinLA}, η, and R control search breadth, admissible TSP relaxation, and refinement depth rather than downstream predictive behavior. Their roles can be characterized independently of labels or validation performance.

Proposition 3 (Bounded-search and scale-equivariant refinement)**.** *For fixed b_{TSP}, b_{MinLA}, and R, the candidate breadth and accepted MinLA refinement depth remain bounded independently of downstream predictive performance. Moreover, for any rescaling $D' = cD$ with $c > 0$, the TSP-budget acceptance rule is invariant when η is fixed.*

Proof. A TSP scan considers at most b_{TSP} candidate neighbors per feature, giving $O(mb_{\mathrm{TSP}})$ candidate examinations per scan. The MinLA stage analogously restricts each feature to b_{MinLA} candidates, while R directly bounds the number of accepted MinLA refinements. Thus, fixed candidate caps control the search breadth without reference to predictive performance. For $D' = cD$, every TSP path cost and adjacent edge length scale by c. Hence, we get Eq. 23.

$$B'_0 = cB_0, \qquad \mu'_{\mathrm{edge}} = c\mu_{\mathrm{edge}}, \qquad B'_{\mathrm{TSP}} = c\left(B_0 + \eta\mu_{\mathrm{edge}}\right) \tag{23}$$

Therefore, $C'_{\mathrm{TSP}}(\Pi') \le B'_{\mathrm{TSP}}$ if and only if $C_{\mathrm{TSP}}(\Pi') \le B_{\mathrm{TSP}}$. Thus, η is dimensionless and the refinement rule is invariant to uniform rescaling of the feature distance geometry. □

The selected values instantiate this bounded-search design. The TSP 2-opt change requires only a constant number of distance lookups, so $b_{\mathrm{TSP}} = 30$ permits a broader local candidate search. Exact MinLA evaluation is comparatively more expensive, motivating the smaller $b_{\mathrm{MinLA}} = 4$ candidate cap. Setting $\eta = 1$ gives the TSP relaxation the direct interpretation of one median adjacent-edge unit, while $R = 20$ limits the expensive MinLA stage to at most 20 accepted refinements. Accordingly, $(30, 4, 1, 20)$ is a fixed computational search-budget configuration, not a set of predictive hyperparameters selected from validation performance.

B.6 Computational Complexity

Dense pairwise graph construction over all features costs $O(nm^2)$, where n is the number of samples and m is the number of features. Once the distance matrix is available, the global medoid nearest-neighbor initialization costs $O(m^2)$. Constructing nearest-candidate lists using full sorting costs $O(m^2 \log m)$, although partial selection can reduce this cost. Let M_{TSP} denote the number of accepted TSP 2-opt moves. With fixed candidate cap b_{TSP}, the candidate-pruned TSP refinement requires approximately $O\left(M_{\mathrm{TSP}} m b_{\mathrm{TSP}}\right)$ up to the cost of restarting the candidate scan after accepted moves. For the MinLA-budgeted phase, LENS recomputes the ordered prefix matrix after each accepted move. With at most R accepted moves, candidate cap b_{MinLA}, and average reversal length $\bar{L}$, the corresponding cost is bounded by $O\left(R\left[m^2 + mb_{\mathrm{MinLA}}\bar{L}\right]\right)$. Since b_{TSP}, b_{MinLA}, and R are fixed constants in the selected implementation, practical runtime is dominated by feature graph construction and a bounded collection of deterministic local refinements.

B.7 Computational Hardness

LENS is an efficient approximation rather than an exact solver for the underlying combinatorial ordering problem. This subsection provides the complete argument for Theorem 1. The two components of the LENS ordering criterion correspond to established combinatorial optimization problems. The TSP component seeks an ordering satisfying Eq. 24.

$$\min_{\Pi \in S_m} C_{\mathrm{TSP}}(\Pi) = \min_{\Pi \in S_m} \sum_{t=1}^{m-1} D_{\Pi_t, \Pi_{t+1}} \tag{24}$$

where S_m denotes the set of all permutations of the m features. The MinLA component seeks Eq. 25.

$$\min_{\Pi \in S_m} C_{\mathrm{MinLA}}(\Pi) = \min_{\Pi \in S_m} \sum_{1 \le i < j \le m} D_{ij} \left|\mathrm{pos}_{\Pi}(i) - \mathrm{pos}_{\Pi}(j)\right| \tag{25}$$

Algorithm B.1 LENS: Layout-Enhanced Neighborhood Sequencing

Require: Tabular matrix $X \in \mathbb{R}^{n \times m}$
Ensure: Global feature ordering Π^*
Standardize columns of X and set $Z \leftarrow X^\top$
Construct feature-distance graph D using Eq. 2
$s^* \leftarrow \arg\min_i \sum_{j=1}^{m} D_{ij}$
$\Pi \leftarrow \mathrm{NNPath}(D, s^*)$
Build distance-ordered candidate lists with cap b_{TSP}
repeat
    changed $\leftarrow$ False
    for candidate 2-opt reversal Π' **do**
        if $C_{\mathrm{TSP}}(\Pi') < C_{\mathrm{TSP}}(\Pi)$ **then**
            $\Pi \leftarrow \Pi'$
            changed $\leftarrow$ True
            break
        end if
    end for
until changed $=$ False
$\Pi^{\mathrm{TSP}} \leftarrow \Pi$
$B_0 \leftarrow C_{\mathrm{TSP}}(\Pi^{\mathrm{TSP}})$
$\mu_{\mathrm{edge}} \leftarrow \mathrm{median}\{D_{\Pi_t^{\mathrm{TSP}}, \Pi_{t+1}^{\mathrm{TSP}}}\}_{t=1}^{m-1}$
$B_{\mathrm{TSP}} \leftarrow B_0 + \eta \mu_{\mathrm{edge}}$
Build distance-ordered candidate lists with cap b_{MinLA}
for $r = 1, \ldots, R$ **do**
    Compute ordered prefix sums for exact MinLA reversal deltas
    Find the candidate reversal Π' with the largest MinLA decrease subject to $C_{\mathrm{TSP}}(\Pi') \leq B_{\mathrm{TSP}}$
    if such a candidate exists **then**
        $\Pi \leftarrow \Pi'$
    else
        break
    end if
end for
$\Pi^* \leftarrow \Pi$
return Π^*

Proof of Theorem 1. If the LENS ordering objective is restricted to adjacent dissimilarity only, Eq. 24 becomes the TSP-path problem on a complete weighted graph, which is NP-hard (Karp, 2009). If the objective is restricted to weighted pairwise layout only, Eq. 25 becomes weighted Minimum Linear Arrangement, which is also NP-hard (Garey et al., 1974). Therefore, an exact optimizer for the general LENS ordering problem would solve these NP-hard special cases. Hence, exact optimization of the general LENS ordering objective is NP-hard.

Corollary 2 (Need for approximate ordering)**.** *Unless $P = NP$, there is no polynomial-time algorithm that exactly solves the general LENS ordering objective for all possible feature graphs. Therefore, deterministic approximation procedures such as TSP-first construction and budgeted MinLA refinement are required for practical high-dimensional optimization.*

B.8 Complete LENS Algorithm

Algorithm B.1 provides the complete LENS procedure. LENS therefore constructs a raw feature-distance graph over transposed feature profiles, initializes a global medoid nearest-neighbor path, improves adjacent locality using candidate-pruned TSP refinement, and then reduces the global MinLA objective under a self-calibrated TSP budget. The final procedure is deterministic, TSP-first, MinLA-aware, and independent of validation-based downstream optimization.

B.9 Ordering Solver Ranking

Since runtime, TSP cost, and MinLA cost have different numerical scales, we compare ordering solvers using a normalized aggregate reward. Let $\mathcal{S}$ denote the set of ordering solvers and Eq. 26 denotes the evaluation metrics.

$$\mathcal{M} = \{\text{Runtime}, \text{TSP}, \text{MinLA}\} \tag{26}$$

For solver $s \in \mathcal{S}$ and metric $j \in \mathcal{M}$, let $x_{s,j}$ denote the corresponding raw value. Since lower values are preferable for all three metrics, the normalized reward is defined by Eq. 27. The overall ordering score is defined by Eq. 28.

$$r_{s,j} = \frac{\max_{s' \in \mathcal{S}} x_{s',j} - x_{s,j}}{\max_{s' \in \mathcal{S}} x_{s',j} - \min_{s' \in \mathcal{S}} x_{s',j}} \tag{27}$$

$$\text{Score}(s) = \frac{1}{|\mathcal{M}|} \sum_{j \in \mathcal{M}} r_{s,j} \tag{28}$$

Solvers are ranked in decreasing order of $\text{Score}(s)$. Thus, the first column of Table 1 reports the rank induced by the normalized ordering score, where a higher score indicates a stronger joint trade-off among runtime, TSP cost, and MinLA cost.

C Additional Details for SCOPE

This section provides additional theoretical and implementation details for SCOPE, including the exact integer representation-budget construction, proof of the analytical budget, transition-partition edge cases and validity, deterministic principal-direction orientation, packed heterogeneous segment projection, computational complexity, tuning-free properties, and the complete algorithm. The main SCOPE methodology is presented in Section 3.2.

C.1 Analytical Token Budget

The feasible continuous representation budget is defined as $\widetilde{M} = \min\{M_0, m, B\}$. The final integer budget in Eq. 10 is selected from the two feasible integers adjacent to $\widetilde{M}$. The candidate set is defined by Eq. 29.

$$\mathcal{C}(\widetilde{M}) = \left\{ \max\left(1, \left\lfloor \widetilde{M} \right\rfloor\right), \ \min\left(m, B, \left\lceil \widetilde{M} \right\rceil\right) \right\} \tag{29}$$

Ties are resolved deterministically in favor of the smaller token budget. Since only the floor and ceiling of the feasible continuous minimizer are evaluated, Eq. 10 is computed in constant time after d_{eff} is available.

Proposition 4 (Optimal one-shot token budget)**.** *The objective in $J(M)$ (Eq. 10) is strictly convex for $M > 0$ and has the unique continuous minimizer $M_0 = \sqrt{m d_{\text{eff}}}$.*

Proof. The first derivative of $J(M)$ (Eq. 10) is defined by Eq. 30.

$$J'(M) = \frac{1}{m} - \frac{d_{\text{eff}}}{M^2} \tag{30}$$

The second derivative is defined by Eq. 31.

$$J''(M) = \frac{2 d_{\text{eff}}}{M^3} > 0 \tag{31}$$

Therefore, $J(M)$ is strictly convex for every $M > 0$. Setting $J'(M) = 0$ gives Eq. 32.

$$M^2 = m d_{\text{eff}} \tag{32}$$

Consequently, the unique continuous minimizer is defined by Eq. 33.

$$M_0 = \sqrt{m d_{\text{eff}}} \tag{33}$$

Since the objective is strictly convex, the optimal feasible integer is one of the two feasible integers adjacent to the continuous minimizer. Therefore, Eq. 10 gives the exact integer solution. □

Since $1 \leq d_{\text{eff}} \leq m$, the unconstrained budget satisfies Eq. 34.

$$d_{\text{eff}} \leq M_0 \leq m \tag{34}$$

Thus, before applying the downstream backbone limit, the one-shot token budget lies between the effective spectral dimension and the ambient feature dimension.

C.2 Transition Partition Edge Cases and Validity

The main methodology assumes $A_{\text{tot}} > 0$ when defining the transition-mass boundaries. When $A_{\text{tot}} = 0$, all adjacent transitions carry zero total transition mass and no transition-based boundary is preferred. In this case, the preliminary boundaries are defined uniformly by Eq. 35.

$$\widetilde{q}_t = \left\lfloor \frac{tm}{M^*} \right\rfloor \qquad t = 1, \ldots, M^* - 1 \tag{35}$$

The same feasibility correction in Eq. 13 is then applied to ensure that all final segments are nonempty.

Proposition 5 (Valid one-shot partition)**.** *The boundary correction in Eq. 13 produces M^* nonempty, disjoint, and contiguous segments whose union is the complete ordered feature axis.*

Proof. By construction, $q_0 = 0$ and $q_{M^*} = m$. For every $t = 1, \ldots, M^* - 1$, Eq. 13 enforces Eq. 36.

$$q_t \geq q_{t-1} + 1 \tag{36}$$

Therefore, the boundary sequence is strictly increasing and every segment in S_t (Eq. 14) contains at least one feature. The upper constraint is given by Eq. 37.

$$q_t \leq m - M^* + t \tag{37}$$

This reserves at least one feature for every remaining segment. Since consecutive segments start at $q_{t-1} + 1$ and end at q_t, they are disjoint and contiguous. Finally, the first segment starts at feature 1 and the final segment ends at feature m. Hence, Eq. 38 holds.

$$\bigcup_{t=1}^{M^*} S_t = \{1, \ldots, m\} \tag{38}$$

□

C.3 Structural Validity of the SCOPE Mapping

Proof of Theorem 2. By Proposition 4, $J(M)$ is strictly convex for $M > 0$, with continuous minimizer $M_0 = \sqrt{m d_{\text{eff}}}$, and Eq. 10 selects the optimal feasible integer budget M^*. By Proposition 5, the corresponding boundary correction constructs exactly M^* nonempty, disjoint, and contiguous segments whose union is the complete ordered feature axis. Each segment contributes one principal meta-feature $z_t(x)$. Therefore, Eq. 15 produces exactly one meta-feature per valid segment and hence an M^*-dimensional SCOPE representation, which establishes the stated structural validity. □

C.4 Deterministic Principal-Direction Orientation

The first right singular vector used in each segment is defined only up to sign. Although this ambiguity does not alter the represented one-dimensional principal subspace, deterministic orientation is desirable for reproducibility. Let v_t denote the first right singular vector of segment S_t. We define the index of its largest absolute loading by Eq. 39.

$$j_t^* = \arg\max_{1 \le j \le |S_t|} |(v_t)_j| \tag{39}$$

The sign-corrected direction is defined by Eq. 40.

$$v_t \leftarrow \operatorname{sign}\left((v_t)_{j_t^*}\right) v_t \tag{40}$$

This convention fixes the largest magnitude loading to have nonnegative orientation and therefore removes numerical sign variability across repeated executions.

C.5 Packed Heterogeneous Segment Projection

The final SCOPE segments generally have different widths. A serial implementation performs a separate decomposition for every segment and incurs repeated kernel-launch overhead with inefficient hardware utilization. Inspired by the heterogeneous packing strategy of TabPack (Gorishniy et al., 2026), SCOPE processes multiple variable-width segments simultaneously through padded and batched linear-algebra operations. Consider a pack containing q segments with widths $d_1, \ldots, d_q$. We define the maximum pack width by Eq. 41.

$$d_{\max} = \max_{1 \le t \le q} d_t \tag{41}$$

The centered segment matrices are zero-padded and stacked into $\mathcal{A} \in \mathbb{R}^{q \times n \times d_{\max}}$. For each packed segment t, the sample-space Gram matrix is defined by Eq. 42.

$$\mathcal{G}_t = \frac{1}{n-1} \mathcal{A}_t \mathcal{A}_t^\top \in \mathbb{R}^{n \times n} \qquad t = 1, \ldots, q \tag{42}$$

All matrices in Eq. 42 are computed using batched matrix multiplication. Let $\mathcal{G}_t u_t = \lambda_t u_t$ and $\|u_t\|_2 = 1$ denote the leading eigenpair for packed segment t. The corresponding right principal direction is recovered by Eq. 43.

$$v_t = \frac{\mathcal{A}_t^\top u_t}{\sqrt{(n-1)\lambda_t}} \tag{43}$$

For a degenerate segment with $\lambda_t = 0$, SCOPE returns a deterministic unit basis vector and the resulting token is identically zero after centering. Padded entries are removed after the principal direction is recovered. To reduce padding overhead, segments may be grouped into width buckets according to Eq. 44.

$$b(d_t) = 2^{\lceil \log_2 d_t \rceil} \tag{44}$$

Segments assigned to the same bucket are packed and processed together. This strategy does not alter the mathematical result of segment-wise PCA. It only improves computational efficiency.

C.6 Computational Complexity

Constructing the global Gram matrix as $G = \frac{1}{n-1} Y Y^\top \in \mathbb{R}^{n \times n}$ requires $\mathcal{O}(n^2 m)$ operations, while its eigendecomposition requires $\mathcal{O}(n^3)$ operations. The one-shot token budget in Eq. 10 is computed in $\mathcal{O}(1)$ time after the spectrum is available. The ordered transition sequence and cumulative transition mass require $\mathcal{O}(m)$ time. All $M^* - 1$ boundaries can be obtained in $\mathcal{O}(m + M^*)$ time through a single monotone scan. Since the final segments are disjoint, constructing all segment Gram matrices

Algorithm C.1 SCOPE: Self-Calibrated Ordered Principal Encoding

Require: Raw tabular matrix $X \in \mathbb{R}^{n \times m}$, backend limit B
Ensure: Compressed representation Z
1: $\Pi^* \leftarrow \mathrm{LENS}(X)$ using Algorithm B.1
2: $Y \leftarrow \mathrm{Standardize}(X)[:, \Pi^*]$
3: Compute d_{eff} from the positive eigenvalues of $YY^\top/(n-1)$
4: $M^* \leftarrow \arg\min_{M \in \mathcal{C}(\widetilde{M})} (M/m + d_{\mathrm{eff}}/M)$
5: Compute $\delta_j = D_{\Pi_j^*, \Pi_{j+1}^*}$ and cumulative transition mass
6: Construct $\mathcal{P}^* = \{S_1, \ldots, S_{M^*}\}$ using one-shot equal transition-mass boundaries
7: $\{(\mu_t, v_t)\}_{t=1}^{M^*} \leftarrow \mathrm{PackedSegPCA}(Y, \mathcal{P}^*)$
8: $Z_{:,t} \leftarrow \left(Y_{:,S_t} - \mathbf{1}_n \mu_t^\top\right) v_t$
9: **return** Z

requires a total of $\mathcal{O}(n^2 m)$ operations. Exact eigendecomposition of the M^* segment Gram matrices requires $\mathcal{O}(M^* n^3)$ operations in the worst case, although packed execution reduces practical wall-clock time through batched hardware utilization. Therefore, the overall fitting complexity is defined by $\mathcal{O}\left(n^2 m + n^3 + M^* n^3\right)$. This matches the principal computational structure of a single NSC-pSP (Habib et al., 2026b) fit while removing recursive intermediate segment decompositions and repeated hyperparameter trials. After fitting, transforming one sample examines every original feature exactly once and therefore requires $\mathcal{O}(m)$ time while producing $\mathcal{O}(M^*)$ output values.

C.7 Tuning-Free Property

SCOPE does not use class labels, validation accuracy, downstream predictions, or hyperparameter search to determine its compression configuration. The effective rank, token count, and segment boundaries are derived exclusively from the training fold feature geometry. In particular, SCOPE removes the manually selected cumulative variance threshold τ, budget multipliers such as γ and β, dataset-specific lower and upper token limits, segmentation rule selection, minimum segment-length tuning, and manually supplied token count M. Only the downstream architectural capacity B remains externally specified. Consequently, SCOPE preserves the ordered, contiguous, segment-wise principal-subspace compression principle of NSC-pSP (Habib et al., 2026b) while providing a deterministic, one-shot, efficient, and self-calibrated compression pipeline.

C.8 Complete SCOPE Algorithm

Algorithm C.1 summarizes the complete SCOPE pipeline.

D Theoretical Properties and Calibration of the Dataset Regime Taxonomy

The categorization introduced in DynaTab (Habib et al., 2026e) and adopted by GOTabPFN (Habib et al., 2026b) is an empirical stratification of tabular datasets. This section establishes mathematical properties of that stratification and its Micro-Macro refinement. We distinguish distribution free properties of matrix shape and median calibration from statistical guarantees that require a specified data model. In particular, the absolute cutoffs 100, 1000, and 10^4 remain benchmark conventions; the theoretical results obtained in this section do not identify them as universal statistical phase transitions or prove that they maximize feature ordering gains. The linear algebra, Gaussian regression, and empirical process results used here are standard tools, applied to clarify the scope of the taxonomy.

Original rule and well-definedness. Let $\mathcal{D}$ have $n \geq 2$ samples and $m \geq 1$ input features, excluding the prediction target. We define Eq. 45.

$$\rho(\mathcal{D}) = \frac{m}{n}, \qquad \kappa(\mathcal{D}) = nm \tag{45}$$

The original assignment is defined in Eq. 46 by Habib et al. (2026e;b).

$$\mathcal{R}_\rho(\mathcal{D}) = \begin{cases} \text{HDLSS}, & m > 1000,\ n < 1000,\ \rho > 2 \\ \text{HDHSS}, & m > 1000,\ n > 10^4,\ 0.005 < \rho \le 2 \\ \text{LDHSS}, & m \le 100,\ n > 10^4,\ \rho \le 0.01 \\ \text{LDLSS}, & m \le 100,\ n \le 1000,\ \rho \le 0.05 \\ \text{MixedRegime}, & \text{otherwise} \end{cases} \tag{46}$$

Here, the subscript ρ is inherited notation, the assignment also depends on the absolute values of n and m.

Proposition D.1 (Partition property)**.** *Equation 46 assigns every admissible dataset to exactly one category.*

Proof sketch. The two high-dimensional categories require $m > 1000$, whereas the two low-dimensional categories require $m \le 100$; these sets do not intersect. Within each pair, the low and high-sample conditions are disjoint. Thus the first four cases are pairwise disjoint. MixedRegime is their complement, so the five cases exhaust the domain. □

Remark D.2 (Logical redundancy of some constraints). For LDHSS, $m \le 100$ and $n > 10^4$ already imply $\rho < 0.01$. For LDLSS, $\rho \le 0.05$ and $n \le 1000$ imply $m \le 50$, making the written condition $m \le 100$ redundant. We retain the original inequalities for comparability with earlier work. These observations also show why each numerical condition should not be presented as an independently derived theoretical boundary.

A distribution-free consequence of the HDLSS ratio. Let $X \in \mathbb{R}^{n\times m}$ be the observed feature matrix and let us have Eq. 47.

$$X_c = \left(I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right) X, \qquad \widehat{\Sigma} = \frac{X_c^\top X_c}{n-1} \tag{47}$$

Theorem D.3 (Empirical rank deficit)**.** *For every X, we get Eq. 48.*

$$\operatorname{rank}(\widehat{\Sigma}) \le \min\{n-1, m\}, \qquad \frac{\dim\ker(\widehat{\Sigma})}{m} \ge \left[1 - \frac{n-1}{m}\right]_+ \tag{48}$$

where $[a]_+ = \max\{a, 0\}$. Consequently, every dataset assigned to HDLSS by Eq. 46 satisfies Eq. 49.

$$\dim\ker(\widehat{\Sigma}) > \frac{m}{2} \tag{49}$$

If $\operatorname{tr}(\widehat{\Sigma}) > 0$, let $\widehat{d}_\tau$ be the minimum number of principal components that explain at least a fraction $\tau \in (0, 1]$ of the empirical variance. Then we get Eq. 50.

$$\frac{\widehat{d}_\tau}{m} \le \min\left\{1, \frac{n-1}{m}\right\} \tag{50}$$

Here, Eq. 50 is strictly less than $1/2$ in the HDLSS category.

