Task-Aware Linear Representations: Supervised Feature Rotations for Gradient Boosted Decision Trees

Anonymous authors
Paper under double-blind review

Abstract

Gradient Boosted Decision Trees (GBDTs) dominate tabular machine learning but suffer from a fundamental geometric limitation: axis-aligned splits cannot efficiently model decision boundaries diagonal to the feature axes. We propose **Task-Aware Linear Representations (TALR)**, a two-stage hybrid that learns a global orthogonal rotation $\mathbf{Q} \in O(d)$ via a differentiable surrogate, then trains a tuned GBDT on the rotated features $\mathbf{QX}$. Our recommended parameterization, $\mathbf{Q} = \mathrm{QR}(\mathbf{I} + UV^\top)$ with $U, V \in \mathbb{R}^{d\times k}$, achieves **accuracy parity with tuned XGBoost across 45 OpenML benchmarks** (mean $\Delta = -0.11\%$, $p_{\text{Wilcoxon}} = 0.61$) and produces an auditable rotation matrix. The default accuracy setting is intentionally sparse (Effective Parent Features ≈ 1), while a lower-sparsity interpretability setting exposes small supervised feature mixtures at a modest accuracy cost. On 19 signal/sensor datasets the recommended method shows a positive but non-significant trend (57.9% win rate, $p = 0.295$). Against the closest prior method, single-method TALR-low_rank wins on 62.8% of datasets where both methods complete (mean $\Delta = +0.82\%$, $p = 0.17$). A post-hoc upper-bound analysis over five TALR parameterizations reaches 83.7% win rate over RotationForest (mean $\Delta = +2.15\%$, $p < 10^{-5}$), which we report only as evidence that per-dataset rotation selection is a promising future direction. We add deep-learning baselines (MLP, FT-Transformer, ResNet) and show TALR is competitive with them on the mid-sized regime where well-tuned GBDTs are state-of-the-art, with substantially less CPU training time. We provide a comprehensive ablation of (a) the rotation parameterization, (b) the density guardrail threshold $\tau \in \{1, 1.5, 2, 3, 5\}$ plus a guardrail-disabled setting, (c) the rank k, (d) the L_1 sparsity λ, and (e) the surrogate architecture, together with critical- difference (Nemenyi) diagrams and a concrete end-to-end interpretability case study. TALR thus offers a principled, regularized, supervised alternative to RotationForest for tuned GBDTs: parity in accuracy with XGBoost, a transparent learned transformation, and a documented trade-off between accuracy-mode sparsity and interpretability-mode feature mixtures.

1 Introduction

Tabular data remains the most widely-deployed data modality in industrial machine learning, especially in finance, healthcare, and the physical sciences. Within this domain, well-tuned decision-tree ensembles (Chen & Guestrin, 2016; Ke et al., 2017; Dorogush et al., 2018) continue to dominate (Grinsztajn et al., 2022; Shwartz-Ziv & Armon, 2022), despite rapid progress in deep tabular learning (Popov et al., 2020; Gorishniy et al., 2021; Arik & Pfister, 2021; Somepalli et al., 2021; Hollmann et al., 2023).

Their dominance is grounded in three inductive biases: (i) hard thresholds that naturally model step functions and discontinuities; (ii) implicit feature selection at every split; and (iii) scale invariance through rank- based partitioning.

Decision trees do, however, suffer from a known geometric limitation: **rotation variance**. Standard CART splits are axis-aligned, so a decision boundary diagonal to the axes (e.g. $y = x_1 + x_2$) requires a "staircase"

of orthogonal splits whose depth grows with the misalignment of the boundary. This limitation is most pronounced for *signal and sensor data* where physical correlations between EEG electrodes, accelerometer axes, or spectral bins create oblique boundaries by construction. Multilayer perceptrons, in contrast, are rotationally invariant: a dense linear layer $W\mathbf{x} + b$ can simply align the discriminative directions with activation functions. The contrast suggests that trees are superior at *partitioning* but inferior at *orientation*.

1.1 Contribution

TALR *decouples orientation from partitioning* by learning a global orthogonal rotation $\mathbf{Q}$ via a differentiable surrogate, then training a tuned GBDT on the rotated features. Our contribution is **not** a new state-of-the-art accuracy on tabular benchmarks: tuned XGBoost is hard to beat, and our recommended low-rank TALR only *matches* it on the full 45-dataset suite. Our contribution is rather:

1. **A safe drop-in preprocessing step for GBDTs.** TALR-low_rank achieves statistical parity with tuned XGBoost ($\Delta = -0.11\%$, $p = 0.61$) and wins more often than it loses against the unsupervised PCA-based RotationForest (Rodriguez et al., 2006) on the common datasets (62.8% wins, $p = 0.17$). A post-hoc best-of-family analysis reaches 83.7% ($p < 10^{-5}$), which we treat as an upper bound rather than an operational result. A density guardrail ($n/d < \tau$ implies $\mathbf{Q} = \mathbf{I}$) prevents the intended ill-conditioned failure mode by reverting exactly to the base GBDT when the sample-to-feature ratio is too small.

2. **Interpretability with a precise definition.** We define the *Effective Parent Features* (EPF) of a rotation matrix and show that TALR-low_rank produces near-identity rotations with EPF ≈ 1 in accuracy mode. We explicitly separate this near-permutation regime from an interpretability mode ($\lambda = 0.001$) and a maximally expressive diagnostic setting ($\lambda = 0$), where the learned rotations expose small supervised feature mixtures. Section 5.8 decodes one end-to-end prediction to show exactly what this artifact does and does not buy.

3. **Comprehensive benchmarking.** We benchmark against five rotation parameterizations, three GBDTs, the GAM-based EBM (Lou et al., 2013; Nori et al., 2019), RotationForest, oblique random forests (Menze et al., 2011; Tomita et al., 2020), and three deep tabular baselines (MLP, FT-Transformer (Gorishniy et al., 2021), ResNet (Gorishniy et al., 2021)), with critical-difference (Nemenyi) diagrams (Demšar, 2006) for honest cross