Proof sketch. The centering matrix in Eq. 47 has rank $n-1$, and $\operatorname{rank}(X_c^\top X_c) = \operatorname{rank}(X_c)$. Rank-nullity gives Eq. 48. The HDLSS condition $\rho > 2$ implies $m > 2n$, hence $1 - (n-1)/m > 1/2$. All nonzero empirical variance is contained in the $\operatorname{rank}(\widehat{\Sigma})$ positive-eigenvalue directions, so $\widehat{d}_\tau \le \operatorname{rank}(\widehat{\Sigma})$. □

What the ratio threshold certifies. The condition $\rho > 2$ is therefore a sufficient certificate of a greater than one half empirical null space fraction. It is not the unique threshold with this interpretation. For any desired fraction $\omega \in (0, 1)$, the exact sufficient condition is defined by Eq. 51.

$$m > \frac{n-1}{1-\omega}; \qquad \rho > \frac{1}{1-\omega} \quad \text{is a simpler sufficient rule} \tag{51}$$

Taking $\omega = 1/2$ recovers $\rho > 2$ as a conservative choice. This supplies a mathematical interpretation of that ratio without deriving the additional $m > 1000$ or $n < 1000$ conditions.

Empirical IDF curves. A low empirical IDF (e.g. DynaTab (Habib et al., 2026e)) can partly result from the rank ceiling imposed by a small sample size. Even an isotropic population with full rank covariance obeys Eq. 50 when sampled with $n \ll m$. Thus empirical PCA compactness alone does not establish low population intrinsic dimension or useful feature modules. Moreover, a feature permutation P replaces $\widehat{\Sigma}$ by $P^\top \widehat{\Sigma} P$, preserving all eigenvalues. Ordering can make related features contiguous for a locality-sensitive operator; it cannot itself change empirical rank or the PCA variance spectrum.

A model-dependent statistical interpretation. To quantify a role for the feature to sample ratio, consider a canonical Gaussian linear model. The result is an illustrative statistical guarantee under explicit assumptions, not a performance bound for DynaTab (Habib et al., 2026e), GOTabPFN (Habib et al., 2026b), or an arbitrary tabular model.

Theorem D.4 (Finite sample Gaussian prediction risk)**.** *Suppose the rows of an uncentered design matrix X are independent $\mathcal{N}(0, I_m)$ vectors, and we get Eq. 52.*

$$y = X\mathbf{w}_{\text{lin}} + \varepsilon, \qquad \mathbb{E}[\varepsilon \mid X] = 0, \qquad \mathbb{E}[\varepsilon\varepsilon^\top \mid X] = \sigma^2 I_n \tag{52}$$

where $\mathbf{w}_{\text{lin}} \in \mathbb{R}^m$ is fixed and $\sigma^2 > 0$. No intercept or sample-dependent standardization is fitted in this model. For an independent $x_{\text{new}} \sim \mathcal{N}(0, I_m)$, define the expected excess prediction risk by Eq. 53 with expectation over training data, noise, and the test covariate. For ordinary least squares when $n > m + 1$, we get Eq. 54.

$$\mathcal{E}_{n,m} = \mathbb{E}\big[(x_{\text{new}}^\top \widehat{\mathbf{w}}_{\text{lin}} - x_{\text{new}}^\top \mathbf{w}_{\text{lin}})^2\big] \tag{53}$$

$$\mathcal{E}_{n,m} = \sigma^2 \frac{m}{n-m-1} = \sigma^2 \frac{\rho}{1-\rho-1/n} \tag{54}$$

For the minimum-norm interpolator we define Eq. 55. $\widehat{\mathbf{w}}_{\text{lin}} = X^\top (XX^\top)^{-1} y$ when $m > n + 1$,

$$\mathcal{E}_{n,m} = \left(1 - \frac{n}{m}\right) \|\mathbf{w}_{\text{lin}}\|_2^2 + \sigma^2 \frac{n}{m-n-1} \tag{55}$$

For the corresponding least-squares solution at $m \in \{n-1, n, n+1\}$, the expected noise contribution is infinite.

Proof. Isotropy of x_{new} gives $\mathcal{E}_{n,m} = \mathbb{E}\|\widehat{\mathbf{w}}_{\text{lin}} - \mathbf{w}_{\text{lin}}\|_2^2$. We first record a Gaussian Gram matrix identity. If $Q \in \mathbb{R}^{a \times b}$ has independent standard normal entries and $a > b+1$, then we define Eq. 56.

$$\mathbb{E}\,\mathrm{tr}\left((Q^\top Q)^{-1}\right) = \frac{b}{a-b-1} \tag{56}$$

To see this, condition on all columns except q_j. The block-inverse formula via the Schur complement Zhang (2005) gives Eq. 57.

$$[(Q^\top Q)^{-1}]_{jj} = \left[q_j^\top (I - H_{-j}) q_j\right]^{-1} \tag{57}$$

where H_{-j} projects onto the other $b-1$ columns. The denominator is χ^2_{a-b+1}. For $U \sim \chi^2_\nu$, direct integration of its density gives Eq. 58 and the integral diverges for $\nu \le 2$. Summing the diagonal expectations proves Eq. 56..

$$\mathbb{E}[U^{-1}] = \frac{\Gamma(\nu/2 - 1)}{2\Gamma(\nu/2)} = \frac{1}{\nu - 2} \quad (\nu > 2) \tag{58}$$

For ordinary least squares, $\widehat{\mathbf{w}}_{\text{lin}} - \mathbf{w}_{\text{lin}} = (X^\top X)^{-1} X^\top \varepsilon$. Its conditional squared error expectation is $\sigma^2\,\mathrm{tr}((X^\top X)^{-1})$. Apply Eq. 56 with $(a, b) = (n, m)$. For $m > n + 1$, let $H_X = X^\top (XX^\top)^{-1} X$. Then we get Eq. 59.

$$\widehat{\mathbf{w}}_{\text{lin}} - \mathbf{w}_{\text{lin}} = -(I_m - H_X)\mathbf{w}_{\text{lin}} + X^\top (XX^\top)^{-1} \varepsilon \tag{59}$$

The two terms lie in orthogonal subspaces. Rotational invariance of the Gaussian row space implies $\mathbb{E}[H_X] = (n/m) I_m$, so the expected squared norm of the first term is $(1 - n/m)\|\mathbf{w}_{\text{lin}}\|_2^2$. The conditional squared norm of the second term has expectation $\sigma^2\,\mathrm{tr}((XX^\top)^{-1})$. Apply Eq. 56 to $X^\top$. The stated divergence at the interpolation boundary follows from the same inverse-χ^2 calculation. These calculations are consistent with standard analyses of ridgeless regression (Hastie et al., 2022). □

Corollary D.5 (Consequences for the low-dimensional labels)**.** *Under the assumptions of Theorem D.4, every LDHSS dataset satisfies* $\mathcal{E}_{n,m}/\sigma^2 \leq 1/99$*, and every LDLSS dataset satisfies* $\mathcal{E}_{n,m}/\sigma^2 \leq 1/18$*. The HDHSS label alone does not imply a uniformly finite expected least-squares risk.*

Proof. For LDHSS, integer-valued dimensions give $n \geq 10001$ and $m \leq 100$. Eq. 54 is largest at the largest m and smallest n, giving $100/(10001 - 100 - 1) = 1/99$. For LDLSS, $\rho \leq 0.05$ and $m \geq 1$ imply $n \geq 20$, $n > m + 1$, and $1/n \leq \rho$. Hence we get Eq. 60.

$$\frac{\mathcal{E}_{n,m}}{\sigma^2} = \frac{\rho}{1 - \rho - 1/n} \leq \frac{\rho}{1 - 2\rho} \leq \frac{0.05}{0.90} = \frac{1}{18} \tag{60}$$

Finally, HDHSS includes $n = m = 10001$, where the expected noise contribution diverges by Theorem D.4. □

How a statistical tolerance determines a cutoff. For an excess-risk budget $\xi\sigma^2$, with $\xi > 0$, Eq. 54 yields the exact equivalence by Eq. 61.

$$\mathcal{E}_{n,m} \leq \xi\sigma^2 \quad \Longleftrightarrow \quad m \leq \frac{\xi}{1+\xi}(n-1), \qquad n > m + 1 \tag{61}$$

Thus even in this simple model, a principled dimensionality cutoff depends on the sample size and the chosen error tolerance. The numerical bounds in Corollary D.5 are consequences of the existing rule; they do not establish that its thresholds are optimal. The Gaussian formulas do not transfer unchanged to correlated, heavy-tailed, dependent, or nonlinear data.

Why an absolute threshold of 1000 is not universal?

Proposition D.6 (Shape does not determine statistical structure)**.** *The values* (n, m) *alone do not determine the population effective rank or certify a positive benefit from feature ordering. In particular, a dataset can cross* $m = 1000$ *without changing its predictive information.*

Proof sketch. Fix n and $m \geq 2$. The populations $x \sim \mathcal{N}(0, I_m)$ and $x = h\mathbf{1}_m$ with $h \sim \mathcal{N}(0, 1)$ have covariance effective ranks m and 1, respectively, using $r_{\text{eff}}(\Sigma) = \text{tr}(\Sigma)/\|\Sigma\|_{\text{op}}$. Both have the same feature count and admit samples with the same number of rows. More explicitly, fix $n = 200$ and any prediction problem (h, y). For each m, represent the input by $x^{(m)} = h\mathbf{1}_m/\sqrt{m}$. The mapping is one to one in h, so the predictive information and Bayes risk are identical for all m. For a realized vector of observations $\mathbf{h} = (h_1, \ldots, h_n)^\top$, its data matrix satisfies $X^{(m)}(X^{(m)})^\top = \mathbf{h}\mathbf{h}^\top$ for every m. Nevertheless, Eq. 46 assigns $m = 999$ to MixedRegime and $m = 1001$ to HDLSS. Every column permutation leaves each of these matrices unchanged. Therefore no strictly positive ordering gain can be guaranteed from these dimensions alone. □

Status of the original absolute limits. The limits $m > 1000$ and $n < 1000$ identify an operational subset of feature-rich, sample-poor datasets. The linear algebraic rank boundary is $m > n - 1$ for centered covariance, while the Gaussian risk calculation depends on the interpolation region around $m = n$ and on a specified tolerance. Neither singles out 1000 independently of n. A fixed model's feature cap can justify an implementation limit, and an explicit resource budget can justify a computational limit, but neither is a universal statistical law.

Separating shape and matrix scale. BeyondArena (Purucker et al., 2026) motivates examining dataset scale in addition to feature dimensionality and sample count. The pair (ρ, κ) preserves the information in (n, m) while separating aspect ratio from the number of matrix entries.

Proposition D.7 (Invertible shape-scale coordinates)**.** *For positive* n, m*, the map* $(n, m) \mapsto (\rho, \kappa)$ *is invertible, with Eq. 62.*

$$n = \sqrt{\frac{\kappa}{\rho}}, \qquad m = \sqrt{\kappa\rho}. \tag{62}$$

In logarithmic coordinates, the normalized variables $\ell_\rho = (\log\rho)/\sqrt{2}$ *and* $\ell_\kappa = (\log\kappa)/\sqrt{2}$ *form an orthogonal change of coordinates from* $(\log n, \log m)$.

Proof sketch. Multiplying and dividing the definitions in Eq. 45 gives Eq. 62. Also, we get Eq. 63 and the displayed matrix, including its factor $1/\sqrt{2}$, is orthogonal. This is geometric orthogonality, not a claim that ρ and κ are statistically independent.

$$\begin{pmatrix} \ell_\rho \\ \ell_\kappa \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \log n \\ \log m \end{pmatrix} \tag{63}$$

□

Computational interpretation and its scope. Reading a dense matrix once involves $\Theta(nm) = \Theta(\kappa)$ entries, and storing its entries using b_{elem} bytes each requires $b_{\text{elem}}\kappa$ bytes, excluding overhead. Standard direct computation of all feature pair dot products requires $\Theta(nm^2)$ arithmetic operations and, if the full affinity matrix is retained, $\Theta(m^2)$ entries. The relevant identities are defined by Eq. 64.

$$nm^2 = \kappa^{3/2}\rho^{1/2}, \qquad m^2 = \kappa\rho, \qquad n^2 m = \kappa^{3/2}\rho^{-1/2} \tag{64}$$

These are costs of specified dense operations, not lower bounds for every possible implementation. Sparsity, approximation, streaming, architecture, and hardware can change actual costs. At fixed ρ, larger κ increases all of these costs; across different ρ, cell count alone does not order them. For example, storing the raw matrix within B_{mem} bytes requires $\kappa \le B_{\text{mem}}/b_{\text{elem}}$, whereas storing a dense feature affinity matrix requires $\kappa\rho \le B_{\text{mem}}/b_{\text{elem}}$, before accounting for other memory.

Micro-Macro refinement and deterministic guarantees. Let $\mathfrak{C}$ be the five original categories. Fix a reference collection $\mathfrak{B}$, and for $g \in \mathfrak{C}$, we define Eq. 65.

$$\mathfrak{B}_g = \{\mathcal{D}_i \in \mathfrak{B} : \mathcal{R}_\rho(\mathcal{D}_i) = g\}, \qquad N_g = |\mathfrak{B}_g| \tag{65}$$

Assume $N_g \ge 1$ for every category to which the refinement is applied. Let $\kappa_{g,(1)} \le \cdots \le \kappa_{g,(N_g)}$ be the sorted cell counts. To make the median convention explicit, we define Eq. 66.

$$\kappa_g^{\text{med}} = \begin{cases} \kappa_{g,((N_g+1)/2)}, & N_g \text{ odd} \\ \big(\kappa_{g,(N_g/2)} + \kappa_{g,(N_g/2+1)}\big)/2, & N_g \text{ even} \end{cases} \tag{66}$$

For $g = \mathcal{R}_\rho(\mathcal{D})$, we set conditions as per Eqs. 67 and 68.

$$\psi(\mathcal{D}) = \log \frac{\kappa(\mathcal{D})}{\kappa_g^{\text{med}}}, \qquad \text{Scale}(\mathcal{D}) = \begin{cases} \text{Micro}, & \psi(\mathcal{D}) < 0 \\ \text{Macro}, & \psi(\mathcal{D}) \ge 0 \end{cases} \tag{67}$$

$$\mathcal{R}_{\kappa,\rho}(\mathcal{D}) = \big(\text{Scale}(\mathcal{D}), \mathcal{R}_\rho(\mathcal{D})\big) \tag{68}$$

This gives ten possible labels when all five κ_g^{med} are defined; the ten cells need not all be populated. For an empty reference category, its median is undefined, so a documented external calibration is needed before assigning its Micro-Macro label.

Proposition D.8 (Median properties)**.** *For each nonempty reference category:*

1. κ_g^{med} *minimizes* $\sum_{i:\mathcal{D}_i \in \mathfrak{B}_g} |\kappa_i - t|$ *over* $t \in \mathbb{R}$.
2. *At most* $\lfloor N_g/2 \rfloor$ *reference datasets are Micro and at least* $\lceil N_g/2 \rceil$ *are Macro. With distinct cell counts the two sizes are exactly these values.*
3. *If fewer than* $N_g/2$ *reference cell counts are replaced arbitrarily, the resulting median remains between the smallest and largest unreplaced counts.*
4. *Multiplying all reference counts and a target count by the same* $c > 0$ *multiplies* κ_g^{med} *by* c *and preserves the target's Micro-Macro label, provided original category memberships are fixed.*

Proof sketch. For $L(t) = \sum_i |\kappa_i - t|$, its subgradient contains zero exactly when no more than half of the observations lie strictly on either side of t. These are precisely the sample medians, and Eq. 66 selects one of them. This also proves the size bounds. With distinct counts, an odd-sample median is the central observation and an even-sample median is strictly between the two central observations, giving the exact sizes. After fewer than half the counts are replaced, a strict majority remain in the interval spanned by the unreplaced observations; both central order statistics must stay in that interval. Finally, multiplying by $c > 0$ preserves order and scales the arithmetic median by c, leaving $\kappa/\kappa_g^{\text{med}}$ unchanged. □

Ties and interpretation. The tie rule assigns $\kappa = \kappa_g^{\mathrm{med}}$ to Macro. If all reference counts are equal, all reference datasets are therefore Macro; an exactly balanced split cannot be asserted in the presence of ties. Median optimality concerns absolute deviation in scale, not prediction accuracy or runtime. Also, the rescaling property is conditional on fixed original memberships: rescaling n and m can cross the absolute boundaries in Eq. 46.

Proposition D.9 (Stability under multiplicative perturbations)**.** *Keep reference and target category memberships fixed. Suppose each reference count and the target count are perturbed so that we get the relationship as per Eq. 69.*

$$e^{-a}\kappa_i \le \kappa_i' \le e^{a}\kappa_i, \qquad e^{-a}\kappa \le \kappa' \le e^{a}\kappa, \qquad a \ge 0 \tag{69}$$

Then $e^{-a}\kappa_g^{\mathrm{med}} \le (\kappa_g^{\mathrm{med}})' \le e^{a}\kappa_g^{\mathrm{med}}$ *and we get Eq. 70.*

$$|\psi'(\mathcal{D}) - \psi(\mathcal{D})| \le 2a \tag{70}$$

In particular, a target with $|\psi(\mathcal{D})| > 2a$ *retains its label.*

Proof sketch. Every order statistic, and hence the arithmetic median, is coordinatewise nondecreasing in the input counts. Apply this monotonicity to the two componentwise inequalities and use positive homogeneity of the median. Taking logarithms bounds both $|\log(\kappa'/\kappa)|$ and $|\log((\kappa_g^{\mathrm{med}})'/\kappa_g^{\mathrm{med}})|$ by a; their difference changes ψ by at most $2a$. □

This guarantee requires no random sampling model and therefore applies to a fixed curated benchmark. It does not cover changes that move datasets between original categories.

Conditional finite sample calibration guarantees. For a population interpretation of κ_g^{med}, an additional assumption is necessary: conditional on category g, the reference datasets are independent draws from a specified population of datasets. The original categorization rule must be fixed before this calibration sample is drawn. Independence of rows within a dataset does not imply independence or representativeness of datasets in a benchmark. Write $L_\kappa = \log \kappa$, let F_g be its population CDF within category g, and let $\widehat{F}_g$ be its empirical CDF from N_g reference datasets. Write $F_g(\ell^-) = \Pr(L_\kappa < \ell)$. Although the arithmetic median does not generally commute with a nonlinear transformation, $\widehat{\vartheta}_g = \log \kappa_g^{\mathrm{med}}$ is a median of the log counts: it lies between their two central order statistics. We do not replace Eq. 66 by $\exp(\mathrm{median}(\log \kappa_i))$.

Theorem D.10 (Median calibration)**.** *Assume the dataset sampling model above. Fix* $\alpha_{\mathrm{cal}} \in (0,1)$ *and let* K_{reg} *be the number of calibrated, nonempty categories (*$K_{\mathrm{reg}} = 5$ *for the complete refinement). Set Eq. 71.*

$$\varepsilon_g^{\mathrm{cal}} = \sqrt{\frac{\log(2K_{\mathrm{reg}}/\alpha_{\mathrm{cal}})}{2N_g}} \tag{71}$$

With probability at least $1 - \alpha_{\mathrm{cal}}$ *over the reference collection, simultaneously for all calibrated categories as per Eq. 72.*

$$F_g(\widehat{\vartheta}_g) \ge \frac{1}{2} - \varepsilon_g^{\mathrm{cal}}, \qquad F_g(\widehat{\vartheta}_g^-) \le \frac{1}{2} + \varepsilon_g^{\mathrm{cal}}. \tag{72}$$

Define the largest population atom by Eq. 73.

$$a_g^{\mathrm{atom}} = \sup_{\ell} \{F_g(\ell) - F_g(\ell^-)\} \tag{73}$$

Conditional on a reference collection in this event, an independent new dataset from category g *satisfies Eq. 74.*

$$\left| \Pr\left(\mathrm{Scale}(\mathcal{D}) = \mathrm{Micro} \mid \mathfrak{B}_g\right) - \frac{1}{2} \right| \le \varepsilon_g^{\mathrm{cal}} + a_g^{\mathrm{atom}} \tag{74}$$

Furthermore, let $\vartheta_g^\star$ *be any fixed population median and let* $\mathrm{Scale}^\star$ *use the threshold* $\exp(\vartheta_g^\star)$ *with the same tie rule. Then we get Eq. 75.*

$$\Pr\left(\mathrm{Scale}(\mathcal{D}) \ne \mathrm{Scale}^\star(\mathcal{D}) \mid \mathfrak{B}_g\right) \le \min\{1, \varepsilon_g^{\mathrm{cal}} + a_g^{\mathrm{atom}}\} \tag{75}$$

Proof. The Dvoretzky-Kiefer-Wolfowitz (DKW) inequality (Massart, 1990) gives Eq. 76, for any CDF including a discrete one,

$$\Pr\left(\sup_{\ell}\left|\widehat{F}_g(\ell) - F_g(\ell)\right| > \varepsilon_g^{\mathrm{cal}}\right) \le 2e^{-2N_g\left(\varepsilon_g^{\mathrm{cal}}\right)^2} = \frac{\alpha_{\mathrm{cal}}}{K_{\mathrm{reg}}} \tag{76}$$

A union bound gives simultaneous uniform control with probability at least $1 - \alpha_{\mathrm{cal}}$. The same control holds for left limits. Every sample median satisfies $\widehat{F}_g(\widehat{\vartheta}_g) \ge 1/2$ and $\widehat{F}_g(\widehat{\vartheta}_g^-) \le 1/2$, which proves Eq. 72. Since the jump at any point is at most a_g^{atom}, we get Eq. 77.

$$\frac{1}{2} - \varepsilon_g^{\mathrm{cal}} - a_g^{\mathrm{atom}} \le F_g(\widehat{\vartheta}_g^-) \le \frac{1}{2} + \varepsilon_g^{\mathrm{cal}} \tag{77}$$

The middle quantity is the conditional probability of Micro, proving Eq. 74. For a population median, $F_g((\vartheta_g^\star)^-) \le 1/2 \le F_g(\vartheta_g^\star)$. If $\widehat{\vartheta}_g < \vartheta_g^\star$, the disagreement probability is $F_g((\vartheta_g^\star)^-) - F_g(\widehat{\vartheta}_g^-) \le \varepsilon_g^{\mathrm{cal}} + a_g^{\mathrm{atom}}$. If $\widehat{\vartheta}_g > \vartheta_g^\star$, it is $F_g(\widehat{\vartheta}_g^-) - F_g((\vartheta_g^\star)^-) \le \varepsilon_g^{\mathrm{cal}} + a_g^{\mathrm{atom}}$, using $F_g((\vartheta_g^\star)^-) \ge 1/2 - a_g^{\mathrm{atom}}$. It is zero if the thresholds coincide. Probabilities are also bounded by one. □

Discreteness and sample size. Actual cell counts are integers, so the atom term cannot generally be dropped. Under a continuous scale model it is zero, yielding the simpler bound $\varepsilon_g^{\mathrm{cal}}$; that is an additional idealization. The confidence bounds concern median calibration and label stability, not feature ordering utility. They are informative only when the right hand sides are sufficiently small. For example, with $K_{\mathrm{reg}} = 5$ and $\alpha_{\mathrm{cal}} = 0.05$, obtaining $\varepsilon_g^{\mathrm{cal}} \le 0.1$ requires Eq. 78 and reference datasets per category. A reference suite with 36 datasets in total used in DynaTab (Habib et al., 2026e) does not yield this precision through Eq. 71. This calculation concerns the stated bound, not an information theoretic minimum for every possible calibration procedure. Cross-validation folds and repeated runs of the same dataset do not increase N_g. For a curated collection without the stated sampling model, only the deterministic guarantees above apply automatically (Eq. 78).

$$N_g \ge \left\lceil \frac{\log(200)}{2(0.1)^2} \right\rceil = 265 \tag{78}$$

Relation to computational variant selection. The Micro-Macro label describes matrix volume relative to a category's reference median. It does not, by itself, certify that which variant is faster or more accurate. A conditional computational guarantee requires a cost model. For example, suppose a matched comparison has a shared preprocessing cost C_{shared}, a full ordering cost $c_{\mathrm{ord}} m^{p_{\mathrm{ord}}}$ with $c_{\mathrm{ord}} > 0$, $p_{\mathrm{ord}} > 1$, and a clustered ordering cost $C_{\mathrm{ovh}} + c_{\mathrm{ord}} \sum_{j=1}^{K_{\mathrm{blk}}} m_j^{p_{\mathrm{ord}}}$, where $\sum_j m_j = m$ and $C_{\mathrm{ovh}} \ge 0$ includes additional clustering and merging overhead. Then we get Eq. 79 and clustered ordering is faster under this model exactly when it satisfies Eq. 80.

$$C_{\mathrm{full}} = C_{\mathrm{shared}} + c_{\mathrm{ord}} m^{p_{\mathrm{ord}}}, \qquad C_{\mathrm{clustered}} = C_{\mathrm{shared}} + C_{\mathrm{ovh}} + c_{\mathrm{ord}} \sum_{j=1}^{K_{\mathrm{blk}}} m_j^{p_{\mathrm{ord}}} \tag{79}$$

$$C_{\mathrm{ovh}} < c_{\mathrm{ord}} \left(m^{p_{\mathrm{ord}}} - \sum_{j=1}^{K_{\mathrm{blk}}} m_j^{p_{\mathrm{ord}}} \right) \tag{80}$$

For equal size blocks, $\sum_j m_j^{p_{\mathrm{ord}}} = m^{p_{\mathrm{ord}}} / K_{\mathrm{blk}}^{p_{\mathrm{ord}}-1}$. Convexity shows that this is the smallest possible sum for a fixed K_{blk}; imbalance increases it. Any shared construction of a full affinity graph remains in C_{shared} and is not removed by partitioning the later ordering problem. These statements follow directly from the specified cost model and do not establish that a benchmark median κ_g^{med} coincides with the runtime crossover of actual implementations. Such a crossover requires implementation-specific cost measurement; downstream utility requires a separate comparison.

Reporting and validation needed for an empirical cutoff. The guarantees above support the mathematical interpretation of the taxonomy while leaving the absolute cutoffs empirically calibrated. To substantiate these cutoffs in a new benchmark, the accompanying evaluation should document the following. First, report the representation at which m is counted, the reference dataset

identifiers, N_g, κ_g^{med}, and the tie convention. For model-dependent guarantees, n is the number of independent training observations actually used; a dataset-level reporting label based on the full dataset need not equal a fold-level label. Second, freeze reference thresholds before assessing their ability to select variants on new datasets. If only the original suite is available, "leave one dataset out" calibration can assess sensitivity, but it does not create independent external evidence or remove the original cutoff-selection bias. Third, perturb the absolute and ratio cutoffs over a prespecified grid and report label changes together with paired changes in utility and computational cost. Finally, compare datasets near the 1000 boundary while controlling aspect ratio, signal structure, and computational budget. These are proposed validation steps, not additional experimental results asserted by this section.

Claim supported by this analysis. The original taxonomy defined by Habib et al. (2026e;b) has a well-defined partition and explicit rank-based consequences; its low-ratio categories admit statistical interpretations under a stated Gaussian model. The Micro-Macro extension preserves the original category and adds a median-based measure of relative matrix scale, with deterministic robustness and conditional finite sample calibration bounds. The absolute thresholds and the association between a regime and a particular algorithm's empirical advantage remain matters for calibration and validation.

Connection to the main paper. Following Sec. 3.3 in the main paper, Fig. D.1 summarizes the key implications of this analysis. This appendix provides the complete theoretical development underlying these results, including the formal assumptions, propositions, proofs, robustness analysis, and finite sample calibration guarantees for the original dataset taxonomy and its Micro-Macro refinement.

E HDLSS-UCBT: Uncertainty-Calibrated Pairwise Ranking

Motivation. Average rank is widely used to summarize predictive performance across multiple datasets because every dataset contributes equally and the statistic is invariant to the numerical scale of the evaluation metric. However, average rank depends only on the ordering of point estimates and therefore cannot distinguish a stable improvement from an unstable improvement of the same rank. This limitation is particularly relevant in the HDLSS settings, where small test folds and strongly overlapping training sets can produce substantial variation across repeated cross-validation splits. TabArena (Erickson et al., 2025) addresses cross-dataset comparison through an Elo-style aggregation of pairwise wins, ties, and losses. Pairwise aggregation is attractive because it avoids directly averaging heterogeneous benchmark scores. Nevertheless, a hard pairwise outcome based only on a point estimate discards both the uncertainty and practical magnitude of the observed difference. For example, an improvement of 0.1 percentage points with high cross-validation variability and an improvement of 10 percentage points with low variability can both be encoded as one deterministic win. To address this issue, we introduce the **HDLSS Uncertainty-Calibrated Bradley-Terry rating (HDLSS-UCBT)**. HDLSS-UCBT retains dataset-balanced pairwise aggregation but replaces deterministic wins and losses with uncertainty-calibrated fractional outcomes constructed from matched repeated cross-validation evaluations and a Region of Practical Equivalence (RoPE). These outcomes are subsequently aggregated through a regularized Bradley-Terry model (Bradley & Terry, 1952). The resulting ratings are reported on an Elo-like scale centered at 1000. Because this centering is internal to the evaluated benchmark and comparison pool, HDLSS-UCBT values should not be interpreted as numerically comparable to external leaderboards such as TabArena (Erickson et al., 2025) or BeyondArena (Purucker et al., 2026).

Evaluation setup and notation. Let N_{ds} denote the number of benchmark datasets and N_{cmp} the number of compressors included in the UCBT comparison pool. We index compressors by $i, j \in \{1, \ldots, N_{\text{cmp}}\}$, datasets by $d \in \{1, \ldots, N_{\text{ds}}\}$, and matched cross-validation evaluations by h. For compressor i, dataset d, and matched evaluation h, we get Eq. 81.

$$\text{AUC}_{i,d,h} \in [0, 1] \tag{81}$$

Eq. 81 denotes the fold-level ROC-AUC obtained using the same train-test split as every other compared compressor. Although the construction applies to any "higher is better" predictive score, ROC-AUC is used in our final TabPFN-3 (Grinsztajn et al., 2026) analysis. We use Eq. 82 to

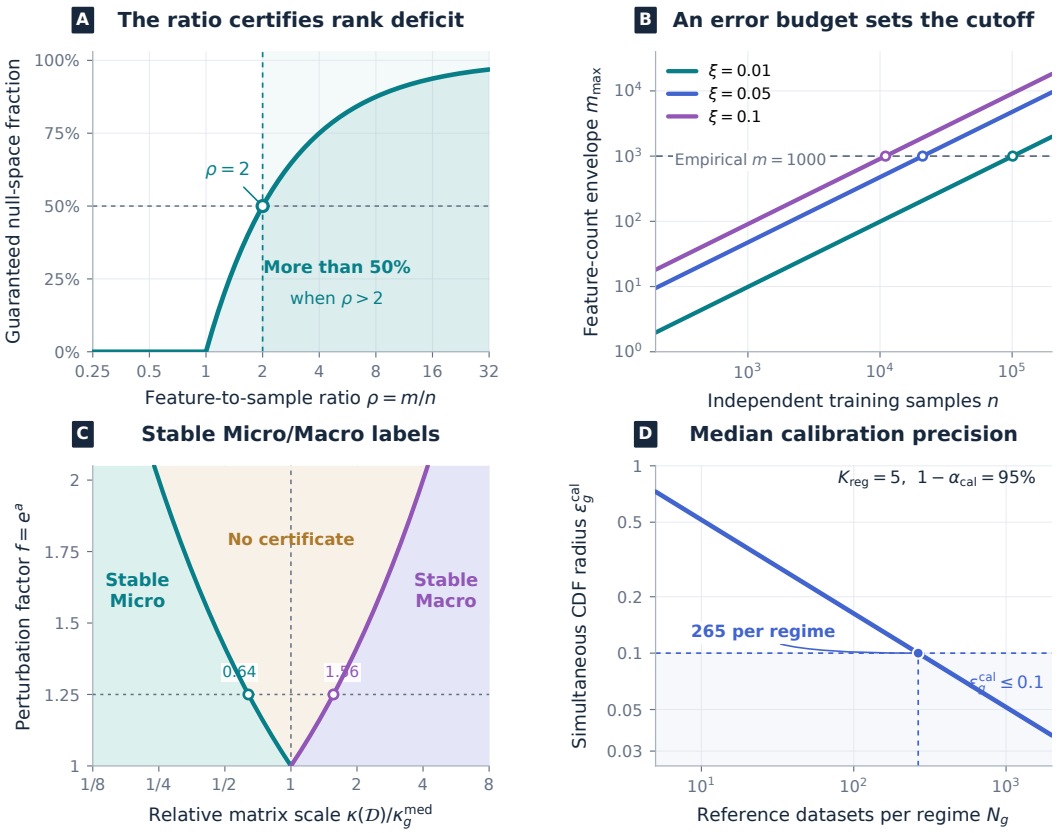


Figure D.1: **Theoretical characterization of the dataset regimes. (A)** $\rho = m/n$ lower-bounds the centered covariance null-space fraction; $\rho > 2$ places more than half of the feature space in the null space. **(B)** Under Gaussian OLS, excess-risk tolerance ξ yields an n-dependent admissible feature-count envelope, so $m = 1000$ is an empirical calibration rather than a fundamental constant. **(C)** Micro-Macro stability under perturbations of $\kappa(\mathcal{D})$ and κ_g^{med}; points far from the median boundary retain their label. **(D)** Simultaneous DKW calibration over $K_{\text{reg}} = 5$ regimes; with $\alpha_{\text{cal}} = 0.05$, $\varepsilon_g^{\text{cal}} \leq 0.1$ requires 265 reference datasets per regime. All curves are analytical, while absolute regime thresholds remain empirical calibration choices.

get matched repeated stratified cross-validation evaluations for each dataset-compressor pair. The dataset-balanced mean ROC-AUC for compressor i on dataset d is evaluated by Eq. 83. Our final UCBT benchmark contains $N_{\text{ds}} = 30$ datasets and $N_{\text{cmp}} = 10$ unsupervised compression methods. The uncompressed Raw representation is treated as an external reference and is therefore excluded from UCBT fitting, UCPW computation, average-rank calculation, and bootstrap ranking. This separation makes HDLSS-UCBT a comparison among the compression methods themselves, while Raw-relative predictive retention is evaluated separately in the main experiments.

$$N_{\text{rep}} = 5, \qquad K_{\text{cv}} = 5, \qquad N_{\text{cv}} = N_{\text{rep}} K_{\text{cv}} = 25 \tag{82}$$

$$\overline{\text{AUC}}_{i,d} = \frac{1}{N_{\text{cv}}} \sum_{h=1}^{N_{\text{cv}}} \text{AUC}_{i,d,h} \tag{83}$$

Corrected paired cross-validation uncertainty. Cross-validation evaluations are not statistically independent because their training sets overlap. Treating all N_{cv} evaluations as independent observations can therefore underestimate the uncertainty of a pairwise comparison. Following the corrected resampled variance principle of Nadeau & Bengio (2003), HDLSS-UCBT operates on the matched fold-level differences rather than independently estimated per-method standard deviations. For compressors i and j, define the paired difference on dataset d and matched evaluation h as

Eq. 84. Its mean is evaluated by Eq. 85.

$$e_{ij,d,h} = \mathrm{AUC}_{i,d,h} - \mathrm{AUC}_{j,d,h}, \qquad h = 1, \ldots, N_{\mathrm{cv}} \tag{84}$$

$$\bar{e}_{ij,d} = \frac{1}{N_{\mathrm{cv}}} \sum_{h=1}^{N_{\mathrm{cv}}} e_{ij,d,h} \tag{85}$$

Positive values favor compressor i, whereas negative values favor compressor j. Let $\mathrm{sd}\,(e_{ij,d,\cdot})$ denote the sample standard deviation of the matched differences across the N_{cv} evaluations. We estimate the corrected standard error by Eq. 86. Under our 5×5 protocol, we get Eq. 87. As the final analysis uses the actual matched fold-level predictions, the variance of the pairwise difference is observed directly. Consequently, no assumed foldwise correlation parameter or summary-statistic covariance reconstruction is required.

$$\widehat{\mathrm{SE}}^{\mathrm{cv}}_{ij,d} = \mathrm{sd}\,(e_{ij,d,\cdot}) \sqrt{\frac{1}{N_{\mathrm{cv}}} + \frac{1}{K_{\mathrm{cv}} - 1}} \tag{86}$$

$$\widehat{\mathrm{SE}}^{\mathrm{cv}}_{ij,d} = \mathrm{sd}\,(e_{ij,d,\cdot}) \sqrt{\frac{1}{25} + \frac{1}{4}} \approx 0.5385\ \mathrm{sd}\,(e_{ij,d,\cdot}) \tag{87}$$

Practical equivalence and soft pairwise outcomes. A numerically positive difference need not be practically meaningful. We therefore introduce a RoPE half-width by $\epsilon_{\mathrm{eq}} \geq 0$ and expressed on the same $[0, 1]$ scale as ROC-AUC. Our default is denoted as $\epsilon_{\mathrm{eq}} = 0.01$. This corresponding to one ROC-AUC percentage point. RoPE-based comparison allows the analysis to distinguish probabilistic superiority, practical equivalence, and probabilistic inferiority rather than forcing every nonzero numerical difference into a win or loss (Benavoli et al., 2017). The effective degrees of freedom for one dataset-level paired comparison are denoted by Eq. 88 for our 5×5 protocol, $\mathrm{df}_{\mathrm{cv}} = 24$.

$$\mathrm{df}_{\mathrm{cv}} = N_{\mathrm{cv}} - 1 \tag{88}$$

Using a Student-t confidence distribution, we define the estimated superiority and inferiority masses as Eqs. 89 and 90.

$$p^{+}_{ij,d} = F_{t_{\mathrm{df}_{\mathrm{cv}}}} \left(\frac{\bar{e}_{ij,d} - \epsilon_{\mathrm{eq}}}{\widehat{\mathrm{SE}}^{\mathrm{cv}}_{ij,d}} \right) \tag{89}$$

$$p^{-}_{ij,d} = F_{t_{\mathrm{df}_{\mathrm{cv}}}} \left(\frac{-\bar{e}_{ij,d} - \epsilon_{\mathrm{eq}}}{\widehat{\mathrm{SE}}^{\mathrm{cv}}_{ij,d}} \right) \tag{90}$$

where $F_{t_{\mathrm{df}_{\mathrm{cv}}}}$ is the cumulative distribution function of the corresponding Student-t distribution. The remaining probability mass is assigned to practical equivalence by Eq. 91.

$$p^{0}_{ij,d} = 1 - p^{+}_{ij,d} - p^{-}_{ij,d} \tag{91}$$

We then convert these three quantities into a single fractional pairwise outcome using Eq. 92.

$$\upsilon_{ij,d} = p^{+}_{ij,d} + \frac{1}{2} p^{0}_{ij,d}, \qquad 0 \leq \upsilon_{ij,d} \leq 1 \tag{92}$$

A confident and practically meaningful advantage therefore produces $\upsilon_{ij,d} \to 1$, a confident disadvantage produces $\upsilon_{ij,d} \to 0$, and an uncertain or practically equivalent comparison approaches 0.5. By construction, we get Eq. 93.

$$\upsilon_{ij,d} + \upsilon_{ji,d} = 1 \tag{93}$$

For fixed $\bar{e}_{ij,d}$, increasing uncertainty moves the fractional outcome toward 0.5. HDLSS-UCBT therefore shrinks unstable comparisons toward ties instead of counting them as deterministic wins. In the numerically degenerate case where $\widehat{\mathrm{SE}}^{\mathrm{cv}}_{ij,d}$ is effectively zero, the implementation assigns $\upsilon_{ij,d} = 1$ when $\bar{e}_{ij,d} > \epsilon_{\mathrm{eq}}$, $\upsilon_{ij,d} = 0$ when $\bar{e}_{ij,d} < -\epsilon_{\mathrm{eq}}$, and $\upsilon_{ij,d} = 0.5$ otherwise.

Uncertainty-Calibrated Pairwise Win Score Before fitting a latent rating model, we report the directly interpretable **Uncertainty-Calibrated Pairwise Win score (UCPW)** in Eq. 94. If every pairwise comparison were practically equivalent, $\mathrm{UCPW}_i = 50$. Values above 50 indicate net uncertainty-adjusted pairwise superiority across the benchmark, while values below 50 indicate net inferiority. Because every dataset contributes exactly $N_{\mathrm{cmp}} - 1$ comparisons for each compressor, the score is dataset balanced and is not dominated by datasets containing more samples.

$$\mathrm{UCPW}_i = \frac{100}{N_{\mathrm{ds}}\,(N_{\mathrm{cmp}} - 1)} \sum_{d=1}^{N_{\mathrm{ds}}} \sum_{\substack{j=1 \\ j \neq i}}^{N_{\mathrm{cmp}}} \upsilon_{ij,d} \tag{94}$$

HDLSS-UCBT rating. UCPW directly averages the uncertainty-calibrated pairwise outcomes but does not estimate a joint latent ordering. We therefore fit a regularized Bradley-Terry (Bradley & Terry, 1952) model to all $N_{\mathrm{ds}}\binom{N_{\mathrm{cmp}}}{2}$ dataset-level pairwise outcomes simultaneously. Let $\theta_i^{\mathrm{U}} \in \mathbb{R}$ denote the latent UCBT ability of compressor i. For a pair (i, j), the Bradley-Terry expected score is evaluated by Eq. 95. All compressor abilities are estimated jointly by minimizing the regularized fractional Bernoulli cross-entropy by Eq. 96.

$$\mathrm{BT}_{ij} = \frac{1}{1 + \exp\left[-\left(\theta_i^{\mathrm{U}} - \theta_j^{\mathrm{U}}\right)\right]} \tag{95}$$

$$\mathcal{J}_{\mathrm{UCBT}}(\boldsymbol{\theta}^{\mathrm{U}}) = -\frac{1}{N_{\mathrm{ds}}\binom{N_{\mathrm{cmp}}}{2}} \sum_{d=1}^{N_{\mathrm{ds}}} \sum_{i<j} \Big[\upsilon_{ij,d} \log \mathrm{BT}_{ij} + (1 - \upsilon_{ij,d}) \log(1 - \mathrm{BT}_{ij})\Big] + c_{\mathrm{BT}} \sum_{i=1}^{N_{\mathrm{cmp}}} (\theta_i^{\mathrm{U}})^2 \tag{96}$$

The scalar $c_{\mathrm{BT}} > 0$ controls the weak quadratic regularization used to prevent extreme abilities under near-separation. We use $c_{\mathrm{BT}} = 10^{-4}$. Identifiability is enforced through Eq. 97.

$$\sum_{i=1}^{N_{\mathrm{cmp}}} \theta_i^{\mathrm{U}} = 0 \tag{97}$$

Unlike sequential Elo updates, the objective in Eq. 96 is fitted jointly and is therefore invariant to the order in which dataset-level comparisons are presented. For familiar interpretation, we transform the fitted abilities to an Elo-like rating scale by Eq. 98.

$$\text{HDLSS-UCBT}_i = 1000 + \frac{400}{\ln 10} \theta_i^{\mathrm{U}} \tag{98}$$

Because of the centering constraint in Eq. 97, the arithmetic mean rating of the comparison pool is exactly 1000. A 400-point rating difference corresponds to 10:1 fitted Bradley Terry odds. The absolute rating origin therefore has no external interpretation; only differences among compressors fitted within the same benchmark and comparison pool are meaningful.

Dataset bootstrap and ranking stability. The paired cross-validation correction quantifies uncertainty within each dataset, whereas benchmark-level ranking uncertainty also depends on which datasets are included in the evaluation. We therefore quantify cross-dataset stability using a nonparametric dataset bootstrap. Let Eq. 99 denote the number of bootstrap replicates. In each replicate, the N_{ds} benchmark datasets are sampled with replacement. A dataset selected multiple times contributes its already estimated uncertainty-calibrated pairwise outcomes with the corresponding multiplicity. UCPW is recomputed and the regularized Bradley-Terry model is refitted for each replicate. This procedure preserves the matched fold-level uncertainty calculation inside each dataset while allowing the benchmark composition itself to vary. We report the percentile 95% interval of the resulting HDLSS-UCBT ratings together with the median bootstrap rank, the percentile 95% bootstrap-rank interval, and the probability of attaining rank one. For compressor i, the latter is estimated as Eq. 100 where $\mathrm{rank}_i^{(b)}$ is the UCBT rank of compressor i in bootstrap replicate b.

$$N_{\mathrm{boot}} = 1000 \tag{99}$$

$$\widehat{P}_i^{(1)} = \frac{1}{N_{\mathrm{boot}}} \sum_{b=1}^{N_{\mathrm{boot}}} \mathbf{1}\left\{\mathrm{rank}_i^{(b)} = 1\right\} \tag{100}$$

Relationship to mean ROC-AUC and average rank. HDLSS-UCBT is intended to complement rather than replace conventional cross-dataset summaries. For each compressor we therefore report four different but related views of predictive performance. First, the dataset-balanced mean ROC-AUC is evaluated with Eq. 101.

$$\overline{\mathrm{AUC}}_i = \frac{1}{N_{\mathrm{ds}}} \sum_{d=1}^{N_{\mathrm{ds}}} \overline{\mathrm{AUC}}_{i,d} \tag{101}$$

Second, let $\mathrm{rank}_d(i)$ denote the rank of compressor i according to $\overline{\mathrm{AUC}}_{i,d}$ on dataset d, with average ranks assigned to exact ties. The cross-dataset Average Rank (AR) is evaluated with Eq. 102.

$$\mathrm{AR}_i = \frac{1}{N_{\mathrm{ds}}} \sum_{d=1}^{N_{\mathrm{ds}}} \mathrm{rank}_d(i) \tag{102}$$

Mean ROC-AUC preserves predictive magnitude but does not account explicitly for uncertainty. Average rank is scale invariant and dataset balanced but discards both effect magnitude and fold-level variability. UCPW retains dataset balance while replacing hard pairwise wins by uncertainty-adjusted fractional outcomes, and HDLSS-UCBT further embeds these outcomes in a joint latent Bradley-Terry rating model. Consequently, agreement among mean ROC-AUC, AR, UCPW, and HDLSS-UCBT provides stronger evidence of stable cross-dataset performance than any one summary alone, while disagreements reveal cases in which point-estimate performance and uncertainty-aware evidence favor different interpretations.

Final benchmark protocol and interpretation. The final TabPFN-3 UCBT analysis uses $N_{\mathrm{ds}} = 30$ datasets, $N_{\mathrm{cmp}} = 10$ unsupervised compressors, $N_{\mathrm{rep}} = 5$ repeated cross-validation repetitions, $K_{\mathrm{cv}} = 5$ folds, $N_{\mathrm{cv}} = 25$ matched evaluations per dataset-compressor pair, $\epsilon_{\mathrm{eq}} = 0.01$, $c_{\mathrm{BT}} = 10^{-4}$, and $N_{\mathrm{boot}} = 1000$ dataset-bootstrap replicates with a fixed random seed. With ten compressors, every dataset contributes as per 45 (Eq. 103) unordered compressor pairs. Across the thirty datasets, the fitted UCBT model therefore aggregates 1350 (Eq. 104) dataset-level uncertainty-calibrated pairwise outcomes, each derived from 25 matched fold-level ROC-AUC evaluations. Only unsupervised compression methods with complete predictive coverage of all thirty datasets are included in this ranking pool. Raw is retained elsewhere as the uncompressed reference for predictive retention and systems comparisons, but does not participate in HDLSS-UCBT. This distinction is important because the UCBT origin is benchmark relative: adding or removing a method changes the jointly fitted comparison pool and can alter the numerical ratings.

$$\binom{N_{\mathrm{cmp}}}{2} = \binom{10}{2} = 45 \tag{103}$$

$$N_{\mathrm{ds}} \binom{N_{\mathrm{cmp}}}{2} = 30 \times 45 = 1350 \tag{104}$$

Fig. 3 in the main paper provides the compact HDLSS-UCBT leaderboard, while Table E.1 reports complete numerical statistics, including UCPW, Average Rank, mean ROC-AUC, bootstrap rank intervals, and the probability of attaining rank one. HDLSS-UCBT should be interpreted as an uncertainty-aware cross-dataset ranking statistic that distinguishes persistent pairwise predictive advantages from differences small relative to repeated cross-validation uncertainty while preserving equal weight across benchmark datasets. **Uncertainty quantification and statistical analysis.** Fig. E.1 provides the detailed statistical view of the 10 unsupervised compressors across 30 high-dimensional datasets. HDLSS-UCBT aggregates paired fold-level ROC-AUC differences while accounting for corrected cross-validation uncertainty and practical equivalence. SCOPE attains the highest uncertainty-calibrated rating and strongest average rank, with Raw retained only as an external uncompressed reference. A compact TabArena-style HDLSS-UCBT leaderboard is shown in Fig. 3 of the main paper.

F Visualization of LENS Ordering Dynamics

Beyond the solver-level comparison reported in the main paper(Table 1, we further visualize how LENS constructs and refines the feature ordering. Fig. F.1 shows the optimization trajectory on the

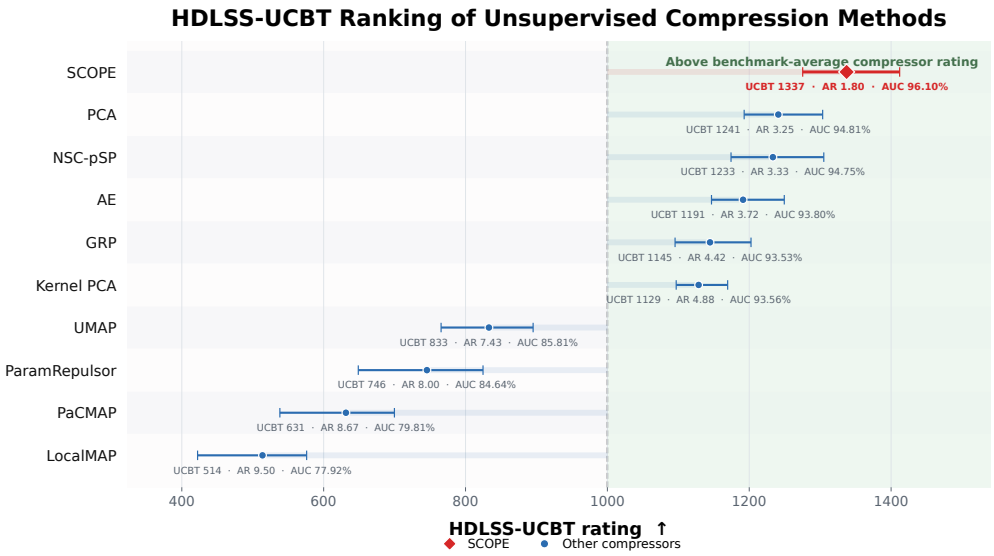


Figure E.1: **Detailed HDLSS-UCBT ranking of unsupervised compressors.** Ratings aggregate paired fold-level ROC-AUC differences across 30 HDLSS datasets with TabPFN-3. Points show fitted ratings and bars 95% dataset-bootstrap percentile intervals from 1,000 replicates; the dashed line at 1000 marks the benchmark-average rating. Raw is excluded from fitting and ranking, and AR denotes Average Rank.

Table E.1: **HDLSS-UCBT ranking of unsupervised compression methods.** Results aggregate paired fold-level ROC-AUC measurements over 30 high-dimensional datasets using TabPFN-3 as the downstream TFM. HDLSS-UCBT is estimated from 25 paired evaluations per dataset–method combination, and its 95% confidence interval is computed using 1,000 nonparametric dataset-bootstrap replicates. Raw is treated as the external uncompressed reference and is excluded from HDLSS-UCBT fitting, UCPW, average-rank computation, and bootstrap ranking. Higher HDLSS-UCBT and UCPW are better, whereas lower Avg. Rank is better. 🥇, 🥈, and 🥉 denote the first, second, and third highest HDLSS-UCBT ratings, respectively. The SCOPE row is highlighted in blue.

Method	HDLSS-UCBT	95% CI	UCPW	Avg. Rank	ROC-AUC (%)	Median Boot. Rank	Rank 95% CI	$P(\text{Rank} = 1)$
🥇 **SCOPE**	**1337.3**	**[1275.4, 1412.1]**	**82.30**	**1.80**	**96.10**	**1.0**	**[1, 1]**	**0.997**
🥈 PCA	**1241.0**	[1192.8, 1303.5]	72.94	3.25	94.81	2.0	[2, 4]	0.000
🥉 NSC-pSP	**1233.3**	[1174.3, 1305.1]	72.15	3.33	94.75	3.0	[2, 4]	0.003
AE	1191.3	[1146.8, 1249.4]	67.71	3.72	93.80	4.0	[3, 5]	0.000
GRP	1144.7	[1095.4, 1202.5]	62.70	4.42	93.53	5.0	[4, 6]	0.000
Kernel PCA	1128.6	[1097.0, 1169.6]	60.97	4.88	93.56	6.0	[5, 6]	0.000
UMAP	833.0	[765.6, 895.3]	31.98	7.43	85.81	7.0	[7, 7]	0.000
ParamRepulsor	745.6	[648.8, 824.7]	24.66	8.00	84.64	8.0	[8, 8]	0.000
PaCMAP	631.4	[538.3, 699.8]	16.02	8.67	79.81	9.0	[9, 9]	0.000
LocalMAP	513.8	[422.2, 576.0]	8.56	9.50	77.92	10.0	[10, 10]	0.000

HDLSS-UCBT is centered at 1000 over the evaluated compressor pool. UCPW denotes uncertainty-calibrated pairwise win percentage. ROC-AUC is first averaged over the 25 folds within each dataset and then averaged equally across the 30 datasets. Bootstrap rank statistics are obtained by resampling datasets rather than individual folds. Raw is not included in any ranking quantity reported in this table.

Colon dataset ($n = 62$, $m = 2000$). LENS begins from a global medoid nearest-neighbor TSP initialization with a path cost of 11874.53 and a MinLA cost of 1.4813×10^{10}. Candidate-pruned TSP 2-opt refinement accepts 580 moves and reduces the TSP cost to 11695.16. This stage primarily improves local adjacency quality and therefore does not necessarily minimize the global MinLA objective, which is 1.4841×10^{10} after TSP refinement. LENS subsequently performs MinLA-aware refinement while constraining the TSP cost through its self-calibrated path-cost budget. In this stage, 11 additional moves reduce the MinLA objective to 1.4725×10^{10}, while the final TSP cost increases only slightly to 11700.95, remaining below the admissible budget of 11700.97. This illustrates the intended behavior of LENS: the first refinement stage constructs a locally coherent feature path,

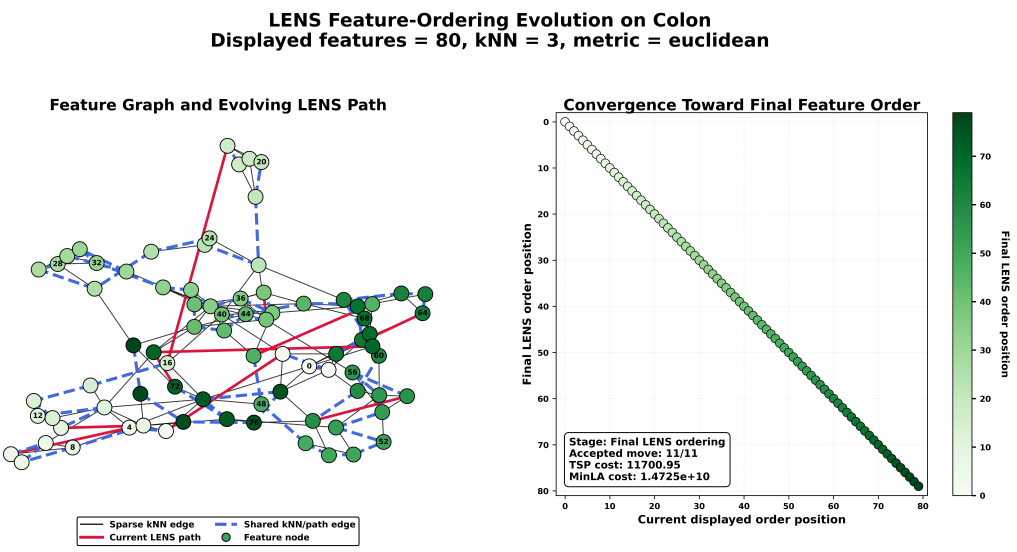


Figure F.1: **Visualization of the LENS ordering trajectory on Colon.** The left panel shows the sparse k-NN feature graph for 80 displayed features together with the final LENS ordering path. Black edges denote sparse k-NN connections, red edges denote the current LENS path, and blue dashed edges indicate connections shared by the k-NN graph and ordering path. Node color represents final LENS order position. The right panel visualizes convergence from the current displayed feature order to the final LENS ordering. At convergence, all displayed features lie on the diagonal. An animated visualization of the complete ordering evolution is provided in the supplementary material in both MP4 and GIF formats.

while the second improves global layout quality without substantially degrading the TSP structure. For visualization, we record the accepted optimization states and animate the transition from the initial medoid nearest-neighbor ordering through TSP refinement and MinLA-aware polishing. The Colon trajectory contains 580 accepted TSP moves and 11 accepted MinLA moves. To maintain a compact animation, the TSP trajectory is sampled every 10 accepted moves, while all MinLA refinement steps are retained, resulting in 92 frames and an approximately 11.5-second animation at 8 frames per second. The corresponding MP4 and GIF files are included in the supplementary material.

G Detailed Experimental Results

Extreme dimensional omics datasets. Table G.1 evaluates the methods in the most demanding dimensional regime, where all six omics datasets contain between 42,728 and 67,420 input features and therefore exceed the standard TabPFN-3 (Grinsztajn et al., 2026) feature frontier before compression. SCOPE performs particularly strongly in this setting, achieving the highest Method Avg. ROC-AUC of 97.23% and the highest mean retention of 100.50%, thereby slightly exceeding the corresponding Raw TabPFN-3 (Grinsztajn et al., 2026) reference on average despite substantial dimensionality reduction. It obtains the best compressed-method result on five of the six datasets, including Cell Cycle, OV, GBM, COAD, and BRCA, while remaining highly competitive on LGG. In contrast, several global, nonlinear, and manifold-based alternatives exhibit larger performance degradation as dimensionality increases. These results highlight SCOPE's ability to preserve, and in some cases improve, the predictive utility of TabPFN-3 (Grinsztajn et al., 2026) in extreme feature spaces where direct inference is outside the model's standard operating regime.

Text datasets. Table G.2 shows that SCOPE remains highly competitive on high-dimensional text datasets, achieving the highest overall Method Avg. ROC-AUC of 95.64% and the best mean retention of 98.98% across the five datasets. SCOPE ranks first on PCMAC and dbworld_bodies, remains near the top on BASEHOCK, and delivers strong performance across both binary and multiclass settings, including the 8-class Biblical_Texts dataset. Its aggregate performance is essentially tied with

Table G.1: ROC-AUC (%) on six extreme dimensional tabular datasets with 42,728-67,420 input features. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabPFN-3 is used as the downstream TFM. At the original feature dimensionalities considered here, the raw inputs exceed the standard TabPFN-3 feature frontier; consequently, the Raw baseline is evaluated with `ignore_pretraining_limits=True`. After dimensionality reduction, the standard TabPFN-3 size policy is used when the compressed representation falls within the supported operating frontier. The SCOPE row is highlighted in blue. Among dimensionality reduction methods, gold, silver, and bronze cells indicate the first, second, and third highest distinct mean ROC-AUC values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean performance across the six datasets. Retention denotes the mean dataset-wise ROC-AUC retention relative to Raw. † indicates a supervised dimensionality-reduction method. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 6 = 72$ method-dataset mean ROC-AUC values.

Method	Cell Cycle	OV	LGG	GBM	COAD	BRCA	Method Avg.	Retention (%)
#Samples	1,067	284	247	244	260	671	N/A	N/A
#Features	42,728	67,420	67,420	67,420	56,239	57,191	N/A	N/A
#Classes	3	4	3	5	4	5	N/A	N/A
Raw	95.62 ± 0.81	98.11 ± 0.70	98.98 ± 1.44	96.17 ± 1.73	94.08 ± 3.78	97.58 ± 0.58	96.76 ± 1.80	100.00
SCOPE (ours)	96.31 ± 0.78	99.09 ± 0.47	98.73 ± 1.92	96.64 ± 1.82	94.71 ± 3.98	97.92 ± 0.52	97.23 ± 1.66	100.50
PCA	95.03 ± 1.17	92.71 ± 2.37	98.41 ± 1.45	93.12 ± 1.96	92.25 ± 4.46	90.17 ± 2.00	93.61 ± 2.82	96.77
GRP	90.41 ± 1.31	94.34 ± 1.84	98.86 ± 1.26	93.68 ± 1.91	91.82 ± 4.83	96.28 ± 0.99	94.23 ± 3.04	97.38
AE	88.53 ± 2.68	89.45 ± 2.15	98.30 ± 1.34	87.85 ± 2.60	91.07 ± 4.39	94.74 ± 1.21	91.66 ± 4.08	94.72
UMAP	72.18 ± 3.59	84.22 ± 3.31	95.30 ± 2.58	83.88 ± 3.37	86.31 ± 4.81	89.20 ± 2.62	85.18 ± 7.63	88.00
PaCMAP	57.09 ± 3.23	76.11 ± 2.89	90.22 ± 4.29	78.39 ± 5.36	88.72 ± 4.11	84.81 ± 2.38	79.22 ± 12.19	81.86
LocalMAP	54.96 ± 2.68	73.67 ± 3.33	89.74 ± 3.74	75.58 ± 5.64	87.64 ± 4.92	84.36 ± 2.59	77.66 ± 12.85	80.24
ParamRepulsor	64.93 ± 4.80	74.07 ± 3.30	96.14 ± 2.46	77.57 ± 3.26	87.50 ± 5.29	91.28 ± 2.25	81.92 ± 11.75	84.62
Kernel PCA	87.69 ± 3.42	91.08 ± 2.51	97.97 ± 1.57	91.87 ± 2.34	91.66 ± 4.35	92.01 ± 1.17	92.05 ± 3.32	95.13
PLS†	88.31 ± 2.77	92.96 ± 2.34	98.97 ± 1.35	93.49 ± 2.66	92.38 ± 4.26	92.05 ± 2.14	93.03 ± 3.44	96.14
NSC-pSP	91.08 ± 1.05	97.91 ± 0.92	98.76 ± 1.26	89.86 ± 2.34	92.34 ± 4.59	96.68 ± 0.89	94.44 ± 3.81	97.58
Dataset Avg.	81.84 ± 15.28	88.64 ± 9.40	96.70 ± 3.35	88.17 ± 7.51	90.87 ± 2.70	92.26 ± 4.62	89.75 ± 9.24	N/A

† PLS uses training-fold labels during dimensionality reduction. Retention is computed independently for each dataset relative to Raw and then averaged across the six datasets.

PCA (Pearson, 1901) and NSC-pSP (Habib et al., 2026b) while providing slightly better overall retention relative to the Raw TabPFN-3 (Grinsztajn et al., 2026) baseline. In contrast, several nonlinear manifold methods exhibit substantial degradation on these sparse, high-dimensional text representations. Overall, the results indicate that SCOPE preserves TabPFN-3 (Grinsztajn et al., 2026) predictive utility consistently across heterogeneous text datasets without requiring task-specific feature engineering or supervised dimensionality reduction.

Face image datasets. Table G.3 shows that SCOPE also performs strongly when high-dimensional face images are represented as tabular pixel features. Across seven datasets spanning 7 to 40 classes and up to 10,304 input features, SCOPE achieves the highest Method Avg. ROC-AUC of 99.30% and the highest mean retention of 100.10%, slightly exceeding the Raw TabPFN-3 (Grinsztajn et al., 2026) reference on average. It matches the best compressed-method performance on warpPIE10P, orlraws10P, ORL, and COIL20, while remaining highly competitive on Yale, warpAR10P, and DrivFace. The narrow gap between SCOPE and the Raw baseline across most datasets indicates that the compressed representation preserves nearly all discriminative image information despite substantial dimensionality reduction. Overall, these results further support SCOPE as a robust cross-domain compression interface for TFMs, extending effectively beyond conventional tabular and omics data to high-dimensional visual representations.

Biological HDLSS datatsets. Table G.4 shows that SCOPE is particularly consistent in the classical biological HDLSS regime, spanning 12 biological datasets with only 50 - 203 samples and 2,000 - 22,283 features. SCOPE achieves the highest overall Method Avg. ROC-AUC of 93.87% and the best mean retention of 99.32%, slightly outperforming even the supervised PLS baseline in both aggregate measures. Notably, SCOPE ranks among the top three dimensionality reduction methods on all twelve datasets, attaining the best compressed-method performance on Colon,

Table G.2: ROC-AUC (%) on five high-dimensional tabular text datasets. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabPFN-3 is used as the downstream TFM. Its standard $n \times m$ operating limits are followed where applicable; otherwise, `ignore_pretraining_limits=True` is enabled without applying additional dimensionality reduction or feature selection. The SCOPE row is highlighted in blue. Among dimensionality-reduction methods, gold, silver, and bronze cells indicate the first, second, and third highest distinct mean ROC-AUC values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean performance across the five datasets. Retention denotes the mean dataset-wise ROC-AUC retention relative to Raw. † indicates a supervised dimensionality-reduction method. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 5 = 60$ method-dataset mean ROC-AUC values.

Method	RELATHE	PCMAC	dbworld_bodies	Biblical_Texts	BASEHOCK	Method Avg.	Retention (%)
#Samples	1,427	1,943	64	590	1,993	N/A	N/A
#Features	4,322	3,289	4,702	8,266	4,862	N/A	N/A
#Classes	2	2	2	8	2	N/A	N/A
Raw	96.22 ± 1.05	97.85 ± 0.56	91.00 ± 9.22	98.55 ± 0.38	99.83 ± 0.07	96.69 ± 3.44	100.00
SCOPE (ours)	93.60 ± 1.07	95.41 ± 1.28	94.34 ± 6.74	95.31 ± 1.65	99.54 ± 0.22	95.64 ± 2.30	98.98
PCA	94.95 ± 0.95	94.53 ± 1.15	92.43 ± 7.38	97.01 ± 1.16	99.21 ± 0.27	95.63 ± 2.58	98.93
GRP	92.90 ± 1.25	91.27 ± 1.20	86.43 ± 8.65	87.87 ± 2.16	98.17 ± 0.48	91.33 ± 4.62	94.46
AE	95.02 ± 1.03	94.98 ± 1.07	91.76 ± 8.03	81.26 ± 5.10	99.33 ± 0.25	92.47 ± 6.82	95.72
UMAP	84.41 ± 2.71	75.66 ± 4.95	77.88 ± 18.84	48.68 ± 4.20	79.97 ± 5.82	73.32 ± 14.15	76.03
PaCMAP	68.11 ± 4.47	55.98 ± 4.17	70.46 ± 17.25	49.10 ± 4.73	58.41 ± 5.14	60.41 ± 8.83	62.75
LocalMAP	68.26 ± 4.77	56.89 ± 3.42	70.93 ± 17.29	48.10 ± 5.12	59.77 ± 5.00	60.79 ± 9.16	63.14
ParamRepulsor	85.43 ± 2.18	74.39 ± 3.09	92.89 ± 8.81	82.17 ± 3.09	89.67 ± 1.81	84.91 ± 7.16	88.02
Kernel PCA	94.04 ± 1.36	93.72 ± 1.09	91.96 ± 8.04	94.84 ± 1.05	98.47 ± 0.39	94.61 ± 2.40	97.89
PLS†	93.76 ± 1.18	90.12 ± 1.67	86.74 ± 12.58	96.47 ± 1.06	97.95 ± 0.82	93.01 ± 4.60	96.17
NSC-pSP	93.35 ± 1.22	95.02 ± 0.70	91.55 ± 8.47	98.42 ± 0.67	99.65 ± 0.16	95.60 ± 3.40	98.88
Dataset Avg.	88.34 ± 10.10	84.65 ± 15.19	86.53 ± 8.61	81.48 ± 20.66	90.00 ± 15.57	86.20 ± 14.47	N/A

† PLS uses training labels during dimensionality reduction. Retention is computed dataset-wise relative to Raw and then averaged across the five datasets.

Lung, Lymphoma, and GLIOMA while remaining second or third on the remaining benchmarks. This consistency across binary and multiclass problems indicates that SCOPE preserves the predictive structure required by TabPFN-3 (Grinsztajn et al., 2026) despite the severe feature-to-sample imbalance characteristic of biological HDLSS data, with substantially less degradation than many conventional linear and nonlinear compression alternatives.

Pareto-optimal fidelity-efficiency trade-off. Table G.5 evaluates computational performance through bi-objective Pareto analysis, jointly considering predictive retention and each efficiency criterion. SCOPE is the only evaluated compressor that remains non-dominated across all four systems criteria, achieving a Pareto Front (PF) coverage of $4/4$; PCA (Pearson, 1901) appears on three fronts and PaCMAP (Wang et al., 2021) on two, while all remaining methods are Pareto optimal in at most one trade-off space. Although more aggressive compression can yield larger isolated speedups, these gains are typically accompanied by substantial predictive degradation. In contrast, SCOPE consistently occupies the high-fidelity region of the Pareto fronts while improving computational efficiency over Raw. As visualized in Fig. 4 in the main paper and Fig. G.1, SCOPE retains 99.46% of Raw ROC-AUC with a $2.97\times$ inference speedup and 99.71% with a $1.39\times$ fit speedup; at the latter retention level, it further achieves $1.10\times$ energy efficiency and $1.09\times$ estimated CO_2 efficiency. Together, these results show that SCOPE provides a consistently high-fidelity operating point across runtime and sustainability criteria rather than optimizing a single efficiency dimension at the expense of TabPFN-3 (Grinsztajn et al., 2026) predictive performance.

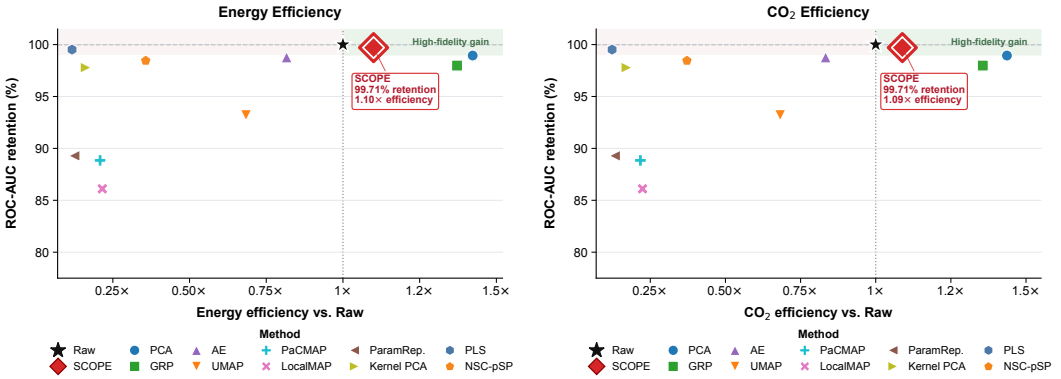
H Experiments with TabICLv2

To assess whether SCOPE transfers beyond TabPFN-3 (Grinsztajn et al., 2026), we further evaluate all unsupervised compression methods with TabICLv2 (QU et al., 2026) on the same twelve biological HDLSS datasets using both ROC-AUC and Accuracy, together with five systems-level criteria:

Table G.3: ROC-AUC (%) on seven high-dimensional face image datasets represented as tabular pixel features. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabPFN-3 is used as the downstream TFM. The SCOPE row is highlighted in blue. Among dimensionality-reduction methods, gold, silver, and bronze cells indicate the first, second, and third highest distinct mean ROC-AUC values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean performance across the seven datasets. Retention denotes the mean dataset-wise ROC-AUC retention relative to Raw. † indicates a supervised dimensionality-reduction method. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 7 = 84$ method-dataset mean ROC-AUC values.

Method	Yale	warpPIE10P	warpAR10P	orlraws10P	ORL	DrivFace	COIL20	Method Avg.	Retention (%)
#Samples	165	210	130	100	400	606	1,440	N/A	N/A
#Features	1,024	2,420	2,400	10,304	1,024	6,400	1,024	N/A	N/A
#Classes	15	10	10	10	40	7	20	N/A	N/A
Raw	98.78 ± 0.70	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	99.99 ± 0.04	95.71 ± 1.76	100.00 ± 0.00	99.21 ± 1.61	100.00
SCOPE (ours)	99.08 ± 0.55	100.00 ± 0.00	99.99 ± 0.04	100.00 ± 0.00	100.00 ± 0.00	96.06 ± 1.77	100.00 ± 0.00	99.30 ± 1.47	100.10
PCA	98.80 ± 0.65	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	95.73 ± 2.26	100.00 ± 0.00	99.22 ± 1.60	100.01
GRP	98.26 ± 0.98	100.00 ± 0.00	100.00 ± 0.00	99.99 ± 0.06	99.98 ± 0.03	95.15 ± 2.20	100.00 ± 0.00	99.05 ± 1.84	99.84
AE	98.14 ± 0.80	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	99.99 ± 0.02	95.67 ± 2.33	100.00 ± 0.00	99.11 ± 1.67	99.90
UMAP	91.13 ± 3.34	97.35 ± 1.52	83.37 ± 3.84	98.84 ± 1.01	98.44 ± 0.53	90.90 ± 2.70	99.37 ± 0.18	94.20 ± 5.97	94.94
PaCMAP	83.75 ± 4.18	91.19 ± 2.08	74.84 ± 6.15	91.51 ± 4.78	96.07 ± 1.28	91.13 ± 2.55	98.94 ± 0.34	89.63 ± 8.06	90.37
LocalMAP	80.60 ± 4.95	86.14 ± 3.28	72.90 ± 6.17	89.83 ± 5.27	89.79 ± 2.42	89.31 ± 2.81	98.27 ± 0.57	86.69 ± 8.04	87.41
ParamRepulsor	88.36 ± 2.82	92.96 ± 2.15	68.33 ± 4.30	97.30 ± 1.43	96.97 ± 0.77	90.30 ± 2.11	99.63 ± 0.11	90.55 ± 10.60	91.29
Kernel PCA	97.44 ± 1.08	100.00 ± 0.00	97.64 ± 1.88	100.00 ± 0.00	99.99 ± 0.02	95.10 ± 2.24	100.00 ± 0.00	98.60 ± 1.93	99.38
PLS†	99.52 ± 0.30	100.00 ± 0.00	100.00 ± 0.00	99.99 ± 0.06	100.00 ± 0.00	92.44 ± 2.59	100.00 ± 0.00	98.85 ± 2.83	99.62
NSC-pSP	98.19 ± 0.89	100.00 ± 0.00	99.88 ± 0.33	100.00 ± 0.00	99.99 ± 0.03	96.37 ± 1.52	100.00 ± 0.00	99.20 ± 1.42	100.00
Dataset Avg.	94.34 ± 6.68	97.30 ± 4.66	91.41 ± 12.68	98.12 ± 3.59	98.43 ± 3.04	93.66 ± 2.62	99.68 ± 0.56	96.14 ± 6.47	N/A

† PLS uses training labels during dimensionality reduction. Retention is computed dataset-wise relative to Raw and then averaged across the seven datasets.



(a) **Energy efficiency.** (b) CO_2 **efficiency.**

Figure G.1: **Sustainability-oriented efficiency analysis w/ TabPFN-3.** Energy and CO_2 efficiency are reported relative to the uncompressed Raw representation, with values above $1\times$ indicating reduced resource cost. SCOPE maintains 99.71% predictive retention while improving both energy and estimated CO_2 efficiency relative to Raw.

inference speedup, fit speedup, energy efficiency, CO_2 efficiency, and NVML GPU-memory efficiency. Across predictive metrics, SCOPE achieves the strongest cross-dataset performance, reaching a mean ROC-AUC of 93.61% with 104.42% retention and a mean Accuracy of 87.69% with 109.99% retention relative to Raw, indicating that compression can improve rather than merely preserve TabICLv2 (QU et al., 2026) performance in this regime. The systems analysis further shows that SCOPE and PCA (Pearson, 1901) are the only methods that remain Pareto optimal across all five efficiency criteria under both ROC-AUC and Accuracy, although they occupy different operating points: SCOPE forms the highest-fidelity endpoint of the frontier, while PCA (Pearson, 1901) generally provides more aggressive raw efficiency gains. LocalMAP (Wang et al., 2025b) and PaCMAP (Wang et al., 2021) contribute isolated high-speed frontier points at substantially lower predictive retention, and Kernel PCA (Schölkopf et al., 1998) contributes a memory-efficient fron-

Table G.4: ROC-AUC (%) on twelve biological HDLSS datasets. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabPFN-3 is used as the downstream TFM. The SCOPE row is highlighted in blue. Among dimensionality-reduction methods, 🥇 gold, 🥈 silver, and 🥉 bronze cells indicate the first, second, and third highest distinct mean ROC-AUC values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean ± sample standard deviation of the dataset-level mean ROC-AUC values across the twelve datasets. Retention denotes the mean dataset-wise ROC-AUC retention relative to Raw. Dataset Avg. denotes the mean ± sample standard deviation across all methods, including Raw, for each dataset. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 12 = 144$ method-dataset mean ROC-AUC values. † indicates a supervised dimensionality reduction method.

Panel A: Biological HDLSS datasets I

Method	Colon	Lung	GLI	SMK	ALLAML	Prostate
#Samples	62	203	85	187	72	102
#Features	2,000	3,312	22,283	19,993	7,129	5,966
#Classes	2	5	2	2	2	2
Raw	89.97 ± 10.51	99.67 ± 0.43	95.38 ± 5.08	76.45 ± 7.61	99.08 ± 2.30	96.05 ± 4.83
SCOPE (ours)	89.35 ± 10.14	99.75 ± 0.35	94.29 ± 6.15	75.07 ± 7.60	98.91 ± 2.79	94.93 ± 5.07
PCA	87.30 ± 10.83	99.60 ± 0.39	89.72 ± 6.92	77.34 ± 7.44	98.40 ± 2.45	94.53 ± 5.18
GRP	84.50 ± 10.89	99.70 ± 0.34	91.22 ± 7.92	72.86 ± 7.32	94.68 ± 7.07	90.75 ± 6.48
AE	88.00 ± 10.76	99.70 ± 0.38	92.90 ± 5.95	73.94 ± 6.40	99.02 ± 2.74	93.82 ± 5.27
UMAP	88.78 ± 12.93	99.46 ± 0.66	87.78 ± 6.51	69.13 ± 6.86	87.10 ± 10.55	86.37 ± 6.17
PaCMAP	84.28 ± 12.23	97.00 ± 4.41	82.84 ± 9.52	69.90 ± 6.47	76.09 ± 13.19	83.43 ± 8.38
LocalMAP	75.08 ± 14.51	91.20 ± 6.54	81.30 ± 11.94	69.17 ± 8.24	74.34 ± 14.26	79.36 ± 8.22
ParamRepulsor	72.50 ± 20.95	97.84 ± 2.12	81.07 ± 10.23	71.30 ± 7.87	81.78 ± 12.73	78.99 ± 10.63
Kernel PCA	88.75 ± 9.96	99.64 ± 0.39	89.11 ± 7.41	72.67 ± 5.88	96.10 ± 4.91	92.73 ± 4.94
PLS†	88.17 ± 12.19	99.23 ± 1.26	95.67 ± 5.11	76.59 ± 6.82	99.07 ± 2.31	96.72 ± 4.34
NSC-pSP	87.93 ± 10.70	99.65 ± 0.46	89.18 ± 8.01	73.70 ± 7.82	95.68 ± 5.59	92.33 ± 5.68
Dataset Avg.	85.38 ± 5.71	98.54 ± 2.47	89.20 ± 5.18	73.18 ± 2.88	91.69 ± 9.38	90.00 ± 6.36

Panel B: Biological HDLSS datasets II and cross-dataset summary

Method	Arcene	TOX	Lymphoma	Leukemia	GLIOMA	CLL	Method Avg.	Retention (%)
#Samples	200	171	96	72	50	111	N/A	N/A
#Features	10,000	5,748	4,026	7,070	4,434	11,340	N/A	N/A
#Classes	2	4	9	2	4	3	N/A	N/A
Raw	95.78 ± 2.04	98.86 ± 1.05	98.72 ± 3.04	99.24 ± 1.98	95.38 ± 3.86	89.21 ± 4.29	94.48 ± 6.65	100.00
SCOPE (ours)	92.92 ± 3.23	98.80 ± 1.07	98.93 ± 2.64	99.31 ± 1.75	95.87 ± 3.48	88.31 ± 5.10	93.87 ± 7.07	99.32
PCA	90.47 ± 4.84	98.39 ± 1.55	98.62 ± 3.17	98.78 ± 2.92	92.73 ± 5.27	83.93 ± 4.78	92.48 ± 7.03	97.89
GRP	89.02 ± 4.31	95.74 ± 1.51	98.32 ± 3.11	96.19 ± 6.32	93.39 ± 4.67	84.13 ± 5.56	90.88 ± 7.51	96.12
AE	86.23 ± 4.80	97.38 ± 1.59	98.62 ± 2.91	98.14 ± 3.96	95.32 ± 4.15	84.70 ± 5.77	92.31 ± 7.76	97.65
UMAP	83.58 ± 4.77	87.21 ± 3.63	96.41 ± 4.10	89.34 ± 10.80	86.18 ± 5.05	75.93 ± 7.19	86.44 ± 8.00	91.45
PaCMAP	82.56 ± 4.07	80.14 ± 5.28	92.29 ± 7.03	89.44 ± 8.86	81.39 ± 9.35	70.21 ± 8.09	82.46 ± 8.07	87.32
LocalMAP	81.99 ± 4.51	74.19 ± 6.68	91.00 ± 6.77	87.22 ± 9.12	86.32 ± 7.39	69.76 ± 9.99	80.08 ± 7.73	84.81
ParamRepulsor	78.93 ± 7.23	84.15 ± 3.84	94.06 ± 4.34	87.19 ± 9.35	86.83 ± 7.02	74.83 ± 6.90	82.46 ± 8.11	87.28
Kernel PCA	87.36 ± 4.78	95.88 ± 1.56	98.16 ± 3.60	97.48 ± 4.16	91.82 ± 5.94	81.65 ± 5.89	90.95 ± 7.78	96.19
PLS†	91.96 ± 3.82	99.77 ± 0.41	98.46 ± 3.51	98.12 ± 3.95	92.48 ± 6.99	88.48 ± 5.68	93.73 ± 6.79	99.20
NSC-pSP	94.34 ± 2.79	97.26 ± 1.40	98.10 ± 3.51	99.66 ± 1.17	90.88 ± 4.59	80.89 ± 4.86	91.63 ± 7.88	96.91
Dataset Avg.	87.93 ± 5.38	92.31 ± 8.64	96.81 ± 2.78	95.01 ± 5.08	90.72 ± 4.54	81.00 ± 6.85	**89.31 ± 8.74**	N/A

† PLS uses training labels during dimensionality reduction. Retention is computed independently for each dataset relative to Raw and then averaged across the twelve datasets.

tier point. Collectively, these results demonstrate that the SCOPE representation generalizes to a second tabular foundation model and maintains a favorable fidelity-efficiency trade-off across predictive, runtime, sustainability, and memory criteria.

Evaluation with ROC-AUC. Table H.1 evaluates the unsupervised compression methods using ROC-AUC with TabICLv2 (QU et al., 2026). SCOPE achieves the strongest cross-dataset performance, with the highest Method Avg. of 93.61% and the highest Raw-relative ROC-AUC retention of 104.42%, compared with 90.08% for the uncompressed Raw representation. PCA (Pearson,

Table G.5: **Pareto front analysis of predictive fidelity and computational efficiency.** Each systems criterion is formulated as a two-objective problem jointly maximizing ROC-AUC retention and the corresponding efficiency measure. Each table entry reports *retention / efficiency*. A compression method belongs to the Pareto Front (PF) when no other compression method achieves both "equal or higher" predictive retention and "equal or higher" efficiency, with at least one strict improvement. Inference results are macro-aggregated across the biological HDLSS, facen image, and text domains, whereas fit, energy, and CO_2 results use the biological HDLSS and face-image domains for which common controlled measurements are available. Raw is reported only as the uncompressed reference and is excluded from PF computation. **PF Coverage** counts the number of trade-off spaces in which each compression method is non-dominated. SCOPE is the only evaluated compressor belonging to all four Pareto fronts. † denotes a supervised dimensionality reduction method.

Method	**Inference** Retention / Speedup Bio. + Face + Text	**Fit** Retention / Speedup Bio. + Face	**Energy** Retention / Efficiency Bio. + Face	CO_2 Retention / Efficiency Bio. + Face	**PF Coverage**
Raw	100.00% 1.00× Reference	100.00% 1.00× Reference	100.00% 1.00× Reference	100.00% 1.00× Reference	*Ref.*
SCOPE (ours)	**99.46%** 2.97× **PF**	**99.71%** 1.39× **PF**	**99.71%** 1.10× **PF**	**99.71%** 1.09× **PF**	**4/4**
PCA	**98.94%** 3.48× **PF**	98.95% 1.27× Dominated	**98.95%** 1.42× **PF**	**98.95%** 1.44× **PF**	**3/4**
GRP	96.81% 3.30× Dominated	97.98% 1.30× Dominated	97.98% 1.37× Dominated	97.98% 1.36× Dominated	0/4
AE	97.76% 3.07× Dominated	98.78% 1.19× Dominated	98.78% 0.82× Dominated	98.78% 0.83× Dominated	0/4
UMAP	**87.47%** 3.58× **PF**	93.20% 1.21× Dominated	93.20% 0.68× Dominated	93.20% 0.68× Dominated	1/4
PaCMAP	**80.15%** 3.78× **PF**	**88.84%** 1.74× **PF**	88.84% 0.21× Dominated	88.84% 0.22× Dominated	**2/4**
LocalMAP	78.45% 3.71× Dominated	**86.11%** 2.01× **PF**	86.11% 0.22× Dominated	86.11% 0.22× Dominated	1/4
ParamRepulsor	88.86% 2.31× Dominated	89.28% 0.17× Dominated	89.28% 0.12× Dominated	89.28% 0.13× Dominated	0/4
Kernel PCA	97.82% 2.51× Dominated	97.78% 0.19× Dominated	97.78% 0.16× Dominated	97.78% 0.17× Dominated	0/4
PLS†	98.40% 2.61× Dominated	99.52% 0.29× Dominated	99.52% 0.12× Dominated	99.52% 0.12× Dominated	0/4
NSC-pSP	98.87% 2.52× Dominated	98.45% 0.40× Dominated	98.45% 0.36× Dominated	98.45% 0.37× Dominated	0/4

PF denotes membership in the non-dominated Pareto front. For two compression methods a and b, method a dominates method b for a given systems criterion if $R_a \geq R_b$ and $E_a \geq E_b$, with at least one strict inequality, where R denotes ROC-AUC retention and E denotes the corresponding efficiency quantity. Inference and fit efficiency are Raw-relative speedups. Energy efficiency is defined as $E_{\text{Raw}}/E_{\text{method}}$, and CO_2 efficiency as $C_{\text{Raw}}/C_{\text{method}}$; therefore, larger values are preferable for all efficiency objectives. Green cells indicate Pareto-optimal baseline operating points. The SCOPE row remains blue for visual consistency with the remaining experimental tables. Gold, silver, and bronze in the final column indicate the three highest distinct PF-Coverage values.

1901) ranks second with 93.36% average ROC-AUC and 104.11% retention, followed by NSC-pSP (Habib et al., 2026b) with 92.97% and 103.66%, respectively. These results show that, under TabICLv2 (QU et al., 2026), SCOPE not only preserves predictive quality after compression but improves average ROC-AUC relative to the uncompressed representation across the twelve biological HDLSS datasets.

Table H.1: ROC-AUC (%) on twelve biological HDLSS datasets. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabICLv2 is used as the downstream TFM. Only unsupervised dimensionality reduction methods are considered. The SCOPE row is highlighted in blue. Among dimensionality reduction methods, gold, silver, and bronze cells indicate the first, second, and third highest distinct mean values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean ± sample standard deviation of the dataset-level mean ROC-AUC values across the twelve datasets. Retention denotes the mean dataset-wise ROC-AUC retention relative to Raw. Dataset Avg. denotes the mean ± sample standard deviation across all included methods, including Raw, for each dataset. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 11 = 132$ method-dataset mean ROC-AUC values.

Panel A: Biological HDLSS datasets I

Method	Colon	Lung	GLI	SMK	ALLAML	Prostate
#Samples	62	203	85	187	72	102
#Features	2,000	3,312	22,283	19,993	7,129	5,966
#Classes	2	5	2	2	2	2
Raw	71.57 ± 9.78	99.60 ± 0.52	89.59 ± 6.80	73.13 ± 7.33	98.42 ± 3.29	91.20 ± 7.01
SCOPE (ours)	88.28 ± 10.21	99.58 ± 0.56	93.57 ± 7.17	75.95 ± 7.67	98.89 ± 3.14	94.37 ± 4.21
PCA	87.25 ± 10.75	97.63 ± 0.38	92.99 ± 6.34	74.40 ± 7.37	97.79 ± 3.10	95.48 ± 4.39
GRP	85.93 ± 11.95	99.69 ± 0.43	93.05 ± 6.38	72.07 ± 8.54	96.30 ± 5.30	91.66 ± 5.34
AE	89.62 ± 10.71	99.69 ± 0.37	92.45 ± 5.89	72.45 ± 7.03	98.82 ± 3.51	93.89 ± 5.20
UMAP	88.03 ± 9.92	99.53 ± 0.52	85.30 ± 9.00	68.78 ± 7.49	88.75 ± 8.35	85.13 ± 7.69
PaCMAP	82.90 ± 13.34	95.35 ± 5.27	83.57 ± 8.99	69.83 ± 7.19	72.74 ± 16.79	81.00 ± 8.36
LocalMAP	75.40 ± 15.08	91.07 ± 7.05	83.50 ± 7.99	69.28 ± 7.39	76.82 ± 13.75	78.24 ± 8.35
ParamRepulsor	69.60 ± 19.80	98.20 ± 1.94	79.36 ± 9.70	71.49 ± 7.13	81.05 ± 11.17	75.91 ± 10.02
Kernel PCA	84.35 ± 11.46	99.52 ± 0.57	91.19 ± 7.30	72.95 ± 7.54	96.86 ± 5.73	93.04 ± 5.55
NSC-pSP	90.28 ± 10.30	99.64 ± 0.44	93.12 ± 6.49	72.65 ± 6.75	98.99 ± 2.93	91.28 ± 6.65
Dataset Avg.	83.02 ± 7.39	98.14 ± 2.71	88.88 ± 5.03	72.09 ± 2.16	91.40 ± 9.94	88.29 ± 6.99

Panel B: Biological HDLSS datasets II and cross-dataset summary

Method	Arcene	TOX	Lymphoma	Leukemia	GLIOMA	CLL	Method Avg.	Retention (%)
#Samples	200	171	62	72	50	111	N/A	N/A
#Features	10,000	5,748	4,026	7,070	4,434	11,340	N/A	N/A
#Classes	2	4	3	2	4	3	N/A	N/A
Raw	90.35 ± 4.29	97.01 ± 2.10	100.00 ± 0.00	97.53 ± 5.78	92.45 ± 4.38	80.10 ± 6.77	90.08 ± 9.99	100.00
SCOPE (ours)	92.92 ± 2.88	99.15 ± 0.94	100.00 ± 0.00	98.98 ± 2.72	95.02 ± 3.23	86.65 ± 4.54	93.61 ± 7.11	104.42
PCA	92.52 ± 3.96	99.94 ± 0.23	100.00 ± 0.00	98.97 ± 2.55	95.84 ± 3.70	87.52 ± 6.10	93.36 ± 7.38	104.11
GRP	90.48 ± 3.97	97.85 ± 1.58	100.00 ± 0.00	98.74 ± 2.79	94.42 ± 5.00	85.28 ± 5.05	92.12 ± 8.03	102.65
AE	87.06 ± 4.78	98.79 ± 1.20	100.00 ± 0.00	98.05 ± 4.96	96.04 ± 3.89	85.23 ± 4.55	92.68 ± 8.12	103.33
UMAP	82.65 ± 5.01	87.12 ± 3.43	98.51 ± 3.44	89.01 ± 10.24	88.54 ± 5.27	71.03 ± 7.89	86.03 ± 9.05	95.93
PaCMAP	82.68 ± 3.50	80.08 ± 5.43	96.40 ± 8.45	89.11 ± 8.96	77.65 ± 10.18	69.06 ± 7.94	81.70 ± 8.86	91.26
LocalMAP	82.29 ± 3.54	74.28 ± 6.05	98.27 ± 4.65	88.26 ± 9.06	84.63 ± 6.30	71.79 ± 8.89	81.15 ± 8.51	90.51
ParamRepulsor	75.65 ± 7.53	81.94 ± 4.15	99.74 ± 0.78	85.91 ± 8.53	85.24 ± 6.51	72.64 ± 6.25	81.39 ± 9.70	90.56
Kernel PCA	87.96 ± 5.76	99.38 ± 0.66	100.00 ± 0.00	98.81 ± 3.27	93.25 ± 4.44	81.38 ± 6.95	91.56 ± 8.51	101.94
NSC-pSP	94.70 ± 2.18	98.64 ± 1.31	100.00 ± 0.00	98.89 ± 2.64	93.49 ± 4.36	83.96 ± 6.21	92.97 ± 8.01	103.66
Dataset Avg.	87.20 ± 5.81	92.20 ± 9.48	99.36 ± 1.17	94.75 ± 5.37	90.60 ± 5.89	79.51 ± 7.02	**88.79 ± 9.57**	N/A

Only unsupervised dimensionality reduction methods are included in the comparison. Retention is computed independently for each dataset relative to Raw and then averaged across the twelve datasets.

Evaluation with Accuracy. Table H.2 further evaluates the unsupervised compression methods using classification Accuracy with TabICLv2 (QU et al., 2026). SCOPE achieves the strongest cross-dataset performance, with the highest Method Avg. of 87.69% and the highest Raw-relative Accuracy retention of 109.99%, compared with 80.39% for the uncompressed Raw representation. PCA (Pearson, 1901) ranks second with 87.07% average Accuracy and 109.21% retention, followed by NSC-pSP (Habib et al., 2026b) with 86.50% and 108.48%, respectively. These results indicate that, for TabICLv2 (QU et al., 2026), SCOPE does not merely preserve predictive performance under

Table H.2: Accuracy (%) on twelve biological HDLSS datasets. Results are reported as mean ± standard deviation over 25 runs (5 repeats × 5 folds). TabICLv2 is used as the downstream TFM. Only unsupervised dimensionality reduction methods are considered. The SCOPE row is highlighted in blue. Among dimensionality-reduction methods, gold, silver, and bronze cells indicate the first, second, and third highest distinct mean values, respectively. Tied methods at the displayed precision receive the same rank. Raw is reported separately and excluded from ranking. Method Avg. denotes the mean ± sample standard deviation of the dataset-level mean Accuracy values across the twelve datasets. Retention denotes the mean dataset-wise Accuracy retention relative to Raw. Dataset Avg. denotes the mean ± sample standard deviation across all included methods, including Raw, for each dataset. The **purple value** at the intersection of Dataset Avg. and Method Avg. is the Grand Avg., computed over all $12 \times 11 = 132$ method-dataset mean Accuracy values.

Panel A: Biological HDLSS datasets I

Method	Colon	Lung	GLI	SMK	ALLAML	Prostate
#Samples	62	203	85	187	72	102
#Features	2,000	3,312	22,283	19,993	7,129	5,966
#Classes	2	5	2	2	2	2
Raw	65.92 ± 8.33	96.45 ± 2.26	71.06 ± 4.47	66.79 ± 7.96	90.02 ± 7.75	85.27 ± 9.45
SCOPE (ours)	84.33 ± 11.09	96.16 ± 2.48	89.41 ± 6.58	70.89 ± 6.58	97.77 ± 3.32	89.78 ± 5.86
PCA	80.77 ± 10.29	96.75 ± 2.54	86.59 ± 7.11	72.80 ± 7.45	94.19 ± 6.42	89.22 ± 5.45
GRP	81.82 ± 10.68	97.63 ± 1.97	87.76 ± 6.78	67.37 ± 7.05	89.75 ± 7.61	86.11 ± 6.89
AE	84.28 ± 11.31	96.55 ± 2.48	85.65 ± 6.14	67.55 ± 5.83	94.15 ± 6.26	90.36 ± 4.62
UMAP	78.15 ± 9.78	95.27 ± 2.93	76.24 ± 8.57	66.38 ± 5.76	80.17 ± 10.03	76.90 ± 11.79
PaCMAP	74.15 ± 9.42	92.40 ± 3.43	74.82 ± 9.98	68.20 ± 5.76	75.03 ± 8.65	72.70 ± 8.64
LocalMAP	68.08 ± 9.40	91.83 ± 4.20	73.18 ± 8.67	67.33 ± 6.40	73.92 ± 7.64	68.84 ± 9.17
ParamRepulsor	65.56 ± 10.68	89.56 ± 4.34	71.53 ± 9.25	67.02 ± 6.53	75.60 ± 9.28	69.97 ± 8.99
Kernel PCA	77.18 ± 9.57	95.97 ± 3.00	79.06 ± 12.94	68.63 ± 6.05	81.39 ± 8.37	82.57 ± 9.62
NSC-pSP	86.26 ± 10.44	96.06 ± 2.56	90.59 ± 7.00	68.77 ± 7.09	96.63 ± 4.14	83.37 ± 7.65
Dataset Avg.	76.96 ± 7.57	94.97 ± 2.54	80.53 ± 7.57	68.34 ± 1.93	86.24 ± 9.19	81.37 ± 7.99

Panel B: Biological HDLSS datasets II and cross-dataset summary

Method	Arcene	TOX	Lymphoma	Leukemia	GLIOMA	CLL	Method Avg.	Retention (%)
#Samples	200	171	62	72	50	111	N/A	N/A
#Features	10,000	5,748	4,026	7,070	4,434	11,340	N/A	N/A
#Classes	2	4	3	2	4	3	N/A	N/A
Raw	82.00 ± 5.59	84.32 ± 7.48	99.05 ± 3.44	92.17 ± 7.11	66.80 ± 10.69	64.84 ± 9.82	80.39 ± 12.75	100.00
SCOPE (ours)	86.30 ± 3.96	91.91 ± 5.47	99.36 ± 2.22	97.52 ± 3.88	76.40 ± 9.95	72.43 ± 6.99	87.69 ± 9.94	109.99
PCA	83.60 ± 6.62	97.07 ± 3.60	98.08 ± 4.24	96.08 ± 6.13	77.60 ± 10.91	72.06 ± 8.06	87.07 ± 9.64	109.21
GRP	81.10 ± 5.36	87.14 ± 6.11	99.67 ± 1.67	91.68 ± 6.41	77.20 ± 11.73	71.34 ± 9.12	84.88 ± 9.72	106.53
AE	78.80 ± 5.87	90.41 ± 5.75	98.69 ± 3.06	94.44 ± 5.26	78.00 ± 10.00	69.72 ± 7.62	85.72 ± 10.32	107.49
UMAP	73.30 ± 6.52	60.71 ± 8.88	91.28 ± 6.98	82.11 ± 11.78	59.60 ± 11.36	51.69 ± 7.61	74.32 ± 12.88	92.90
PaCMAP	72.70 ± 6.16	54.48 ± 8.77	96.72 ± 4.76	83.33 ± 9.99	52.40 ± 12.68	50.85 ± 8.00	72.32 ± 14.55	90.21
LocalMAP	73.50 ± 5.05	46.54 ± 7.55	94.21 ± 6.83	80.95 ± 10.77	53.20 ± 10.69	53.91 ± 9.21	70.46 ± 14.53	87.97
ParamRepulsor	67.20 ± 7.05	51.14 ± 9.27	95.82 ± 5.70	79.22 ± 9.48	60.80 ± 11.15	53.00 ± 8.00	70.54 ± 13.25	88.12
Kernel PCA	72.90 ± 8.41	90.99 ± 4.03	99.05 ± 2.62	89.49 ± 8.13	70.80 ± 12.22	63.75 ± 7.36	80.98 ± 11.13	101.34
NSC-pSP	87.80 ± 4.47	90.53 ± 5.66	98.38 ± 3.30	96.93 ± 3.53	74.00 ± 9.13	68.73 ± 9.37	86.50 ± 10.77	108.48
Dataset Avg.	78.11 ± 6.60	76.84 ± 19.26	97.30 ± 2.61	89.45 ± 6.86	67.89 ± 9.89	62.94 ± 8.83	80.08 ± 13.23	N/A

Only unsupervised dimensionality reduction methods are included in the comparison. Retention is computed independently for each dataset relative to Raw and then averaged across the twelve datasets.

compression but improves average Accuracy relative to the uncompressed representation across the twelve biological HDLSS datasets.

Pareto optimal fidelity-efficiency trade-off with TabICLv2. Table H.3 jointly evaluates predictive fidelity and systems efficiency across five criteria: inference speedup, fit speedup, energy efficiency, CO_2 efficiency, and NVML GPU-memory efficiency. SCOPE and PCA (Pearson, 1901) are the only unsupervised compressors that remain non-dominated under all five criteria for both ROC-AUC and Accuracy, each attaining complete PF coverage of $5/5$. They nevertheless occupy distinct operating points on the high-fidelity Pareto frontier. SCOPE provides the highest predictive

retention among the evaluated compressors, reaching 104.42% ROC-AUC retention and 109.99% Accuracy retention, while simultaneously achieving $12.94\times$ inference speedup, $10.48\times$ fit speedup, $2.50\times$ energy and CO_2 efficiency, and $7.68\times$ GPU-memory efficiency relative to Raw. PCA (Pearson, 1901) sacrifices a small amount of predictive retention in exchange for more aggressive resource reductions, and therefore represents the higher efficiency portion of the same frontier. The full trade-off geometry is visualized in Fig. H.1 for inference, fitting, and GPU memory and in Fig. H.2 for energy and CO_2, with ROC-AUC and Accuracy shown side by side. Other methods enter the Pareto front only selectively: LocalMAP (Wang et al., 2025b) contributes high-speed inference and fitting operating points, PaCMAP (Wang et al., 2021) contributes a fit-efficient point, and Kernel PCA (Schölkopf et al., 1998) provides the most memory-efficient frontier point, but each does so at substantially lower predictive retention than SCOPE or PCA (Pearson, 1901). Overall, the TabICLv2 (QU et al., 2026) results position SCOPE as the highest-fidelity endpoint of a Pareto optimal region that remains efficient across all five evaluated systems dimensions, rather than as the maximizer of any single resource metric.

I TabPFN Override Stress Test

We additionally examined whether TabPFN-family checkpoints can be forced to operate in the HDLSS regime by setting `ignore_pretraining_limits=True`. These results are treated as an out-of-scope stress test rather than part of the primary leaderboard, since the override bypasses the feature limits associated with the original checkpoints. TabPFN-3.5 (Jäger et al., 2026) was not yet released when these experiments were completed and is therefore not included in this analysis. We also evaluated the specialized RealTabPFN-2.5 (Grinsztajn et al., 2025) large feature checkpoint under the same override. Performance varied substantially across datasets: TabPFN-3^+ (Grinsztajn et al., 2026) achieved the highest mean accuracy on Colon, ALLAML, and Arcene, TabPFN-2.6^+ (Grinsztajn et al., 2025) performed best on Lung, and TabPFN $v2^+$ (Hollmann et al., 2025) was best on SMK and Prostate. TabPFN-2.5^+ (Grinsztajn et al., 2025) obtained the best result on TOX and tied the two RealTabPFN variants on GLI. Notably, the large feature RealTabPFN checkpoint provided only modest changes relative to the default RealTabPFN-2.5 (Grinsztajn et al., 2025) override, indicating that bypassing the configured limits does not yield a uniformly superior checkpoint in the extreme high-dimensional regime.

J Dataset Sources

This appendix lists the 30 publicly available high-dimensional datasets ($m \gg n$) used throughout our experiments together with their original or benchmark sources. The collection spans extreme dimensional omics, biological HDLSS, face image, and high-dimensional text datasets (bag of words), and includes datasets adopted from established HDLSS benchmark repositories as well as task-specific sources. Table J.1 provides the corresponding references for reproducibility and dataset provenance.

K HDLSS Block Model Analysis and Ablation for Dimensionality Reduction of SCOPE

To isolate the behavior of SCOPE under controlled block structure, we evaluate all methods on a BSTabDiff-style (Habib et al., 2026a) block-structured synthetic HDLSS benchmark with the representation dimension fixed at $M = 32$ following NSC-pSP (Habib et al., 2026b). The experiment is repeated independently 10 times and evaluates both predictive separability, through linear and kNN accuracy, and representation geometry, through silhouette and Davies-Bouldin (DB) scores. Fixing the output dimensionality allows the comparison to focus on representation quality rather than differences in compression budget.

Per-repetition behavior. Table K.1 reports the complete results across the 10 repetitions. Relative to NSC-pSP (Habib et al., 2026b), SCOPE records 23 improvements, 3 ties, and 14 degradations across the 40 repetition-metric comparisons. The gains are particularly consistent for linear accuracy, kNN accuracy, and silhouette, where SCOPE improves over NSC-pSP (Habib et al., 2026b)

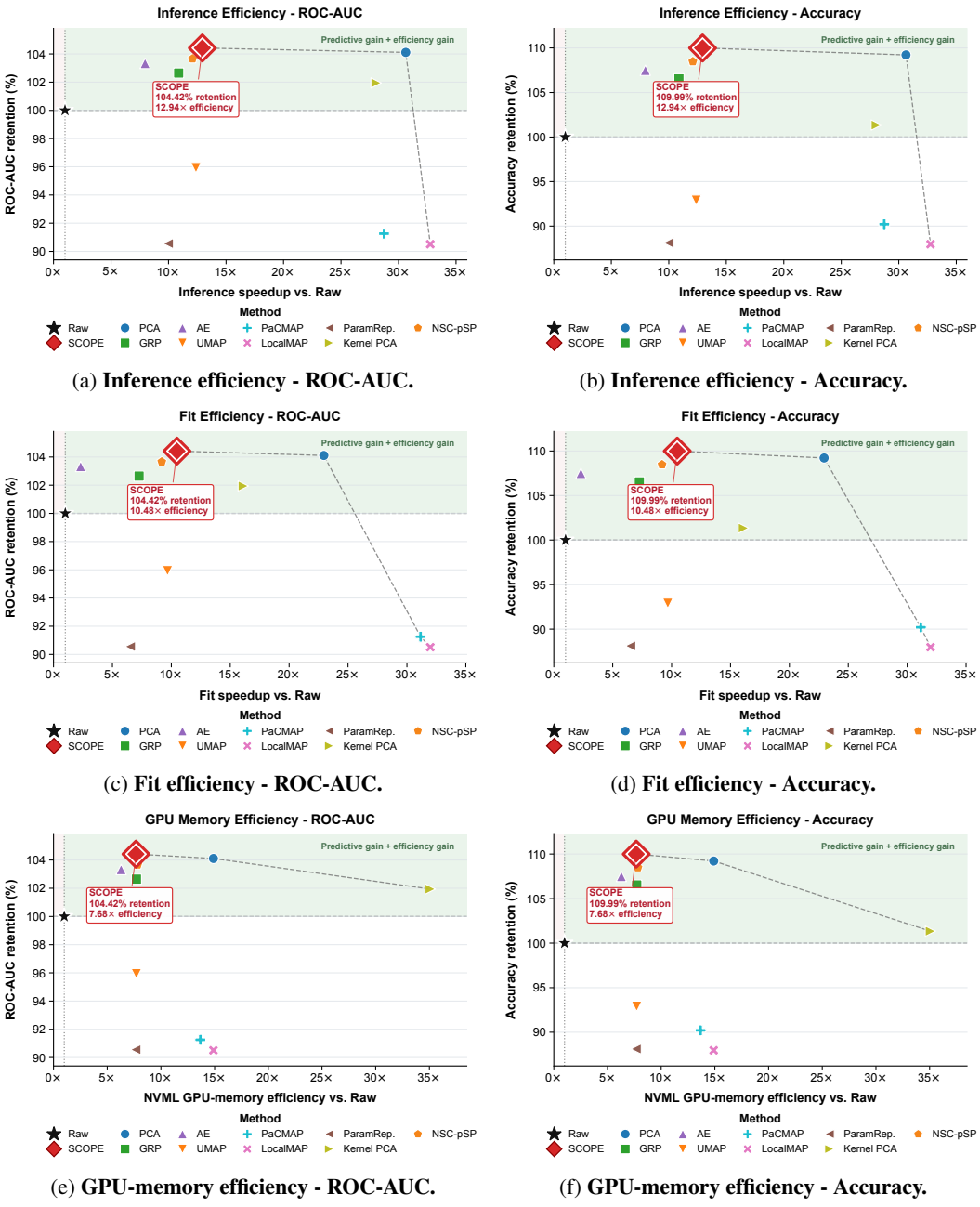
Table H.3: **Pareto front analysis with TabICLv2 on the twelve biological HDLSS datasets.** Each systems criterion jointly evaluates predictive fidelity and computational efficiency. Within each cell, the first line reports ROC-AUC retention, the second line reports Accuracy retention, and the third line reports the corresponding Raw-relative speedup or efficiency. Green cells indicate compression methods that lie on the non-dominated PF under both ROC-AUC and Accuracy for the corresponding systems criterion. Raw is shown only as the uncompressed reference and is excluded from PF computation. **PF Coverage** counts the number of trade-off spaces in which a method is non-dominated. Inference and fit are evaluated using Raw-relative speedups. Energy and CO_2 efficiencies are computed from CodeCarbon (Courty et al., 2026) whole-run measurements, and GPU-memory efficiency is computed from NVML peak-used memory. Among compression methods, gold, silver, and bronze in the final column denote the first, second, and third highest distinct PF-Coverage values. LocalMAP has no available CodeCarbon (Courty et al., 2026) energy or CO_2 measurements and is therefore marked N/A in those columns.

Method	Inference ROC / Acc / Speedup	Fit ROC / Acc / Speedup	Energy ROC / Acc / Efficiency	CO_2 ROC / Acc / Efficiency	NVML Memory ROC / Acc / Efficiency	PF Coverage
Raw	**100.00%** **100.00%** 1.00× Reference	**100.00%** **100.00%** 1.00× Reference	**100.00%** **100.00%** 1.00× Reference	**100.00%** **100.00%** 1.00× Reference	**100.00%** **100.00%** 1.00× Reference	*Ref.*
SCOPE (ours)	**104.42%** **109.99%** 12.94× **PF**	**104.42%** **109.99%** 10.48× **PF**	**104.42%** **109.99%** 2.50× **PF**	**104.42%** **109.99%** 2.50× **PF**	**104.42%** **109.99%** 7.68× **PF**	**5/5**
PCA	**104.11%** **109.21%** 30.63× **PF**	**104.11%** **109.21%** 22.95× **PF**	**104.11%** **109.21%** 5.70× **PF**	**104.11%** **109.21%** 5.70× **PF**	**104.11%** **109.21%** 14.89× **PF**	**5/5**
GRP	**102.65%** **106.53%** 10.88× Dominated	**102.65%** **106.53%** 7.26× Dominated	**102.65%** **106.53%** 2.04× Dominated	**102.65%** **106.53%** 2.04× Dominated	**102.65%** **106.53%** 7.72× Dominated	**0/5**
AE	**103.33%** **107.49%** 7.96× Dominated	**103.33%** **107.49%** 2.30× Dominated	**103.33%** **107.49%** 0.72× Dominated	**103.33%** **107.49%** 0.72× Dominated	**103.33%** **107.49%** 6.29× Dominated	**0/5**
UMAP	**95.93%** **92.90%** 12.39× Dominated	**95.93%** **92.90%** 9.68× Dominated	**95.93%** **92.90%** 3.75× Dominated	**95.93%** **92.90%** 3.75× Dominated	**95.93%** **92.90%** 7.72× Dominated	**0/5**
PaCMAP	**91.26%** **90.21%** 28.74× Dominated	**91.26%** **90.21%** 31.16× **PF**	**91.26%** **90.21%** 0.25× Dominated	**91.26%** **90.21%** 0.25× Dominated	**91.26%** **90.21%** 13.67× Dominated	**1/5**
LocalMAP	**90.51%** **87.97%** 32.76× **PF**	**90.51%** **87.97%** 31.98× **PF**	**90.51%** **87.97%** N/A	**90.51%** **87.97%** N/A	**90.51%** **87.97%** 14.89× Dominated	**2/5**
ParamRepulsor	**90.56%** **88.12%** 9.98× Dominated	**90.56%** **88.12%** 6.52× Dominated	**90.56%** **88.12%** 1.11× Dominated	**90.56%** **88.12%** 1.11× Dominated	**90.56%** **88.12%** 7.66× Dominated	**0/5**
Kernel PCA	**101.94%** **101.34%** 28.05× Dominated	**101.94%** **101.34%** 16.11× Dominated	**101.94%** **101.34%** 2.91× Dominated	**101.94%** **101.34%** 2.91× Dominated	**101.94%** **101.34%** 35.08× **PF**	**1/5**
NSC-pSP	**103.66%** **108.48%** 12.09× Dominated	**103.66%** **108.48%** 9.19× Dominated	**103.66%** **108.48%** 1.71× Dominated	**103.66%** **108.48%** 1.71× Dominated	**103.66%** **108.48%** 7.84× Dominated	**0/5**

PF denotes membership in the non-dominated Pareto front. For two compression methods a and b, method a dominates method b for a given systems criterion if it achieves both "equal or higher" predictive retention and "equal or higher" efficiency, with at least one strict improvement, under both ROC-AUC and Accuracy. Inference and fit are Raw-relative speedups. Energy efficiency is defined as $E_{\text{Raw}}/E_{\text{method}}$, CO_2 efficiency is defined as $C_{\text{Raw}}/C_{\text{method}}$, and GPU-memory efficiency is defined as $M_{\text{Raw}}/M_{\text{method}}$ using NVML peak-used memory; thus, larger values are preferred. Green cells indicate Pareto-optimal operating points. The SCOPE row remains blue for visual consistency with the other tables. Gold, silver, and bronze in the final column indicate the three highest distinct PF-Coverage values.

in six of the ten repetitions for each metric. DB exhibits greater repetition-level variability, with SCOPE improving in five repetitions. These results indicate that the aggregate performance of SCOPE is not driven by a single favorable repetition.

Overall block-model performance. Table K.2 summarizes performance over the 10 repetitions. SCOPE achieves the highest mean linear accuracy of 0.8518, the highest kNN accuracy of 0.7363,



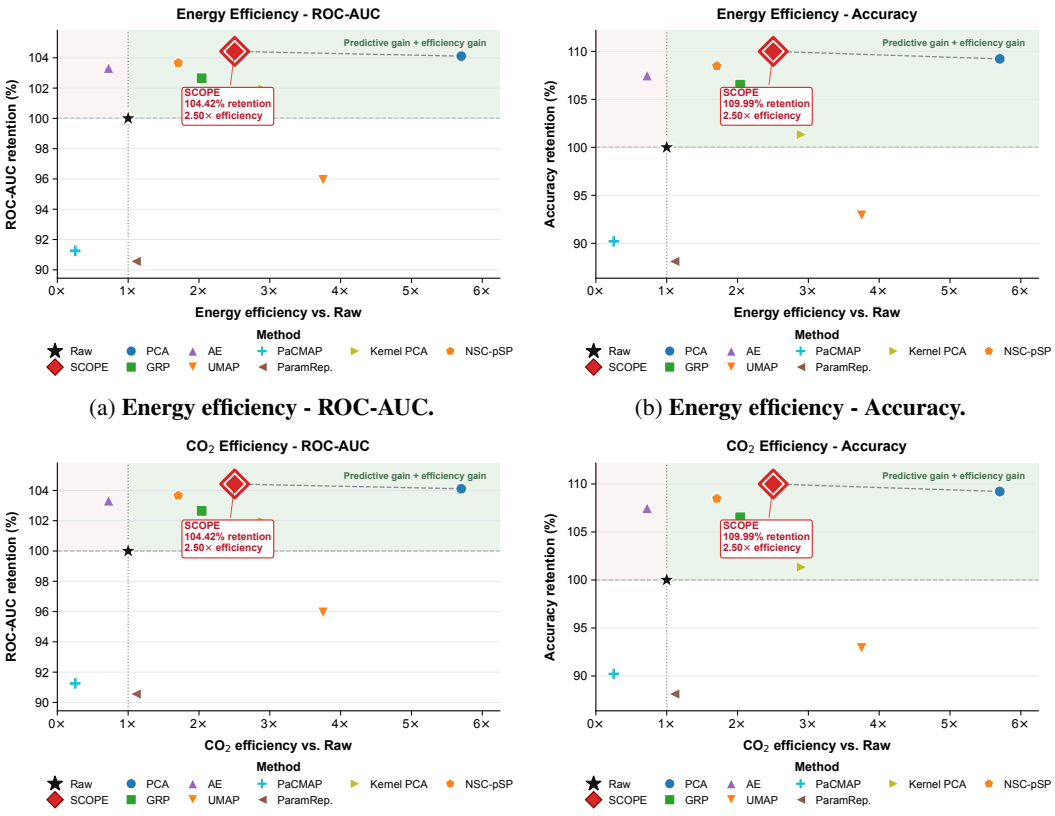
(a) **Inference efficiency - ROC-AUC.** (b) **Inference efficiency - Accuracy.**

(c) **Fit efficiency - ROC-AUC.** (d) **Fit efficiency - Accuracy.**

(e) **GPU-memory efficiency - ROC-AUC.** (f) **GPU-memory efficiency - Accuracy.**

Figure H.1: **Predictive fidelity versus computational efficiency with TabICLv2.** Left panels report ROC-AUC retention and right panels report Accuracy retention relative to the uncompressed Raw representation.The rows respectively evaluate inference speedup, fit speedup, and NVML GPU-memory efficiency. The horizontal and vertical reference lines denote Raw predictive performance and Raw computational cost, respectively. The upper-right shaded region therefore represents simultaneous predictive improvement and computational-efficiency improvement. Dashed curves connect non-dominated operating points on the corresponding Pareto front.

and the lowest DB score of 4.0977 among all evaluated methods. It also remains highly competitive in silhouette score at 0.0450, compared with 0.0460 for NSC-pSP (Habib et al., 2026b). Thus, under a common representation size of $M = 32$, SCOPE provides the strongest aggregate performance on three of the four evaluation criteria while remaining close to the best result on the remaining metric.



(a) **Energy efficiency - ROC-AUC.**

(b) **Energy efficiency - Accuracy.**

(c) CO_2 **efficiency - ROC-AUC.**

(d) CO_2 **efficiency - Accuracy.**

Figure H.2: **Predictive fidelity versus sustainability-oriented efficiency with TabICLv2.** Left panels report ROC-AUC retention an right panels report Accuracy retention. Energy and CO_2 efficiencies are expressed relative to Raw, such that values above $1\times$ indicate reduced resource consumption. The upper-right shaded region indicates simultaneous predictive gain and resource-efficiency gain, while dashed curves connect non-dominated Pareto-optimal operating points.

Table I.1: High-dimensional stress test of TabPFN-family checkpoints on eight HDLSS datasets. The superscript $+$ indicates that `ignore_pretraining_limits=True` was used, forcing the corresponding checkpoint beyond its configured feature limits. LF denotes the specialized RealTabPFN-2.5 large feature checkpoint. Results are reported as mean accuracy (%) $\pm$ standard deviation over repeated cross-validation runs. These out-of-scope configurations are presented separately from the primary in-scope leaderboard. Among the evaluated checkpoints, 🥇 gold, 🥈 silver, and 🥉 bronze cells indicate the first, second, and third highest distinct mean accuracies for each dataset, respectively. Tied methods at the displayed precision receive the same rank.

Model	Colon	Lung	GLI	SMK	ALLAML	Prostate	Arcene	TOX
TabPFN v2$^+$	85.90 ± 11.10	95.96 ± 2.60	85.88 ± 5.52	**73.02 ± 5.27**	95.81 ± 5.25	**94.90 ± 4.70**	83.50 ± 5.48	93.33 ± 4.32
TabPFN-2.5$^+$	86.26 ± 10.68	96.46 ± 2.52	**90.82 ± 5.28**	71.62 ± 6.87	96.67 ± 4.44	93.50 ± 5.68	89.90 ± 4.82	**94.51 ± 3.60**
TabPFN-2.6$^+$	86.54 ± 10.32	**96.65 ± 2.60**	89.65 ± 5.59	71.63 ± 6.97	96.91 ± 4.88	94.89 ± 4.25	88.70 ± 4.25	93.69 ± 4.59
TabPFN-3$^+$	**86.87 ± 10.34**	96.55 ± 2.21	88.94 ± 5.60	70.55 ± 6.36	**97.22 ± 4.39**	92.33 ± 4.82	**90.70 ± 4.39**	89.59 ± 5.51
RealTabPFN-2.5$^+$	86.26 ± 10.68	96.46 ± 2.52	**90.82 ± 5.28**	71.52 ± 6.42	96.67 ± 4.44	93.50 ± 5.68	89.80 ± 4.74	94.40 ± 3.56
RealTabPFN-2.5-LF$^+$	86.26 ± 10.68	96.46 ± 2.52	**90.82 ± 5.28**	71.30 ± 7.05	96.95 ± 4.41	93.50 ± 5.68	90.00 ± 4.69	94.40 ± 3.93

Gold, silver, and bronze indicate the top three distinct mean accuracies within each dataset column. SMK abbreviates SMK_CAN_187, and Prostate abbreviates Prostate-GE.

Statistical analysis. Table K.3 further evaluates whether the observed differences are systematic across repetitions. The Friedman tests (Friedman, 1937) reject equal performance among the ten methods for all four metrics, with $p < 10^{-5}$ in every case. Pairwise one-sided Wilcoxon

Table J.1: Datasets used in the experiments and their references.

Dataset	Source	Dataset	Source
Cell Cycle	Mahdessian et al. (2021); NCBI (2021)	orlraws10P	Li et al. (2018; 2026)
OV	Yang et al. (2025)	ORL	Li et al. (2018; 2026)
LGG	Yang et al. (2025)	DrivFace	Hernández-Sabat et al. (2016)
GBM	Yang et al. (2025)	COIL20	Li et al. (2018; 2026)
COAD	Yang et al. (2025)	Colon	Li et al. (2018; 2026)
BRCA	Yang et al. (2025)	Lung	Li et al. (2018; 2026)
RELATHE	Li et al. (2018; 2026)	GLI	Li et al. (2018; 2026)
PCMAC	Li et al. (2018; 2026)	SMK	Li et al. (2018; 2026)
dbworld_bodies	Filannino (2011a;b)	ALLAML	Li et al. (2018; 2026)
Prostate	Li et al. (2018; 2026)	Arcene	Li et al. (2018; 2026)
Biblical_Texts	Sah & Fokoué (2019); UCI (2019)	TOX	Li et al. (2018; 2026)
BASEHOCK	Li et al. (2018; 2026)	Lymphoma	Li et al. (2018; 2026)
Yale	Li et al. (2018; 2026)	Leukemia	Li et al. (2018; 2026)
warpPIE10P	Li et al. (2018; 2026)	GLIOMA	Li et al. (2018; 2026)
warpAR10P	Li et al. (2018; 2026)	CLL	Li et al. (2018; 2026)

tests (Demšar, 2006) show that the differences between SCOPE and NSC-pSP (Habib et al., 2026b) do not reach the nominal $p < 0.05$ threshold with only 10 repetitions, although the kNN comparison approaches this threshold with $p = 0.05273$. In contrast, SCOPE shows significant improvements over several alternative compressors, including PCA (Pearson, 1901), NSC-P (Habib et al., 2026b), and RP (Johnson & Lindenstrauss, 1984) across all four criteria, supporting the competitiveness of the proposed representation while avoiding an unsupported significance claim over NSC-pSP (Habib et al., 2026b).

Competitiveness against standard baselines. Table K.4 provides paired effect and ranking analyses against PCA (Pearson, 1901), AE (Bourlard & Kamp, 1988), UMAP (**?**), and PaCMAP (Wang et al., 2021). SCOPE attains positive mean improvements and better average ranks in all 16 metric-baseline comparisons. For linear accuracy, both the paired-bootstrap intervals (Efron, 1992) and two-sided Wilcoxon tests (Demšar, 2006) support improvements over all four baselines. The same joint criterion supports SCOPE over PCA (Pearson, 1901) for kNN accuracy, over PCA (Pearson, 1901) and AE (Bourlard & Kamp, 1988) for silhouette, and over PCA (Pearson, 1901) and PaCMAP (Wang et al., 2021) for DB. The remaining comparisons retain positive mean differences but exhibit greater repetition-level uncertainty, providing a more conservative view of SCOPE's advantage under the synthetic HDLSS setting.

L Detailed Ablation Analysis

SCOPE component ablation. Table L.1 shows that removing individual structural components produces only modest ROC-AUC changes on SMK, with all paired corrected confidence intervals overlapping zero. The ablation nevertheless reveals substantial differences in representation efficiency. In particular, replacing entropy-based spectral calibration with the positive spectral rank increases the mean representation size from 720.4 to 1723.6 dimensions, reducing compression from $27.76\times$ to $11.60\times$ and increasing reducer and inference cost. Native and random feature orders retain the same compression budget but alter predictive performance, while equal-width segmentation and mean pooling remain close to the complete method on this dataset.

LENS stage ablation. Table L.2 further decomposes the LENS ordering procedure. The medoid nearest-neighbor initialization attains the lowest runtime, whereas TSP refinement reduces the TSP objective and the complete LENS pipeline achieves the lowest MinLA cost while remaining within its TSP budget.

Table K.1: Per-repetition results on the block-structured synthetic HDLSS benchmark with $M = 32$. The synthetic construction follows the block-subunit style of BSTabDiff (Habib et al., 2026a), while the block-structured HDLSS evaluation follows NSC. SCOPE denotes our proposed self-calibrated compression method, whereas NSC-pSP denotes the prior variant from GOTabPFN (Habib et al., 2026b). ↑ indicates higher is better and ↓ indicates lower is better. For SCOPE, ✅ green, 〓 gray, and ❌ orange cells indicate improvement, tie, and degradation relative to NSC-pSP, respectively.

Method	Dim.	Lin. ↑	kNN ↑	Sil. ↑	DB ↓	Method	Dim.	Lin. ↑	kNN ↑	Sil. ↑	DB ↓
		Rep0						**Rep5**			
SCOPE (ours)	32	**0.850**	0.675	0.047	**3.240**	**SCOPE (ours)**	32	0.800	0.663	0.050	4.402
NSC-pSP (prior)	32	0.825	0.675	0.047	3.802	NSC-pSP (prior)	32	0.875	0.688	0.064	3.577
NSC-SP	32	0.825	0.662	0.044	3.963	NSC-SP	32	0.838	0.788	0.048	3.729
NSC	32	0.775	0.713	0.035	4.291	NSC	32	0.775	0.625	0.030	4.717
NSC-P	32	0.825	0.688	0.038	4.130	NSC-P	32	0.600	0.588	0.021	5.910
PCA	32	0.725	0.588	0.035	4.354	PCA	32	0.713	0.562	0.028	4.842
AE	32	0.762	0.725	0.041	4.094	AE	32	0.738	0.700	0.036	4.433
RP	32	0.713	0.650	0.038	4.098	RP	32	0.700	0.650	0.030	4.365
UMAP	32	0.789	0.652	0.068	3.502	UMAP	32	0.684	0.639	0.001	5.213
PaCMAP	32	0.776	0.742	0.116	3.783	PaCMAP	32	0.697	0.584	-0.004	5.677
		Rep1						**Rep6**			
SCOPE (ours)	32	**0.843**	**0.738**	**0.047**	4.770	**SCOPE (ours)**	32	**0.863**	**0.763**	0.040	4.156
NSC-pSP (prior)	32	0.825	0.688	0.043	3.986	NSC-pSP (prior)	32	0.825	0.675	0.053	3.684
NSC-SP	32	0.762	0.775	0.045	3.984	NSC-SP	32	0.863	0.725	0.058	3.525
NSC	32	0.713	0.588	0.034	4.493	NSC	32	0.788	0.625	0.034	4.525
NSC-P	32	0.675	0.613	0.028	5.008	NSC-P	32	0.750	0.575	0.023	5.688
PCA	32	0.763	0.625	0.032	4.643	PCA	32	0.775	0.675	0.030	4.596
AE	32	0.725	0.638	0.036	4.418	AE	32	0.825	0.625	0.032	4.562
RP	32	0.588	0.550	0.022	5.119	RP	32	0.812	0.675	0.028	4.715
UMAP	32	0.736	0.612	-0.006	5.827	UMAP	32	0.855	0.811	0.088	3.398
PaCMAP	32	0.749	0.637	-0.007	5.720	PaCMAP	32	0.842	0.847	0.061	4.598
		Rep2						**Rep7**			
SCOPE (ours)	32	**0.900**	**0.813**	**0.053**	**3.687**	**SCOPE (ours)**	32	0.838	0.775	0.037	4.193
NSC-pSP (prior)	32	0.875	0.713	0.046	3.886	NSC-pSP (prior)	32	0.850	0.800	0.071	3.034
NSC-SP	32	0.862	0.800	0.053	3.539	NSC-SP	32	0.775	0.650	0.034	4.357
NSC	32	0.725	0.600	0.025	4.786	NSC	32	0.850	0.800	0.057	3.628
NSC-P	32	0.613	0.575	0.022	5.891	NSC-P	32	0.738	0.613	0.028	4.858
PCA	32	0.750	0.600	0.030	4.690	PCA	32	0.838	0.675	0.030	4.608
AE	32	0.688	0.625	0.030	4.775	AE	32	0.850	0.825	0.036	4.144
RP	32	0.688	0.538	0.026	4.801	RP	32	0.788	0.688	0.028	4.648
UMAP	32	0.894	0.797	0.091	3.423	UMAP	32	0.723	0.705	0.021	4.561
PaCMAP	32	0.868	0.834	0.047	4.643	PaCMAP	32	0.684	0.649	0.029	4.820
		Rep3						**Rep8**			
SCOPE (ours)	32	0.838	**0.738**	**0.049**	4.638	**SCOPE (ours)**	32	0.850	**0.763**	**0.044**	**4.041**
NSC-pSP (prior)	32	0.850	0.725	0.031	4.630	NSC-pSP (prior)	32	0.850	0.650	0.038	4.283
NSC-SP	32	0.900	0.788	0.041	4.176	NSC-SP	32	0.725	0.775	0.045	3.958
NSC	32	0.775	0.662	0.035	4.517	NSC	32	0.725	0.725	0.048	3.904
NSC-P	32	0.725	0.588	0.019	6.260	NSC-P	32	0.863	0.700	0.030	4.636
PCA	32	0.775	0.600	0.030	4.672	PCA	32	0.762	0.613	0.028	4.745
AE	32	0.800	0.763	0.034	4.493	AE	32	0.763	0.762	0.030	4.739
RP	32	0.688	0.600	0.023	5.047	RP	32	0.650	0.625	0.018	5.819
UMAP	32	0.696	0.666	0.021	4.404	UMAP	32	0.762	0.719	0.038	4.193
PaCMAP	32	0.670	0.584	0.037	4.591	PaCMAP	32	0.828	0.782	0.089	4.093
		Rep4						**Rep9**			
SCOPE (ours)	32	**0.875**	0.663	**0.041**	**3.604**	**SCOPE (ours)**	32	**0.863**	**0.775**	**0.042**	**4.248**
NSC-pSP (prior)	32	0.863	0.675	0.033	4.437	NSC-pSP (prior)	32	0.850	0.688	0.034	4.579
NSC-SP	32	0.812	0.750	0.032	4.620	NSC-SP	32	0.762	0.762	0.033	4.527
NSC	32	0.763	0.625	0.033	4.596	NSC	32	0.750	0.625	0.029	4.755
NSC-P	32	0.675	0.613	0.024	5.742	NSC-P	32	0.600	0.562	0.019	6.010
PCA	32	0.725	0.650	0.029	4.729	PCA	32	0.775	0.600	0.026	4.853
AE	32	0.712	0.650	0.031	4.631	AE	32	0.725	0.638	0.030	4.658
RP	32	0.688	0.612	0.020	5.389	RP	32	0.712	0.575	0.018	5.266
UMAP	32	0.802	0.744	0.038	4.196	UMAP	32	0.723	0.705	0.059	3.659
PaCMAP	32	0.723	0.715	0.037	4.748	PaCMAP	32	0.737	0.677	0.005	5.489

Green, gray, and orange cells in the SCOPE rows denote improvement, tie, and degradation relative to NSC-pSP, respectively. Across the ten repetitions and four evaluation criteria, SCOPE records 23 improvements, 3 ties, and 14 degradations relative to NSC-pSP.

Representation visualization. Fig. L.1 and Table L.3 show that SCOPE substantially compresses SMK from 19,993 features to 753 dimensions while preserving the maximum centered sample rank of 186. Although both SCOPE and NSC-pSP (Habib et al., 2026b) produce elongated embeddings, their spectral behavior differs markedly: NSC-pSP (Habib et al., 2026b) concentrates 95.03% of the variance in its first principal direction and 96.66% in the first two, whereas SCOPE reduces this concentration to 66.20% and 77.02%, respectively. Thus, SCOPE retains considerably more variance beyond the dominant directions while maintaining a compact representation, indicating that

Table K.2: Overall results on the block-structured synthetic HDLSS benchmark with $M = 32$, averaged over 10 independent repetitions. The synthetic construction follows the block-subunit style of BSTabDiff, while the block-structured HDLSS evaluation follows NSC. SCOPE denotes our proposed self-calibrated compression method, whereas NSC-pSP denotes the prior variant from GOTabPFN (Habib et al., 2026b). ↑ indicates higher is better and ↓ indicates lower is better. For SCOPE, ✅ green and ❌ orange cells indicate improvement and degradation relative to NSC-pSP, respectively.

Method	Dim.	Lin. Acc. ↑	kNN Acc. ↑	Sil. ↑	DB ↓
SCOPE (ours)	32	**0.8518 ± 0.0264**	**0.7363 ± 0.0525**	0.0450 ± 0.0049	**4.0977 ± 0.4740**
NSC-pSP (prior)	32	0.8488 ± 0.0406	0.7063 ± 0.0590	**0.0460 ± 0.0257**	4.1198 ± 0.9657
NSC-SP	32	0.8263 ± 0.0542	0.7313 ± 0.0722	0.0432 ± 0.0209	4.1806 ± 0.8407
AE	32	0.7688 ± 0.0652	0.6950 ± 0.0825	0.0340 ± 0.0102	4.4959 ± 0.5068
UMAP	32	0.7663 ± 0.0651	0.7050 ± 0.0621	0.0415 ± 0.0321	4.2374 ± 0.7654
NSC	32	0.7638 ± 0.0614	0.6538 ± 0.0932	0.0348 ± 0.0187	4.5613 ± 1.1626
PCA	32	0.7588 ± 0.0553	0.6188 ± 0.0659	0.0296 ± 0.0071	4.7186 ± 0.4223
PaCMAP	32	0.7575 ± 0.0657	0.7050 ± 0.0907	0.0410 ± 0.0375	4.8160 ± 0.6112
NSC-P	32	0.7263 ± 0.1021	0.6225 ± 0.0671	0.0270 ± 0.0125	5.0121 ± 1.5534
RP	32	0.7163 ± 0.0921	0.6113 ± 0.0781	0.0251 ± 0.0108	5.0218 ± 0.7847

Values are mean ± standard deviation over 10 repetitions. Bold indicates the best mean across all methods. Green and orange cells in the SCOPE row indicate improvement and degradation relative to NSC-pSP, respectively.

its ordered local projections preserve a richer multidimensional structure rather than concentrating nearly all variation along a single latent axis.

Scalability of matrix-factorization baselines. Table L.4 highlights an important limitation of conventional Sparse PCA (Zou et al., 2006) and NMF (Lee & Seung, 1999) in these very high-dimensional HDLSS data: their standard full-batch formulations become computationally impractical at nearly 20,000 input features, requiring faster mini-batch approximations for repeated evaluation. Even with these approximations, Mini-Batch Sparse PCA and Mini-Batch NMF require 20.94 and 18.32 seconds per fold for dimensionality reduction, respectively, compared with only 0.283 seconds for SCOPE. Thus, while the mini-batch variants attain comparable predictive performance, SCOPE constructs its representation approximately 74× faster than Mini-Batch Sparse PCA and 65× faster than Mini-Batch NMF, illustrating its substantially lower preprocessing burden in the HDLSS regime.

Feature selection vs. feature compression. Table L.5 compares SCOPE with two feature selection controls matched to its fold-specific output dimension. Supervised ANOVA-k attains the highest point-estimate ROC-AUC of 76.91, but requires training-fold labels to determine which features are retained. Label-free Random-k achieves 75.16, closely matching SCOPE at 75.07, indicating that aggressive feature removal alone can remain competitive on SMK. The methods nevertheless represent fundamentally different reductions: feature selection permanently discards approximately 96.4% of the original features, whereas SCOPE uses all 19,993 input features to construct 720.4 structured meta-features on average. SCOPE therefore provides a label-free compressed representation of the complete feature space while retaining comparable predictive performance and the lowest downstream inference time among the three variants.

BeyondArena HDLSS subset and controlled Elo comparison. To evaluate SCOPE under the high-dimensional regime targeted in this work, we curated a subset of BeyondArena (Purucker et al., 2026) using the same dimensionality criterion adopted throughout our analysis in Sec. 3.3 and Sec. D, retaining only datasets satisfying $m > n$, where m and n denote the numbers of features and samples, respectively. This filtering yields six datasets: *clock protein toxicity*, *lung cancer*, *lung cancer epithelial gene expression*, *micro mass*, *pancreatic cancer mouse detection*, and *prostate cancer detection*. Rather than mixing our results with pre-computed BeyondArena (Purucker et al., 2026) baselines that may follow different internal fitting configurations, we perform a controlled

Table K.3: Nonparametric comparisons on the block-structured synthetic HDLSS benchmark ($M = 32$, 10 repetitions). Friedman tests compare all 10 methods, while paired one-sided Wilcoxon signed-rank tests use **SCOPE** as the reference. Higher values are preferred for linear accuracy, kNN accuracy, and silhouette, whereas lower values are preferred for Davies-Bouldin (DB). For the Wilcoxon comparisons, ✅ green p-value cells indicate nominal significance at $p < 0.05$, while ❌ orange cells indicate $p \geq 0.05$. Gray rows denote omnibus Friedman tests.

Accuracy metrics			Clustering metrics		
Comparison	**Stat.**	p	**Comparison**	**Stat.**	p
Friedman (linear; 10 methods)	43.922	**1.459e-06**	**Friedman (silhouette; 10 methods)**	41.404	**4.222e-06**
SCOPE > NSC-pSP (linear)	31.0	0.1777	**SCOPE** > NSC-pSP (sil.)	30.0	0.4229
SCOPE > NSC-SP (linear)	45.5	**0.03418**	**SCOPE** > NSC-SP (sil.)	42.0	0.08008
SCOPE > NSC (linear)	54.0	**0.001953**	**SCOPE** > NSC (sil.)	46.0	**0.03223**
SCOPE > NSC-P (linear)	54.0	**0.001953**	**SCOPE** > NSC-P (sil.)	55.0	**0.0009766**
SCOPE > PCA (linear)	54.0	**0.001953**	**SCOPE** > PCA (sil.)	55.0	**0.0009766**
SCOPE > AE (linear)	54.0	**0.001953**	**SCOPE** > AE (sil.)	55.0	**0.0009766**
SCOPE > RP (linear)	55.0	**0.0009766**	**SCOPE** > RP (sil.)	55.0	**0.0009766**
SCOPE > UMAP (linear)	55.0	**0.0009766**	**SCOPE** > UMAP (sil.)	31.0	0.3848
SCOPE > PaCMAP (linear)	55.0	**0.0009766**	**SCOPE** > PaCMAP (sil.)	33.0	0.3125
Friedman (kNN; 10 methods)	40.028	**7.510e-06**	**Friedman (DB; 10 methods)**	48.196	**2.347e-07**
SCOPE > NSC-pSP (kNN)	36.5	0.05273	**SCOPE** < NSC-pSP (DB)	31.0	0.6523
SCOPE > NSC-SP (kNN)	22.5	0.7012	**SCOPE** < NSC-SP (DB)	30.0	0.6152
SCOPE > NSC (kNN)	49.0	**0.01367**	**SCOPE** < NSC (DB)	13.0	0.08008
SCOPE > NSC-P (kNN)	54.0	**0.001953**	**SCOPE** < NSC-P (DB)	0.0	**0.0009766**
SCOPE > PCA (kNN)	55.0	**0.0009766**	**SCOPE** < PCA (DB)	2.0	**0.002930**
SCOPE > AE (kNN)	37.0	0.1797	**SCOPE** < AE (DB)	9.0	**0.03223**
SCOPE > RP (kNN)	55.0	**0.0009766**	**SCOPE** < RP (DB)	1.0	**0.001953**
SCOPE > UMAP (kNN)	41.0	0.09277	**SCOPE** < UMAP (DB)	20.0	0.2461
SCOPE > PaCMAP (kNN)	39.0	0.1377	**SCOPE** < PaCMAP (DB)	1.0	**0.001953**

Green and orange cells denote nominally significant ($p < 0.05$) and non-significant ($p \geq 0.05$) tests, respectively. Blue comparison cells identify SCOPE as the reference method. Gray rows denote omnibus Friedman tests.

comparison between Raw TabPFN-3 (Grinsztajn et al., 2026) and SCOPE + TabPFN-3 only, with both methods evaluated under the same BeyondArena tiny-data protocol using 5×5 internal validation/bagging (25 fits per outer task). Pairwise Elo (Erickson et al., 2025) is then computed exclusively from the 43 shared BeyondArena outer tasks generated by these six datasets. Raw TabPFN-3 obtains a pairwise Elo of 1018.9, whereas SCOPE + TabPFN-3 obtains 981.1, corresponding to a modest Elo decrease of 37.8 points. In return, SCOPE reduces the median training time from 227.9 s to 162.6 s and the median inference time from 3.53 s to 2.15 s, yielding $1.40\times$ and $1.64\times$ speedups, respectively. These results highlight the intended role of SCOPE as an efficiency-oriented representation layer for TabPFN-3, substantially reducing computational cost while retaining competitive predictive performance.

Compressor configurations. All user-specified compressor hyperparameters were fixed across datasets, with no downstream validation-based selection. Only data-dependent quantities, primarily the output dimension, varied across folds according to the rules in Table L.6. For dimensionality-controlled comparisons, most baselines used SCOPE's fold-specific dimension as the reference, subject to method-specific rank or dimensionality constraints. Stochastic compressors used a fixed seed of 42. All reducers were fitted using training-fold data only; PLS (Wold et al., 1984) was the only supervised compressor and used training labels. For fairness, we evaluate NSC-pSP (Habib et al., 2026b) under a single fixed configuration across all datasets, avoiding the dataset-specific hyperparameter selection used in the original formulation and ensuring a consistent evaluation protocol across compressors.

Table K.4: Additional competitiveness analyses of SCOPE against PCA, AE, UMAP, and PaCMAP on the block-structured synthetic HDLSS benchmark ($M = 32$, 10 repetitions). For linear accuracy, kNN accuracy, and silhouette, $\Delta = \mathrm{SCOPE} - \mathrm{baseline}$; for Davies-Bouldin (DB), $\Delta = \mathrm{baseline} - \mathrm{SCOPE}$, so that $\Delta > 0$ always favors SCOPE. Confidence intervals are paired bootstrap percentile intervals for the mean improvement, and W/T/L denotes wins/ties/losses. Wilcoxon tests are paired and two-sided, while r_{rb} denotes the matched-pairs rank-biserial effect size. Average ranks are computed over all 10 methods, with rank 1 denoting the best result. For the pairwise comparisons, ✅ green cells indicate cases where the 95% confidence interval excludes zero and the two-sided Wilcoxon test yields $p < 0.05$, whereas ❌ orange cells indicate comparisons that do not satisfy both criteria.

Metric	Base	Δ mean	95% CI	W/T/L	Wilcoxon p (2s)	r_{rb}	Avg. rank (SCOPE/base)
Lin. Acc.	PCA	0.0917	**[0.0648, 0.1164]**	9/0/1	**0.003906**	0.964	2.05/5.80
Lin. Acc.	AE	0.0930	**[0.0549, 0.1328]**	9/0/1	**0.003906**	0.964	2.05/6.00
Lin. Acc.	UMAP	0.0854	**[0.0555, 0.1126]**	10/0/0	**0.001953**	1.000	2.05/6.05
Lin. Acc.	PaCMAP	0.0944	**[0.0613, 0.1268]**	10/0/0	**0.001953**	1.000	2.05/6.40
kNN Acc.	PCA	0.1175	**[0.0849, 0.1499]**	10/0/0	**0.001953**	1.000	3.25/7.85
kNN Acc.	AE	0.0412	[-0.0100, 0.0961]	6/0/4	0.359375	0.345	3.25/4.20
kNN Acc.	UMAP	0.0313	[-0.0058, 0.0657]	8/0/2	0.185547	0.491	3.25/4.90
kNN Acc.	PaCMAP	0.0312	[-0.0209, 0.0832]	5/0/5	0.275391	0.418	3.25/4.50
Sil.	PCA	0.0152	**[0.0123, 0.0182]**	10/0/0	**0.001953**	1.000	2.50/7.40
Sil.	AE	0.0114	**[0.0081, 0.0148]**	10/0/0	**0.001953**	1.000	2.50/5.60
Sil.	UMAP	0.0031	[-0.0170, 0.0235]	6/0/4	0.769531	0.127	2.50/4.95
Sil.	PaCMAP	0.0040	[-0.0204, 0.0269]	7/0/3	0.625000	0.200	2.50/5.10
DB	PCA	0.5755	**[0.3212, 0.8234]**	9/0/1	**0.005859**	0.927	3.50/7.30
DB	AE	0.3970	[0.0952, 0.6977]	7/0/3	0.064453	0.673	3.50/5.20
DB	UMAP	0.1399	[-0.2092, 0.4894]	6/0/4	0.492188	0.273	3.50/3.80
DB	PaCMAP	0.7185	**[0.4324, 0.9885]**	9/0/1	**0.003906**	0.964	3.50/6.90

Blue cells identify SCOPE-centered improvement or ranking quantities. Green cells indicate comparisons supported by both a strictly positive 95% paired-bootstrap interval and a two-sided Wilcoxon $p < 0.05$. Orange cells indicate comparisons for which at least one of these conditions is not satisfied.

Table L.1: **Component ablation of SCOPE on SMK with TabPFN-3.** Results are mean ± standard deviation over 5×5 repeated cross-validation. Δ denotes $\mathrm{AUC}_{\mathrm{SCOPE}} - \mathrm{AUC}_{\mathrm{variant}}$, so positive values favor full SCOPE. Intervals are 95% corrected cross-validation confidence intervals for the paired difference. M denotes the selected representation dimension and Comp. the input-to-output compression ratio. The full SCOPE configuration is highlighted in blue.

Variant	ROC-AUC	Δ	Corrected 95% CI	M	Comp.	Reducer / Inference (s)
SCOPE (full)	0.7507 ± 0.0760	0.0000	[0.0000, 0.0000]	720.4 ± 10.8	27.76×	0.283/0.481
w/o LENS (native order)	0.7421 ± 0.0735	0.0086	$[-0.0243, 0.0415]$	720.4 ± 10.8	27.76×	0.265/0.480
Random order	0.7389 ± 0.0730	0.0118	$[-0.0465, 0.0339]$	720.4 ± 10.8	27.76×	0.273/0.503
w/o spectral calibration	0.7456 ± 0.0780	0.0051	$[-0.0367, 0.0269]$	1723.6 ± 3.0	11.60×	0.507/0.573
w/o transition segmentation	0.7413 ± 0.0748	0.0094	$[-0.0309, 0.0298]$	720.4 ± 10.8	27.76×	0.272/0.496
w/o principal projection	0.7443 ± 0.0747	0.0064	$[-0.0278, 0.0179]$	720.4 ± 10.8	27.76×	0.130/0.503

Gray cells indicate paired differences whose corrected 95% interval includes zero. Orange highlights the substantially larger representation and computational cost obtained without spectral calibration. Green marks the lowest reducer time.

Table L.2: **Stage-wise ablation of LENS on SMK.** Each ordering is supplied to the same SCOPE compression pipeline and evaluated with TabPFN-3 under 5×5 repeated cross-validation. Higher ROC-AUC is better, whereas lower TSP cost, MinLA cost, and cumulative ordering time are better. All stages yield the same SCOPE representation dimension and compression ratio. The complete LENS configuration is highlighted in blue.

Ordering stage	ROC-AUC	M	Comp.	TSP Cost ↓	MinLA Cost ↓	Ordering Time (s) ↓
LENS: Medoid NN	0.7483 ± 0.0675	720.4	$27.76\times$	205148.16	2.5039×10^{13}	**7.17**
LENS: + TSP 2-opt	0.7488 ± 0.0676	720.4	$27.76\times$	**203961.69**	2.4927×10^{13}	235.22
Full LENS	0.7507 ± 0.0760	720.4	$27.76\times$	203971.84	2.4781×10^{13}	370.73

Gold highlights the best value for each varying criterion. The TSP refinement achieves the lowest TSP cost, while the complete LENS pipeline obtains the lowest MinLA cost under its self-calibrated TSP budget.

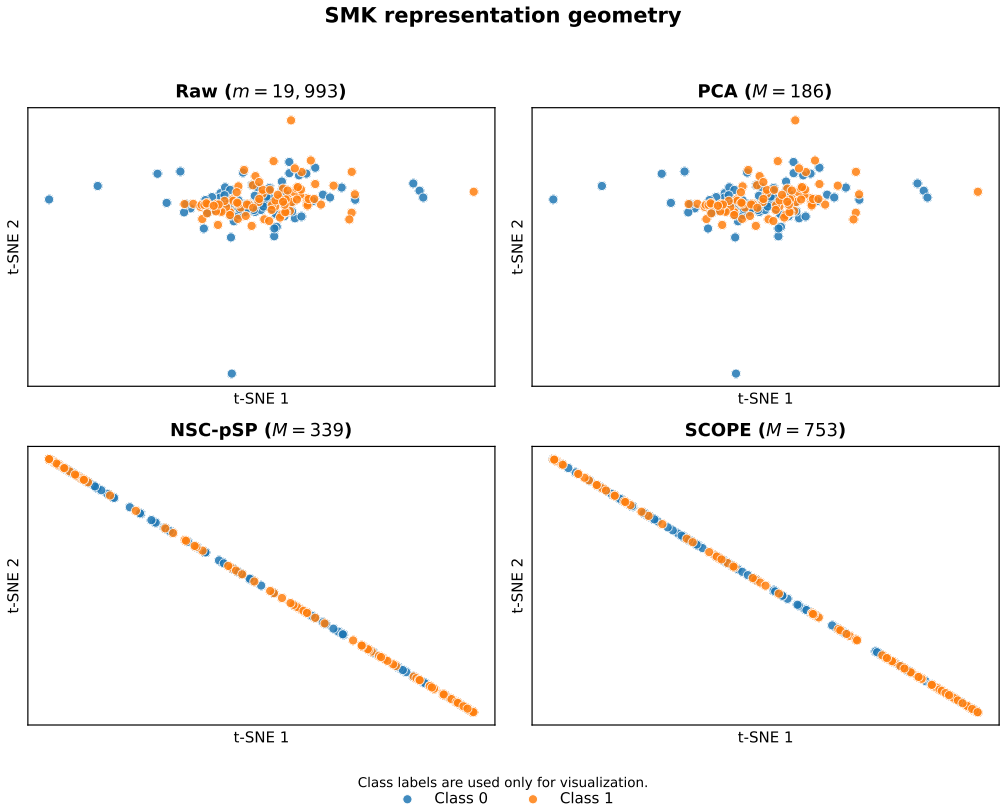


Figure L.1: **t-SNE visualization of SMK representation geometry.** Raw, PCA, NSC-pSP, and SCOPE representations are embedded using identical t-SNE settings, with class labels used only for coloring. Raw and PCA retain a comparatively diffuse geometry, whereas NSC-pSP and SCOPE exhibit strongly elongated embeddings. This behavior reflects spectral concentration rather than rank collapse, since all representations retain the maximum centered sample rank of 186. The corresponding variance concentration is quantified in Table L.3.

Table L.3: **Spectral concentration of SMK representations.** All representations retain the maximum centered sample rank of 186. PC1 and PC1+PC2 report the fraction of total representation variance captured by the leading one and two principal directions, respectively.

Representation	Dim.	Rank	PC1 (%)	PC1+PC2 (%)
Raw	19,993	186	36.67	45.46
PCA	186	186	36.67	45.46
NSC-pSP	339	186	95.03	96.66
SCOPE	753	186	**66.20**	**77.02**

Table L.4: **Scalability ablation of matrix-factorization baselines on SMK with TabPFN-3.** Standard full-batch SparsePCA and NMF are computationally impractical for repeated evaluation at $m = 19{,}993$, so we additionally evaluate their faster mini-batch approximations using fixed configurations without HPO. ROC-AUC is reported as mean $\pm$ standard deviation over 5×5 repeated cross-validation. M denotes the output dimension, Comp. the input-to-output compression ratio, and Reducer the dimensionality-reduction runtime per fold. Reducer speedup is measured relative to the corresponding mini-batch method. SCOPE is highlighted in blue.

Method	Variant	ROC-AUC	M	Comp.	Reducer (s) ↓	Inference (s) ↓
SparsePCA	Full-batch	*Computationally impractical at $m = 19{,}993$*				
SparsePCA	Mini-batch	75.14 ± 6.61	148.6	$134.54\times$	20.94	0.511
NMF	Full-batch	*Computationally impractical at $m = 19{,}993$*				
NMF	Mini-batch	75.77 ± 8.58	148.6	$134.54\times$	18.32	**0.450**
SCOPE (ours)	**Full**	75.07 ± 7.60	720.4 ± 10.8	$27.76\times$	**0.283**	0.481

MiniBatchSparsePCA and MiniBatchNMF use scalable approximations because their standard full-batch counterparts are impractical in this HDLSS setting. Despite comparable ROC-AUC, SCOPE reduces representation construction time by approximately $73.9\times$ relative to MiniBatchSparsePCA and $64.7\times$ relative to MiniBatchNMF.

Table L.5: **Feature-selection ablation on SMK with TabPFN-3.** ANOVA-k and Random-k retain exactly the fold-specific number of features selected by SCOPE, yielding the same mean output dimension and compression ratio. ANOVA-k uses training-fold labels through f_{classif}, whereas Random-k is label-free. SCOPE is also label-free but compresses the complete feature space into structured meta-features rather than discarding unselected features. Results are averaged over 5×5 repeated cross-validation. SCOPE is highlighted in blue.

Method	Labels	ROC-AUC ↑	M	Comp.	Reducer (s) ↓	Inference (s) ↓
ANOVA-k	Yes	**76.91**	720.4	$27.76\times$	0.0208	0.507
Random-k	No	75.16	720.4	$27.76\times$	**0.0003**	0.501
SCOPE (ours)	**No**	**75.07**	720.4	$27.76\times$	0.2833	**0.481**

ANOVA-k is supervised and uses labels from the training fold. Random-k and SCOPE are label-free. Feature selection baselines discard all unselected variables, whereas SCOPE aggregates information from the full feature space into M structured meta-features.

Table L.6: **Compressor configurations used in the benchmark.** Let k_{S} denote the fold-specific dimension selected by SCOPE, n_{tr} the training-fold sample size, and m the number of input features. User-specified hyperparameters were fixed across datasets; only the stated data-dependent dimension rules varied by fold. Row colors distinguish SCOPE, linear/projection, learned, manifold, supervised, and prior order-aware compressors. † denotes supervised dimensionality reduction.

Compressor	Output dimension	Fixed configuration
SCOPE (ours)	**Self-calibrated M^***	LENS metric = Euclidean; bins = 32; $b_{\text{TSP}} = 30$; $b_{\text{MinLA}} = 4$; $\eta = 1$; $R = 20$; standardization and segment centering enabled; no backbone dimension limit.
PCA	$\min(k_{\text{S}}, n_{\text{tr}} - 1, m)$	Full SVD solver.
GRP	k_{S}	Gaussian random projection; random state = 42.
AE	k_{S}	Architecture $m \to h \to k_{\text{S}} \to h \to m$; $h = \max(768, k_{\text{S}} + 64)$; ReLU; MSE loss; Adam; learning rate 10^{-3}; 100 full-batch epochs; seed = 42.
UMAP	$\min(k_{\text{S}}, 100)$	$n_{\text{neighbors}} = 15$ when feasible; minimum distance = 0.1; Euclidean metric; spectral initialization; random state = 42; transform seed = 42.
PaCMAP	$\min(k_{\text{S}}, 100)$	$n_{\text{neighbors}} = 10$; MN ratio = 0.5; FP ratio = 2.0; PCA preprocessing disabled; random initialization; random state = 42.
LocalMAP	$\min(k_{\text{S}}, 100, n_{\text{tr}} - 1, m)$	$n_{\text{neighbors}} = 10$; MN ratio = 0.5; FP ratio = 2.0; PCA preprocessing disabled; random initialization; random state = 42.
ParamRepulsor	$\min(k_{\text{S}}, 100, n_{\text{tr}} - 1, m)$	$n_{\text{neighbors}} = 10$; $n_{\text{FP}} = 20$; $n_{\text{MN}} = 5$; Euclidean distance; PCA preprocessing disabled; scaling option = None; 450 epochs; seed = 42.
Kernel PCA	$\min(k_{\text{S}}, n_{\text{tr}} - 1)$	RBF kernel; γ = scikit-learn default; dense eigensolver; inverse transform disabled; zero-eigenvalue removal disabled; random state = 42.
PLS†	$\min(k_{\text{S}}, n_{\text{tr}} - 1, m)$	Scaling enabled; maximum iterations = 500; tolerance = 10^{-6}; training labels used only within each fold.
NSC-pSP	Internal NSC rule, $32 \le M \le 512$	GO-LR: KL-divergence metric, 7 clusters, 32 bins, one refinement pass, direction selection enabled, 3,000-feature computational cap. NSC-pSP: uniform segmentation, default M rule, $\tau = 0.99$, $\gamma = 2$, $\beta = 0.5$, $l_{\min} = 8$.

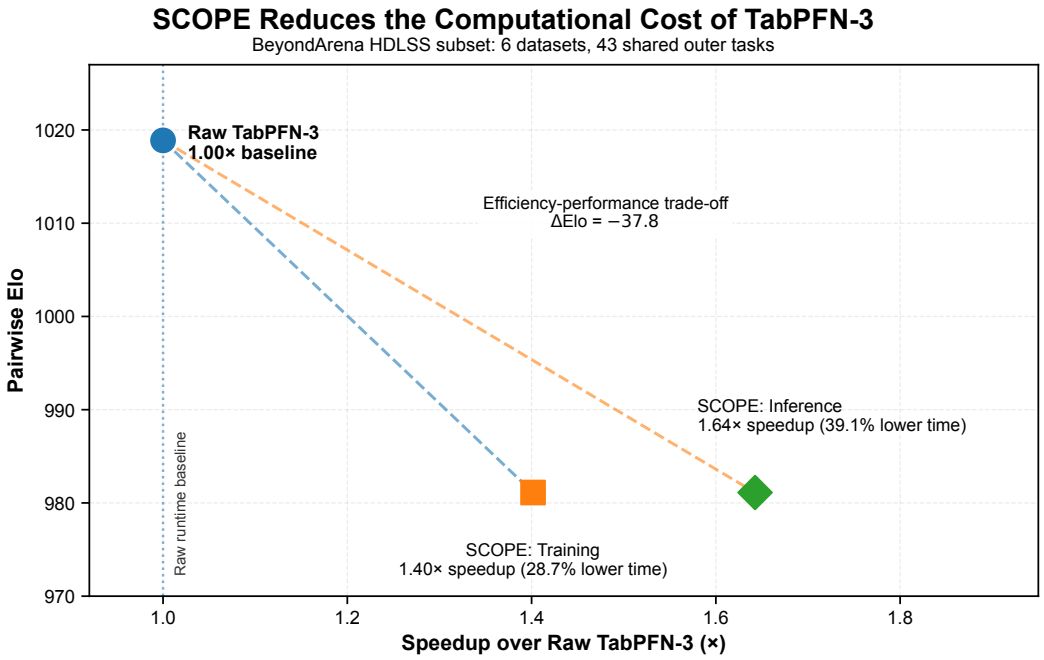


Figure L.2: **Efficiency-performance trade-off between Raw TabPFN-3 and SCOPE + TabPFN-3 on the BeyondArena HDLSS subset.** We retain only the six BeyondArena datasets satisfying $m > n$ and compare the two methods under the same 5×5 internal validation/bagging protocol. Pairwise Elo is computed over the resulting 43 shared outer tasks and excludes the other pre-computed BeyondArena methods from the comparison. SCOPE reduces median training time by 28.7% and median inference time by 39.1%, corresponding to $1.40\times$ and $1.64\times$ speedups, respectively, while incurring a 37.8-point decrease in pairwise Elo relative to Raw TabPFN-3.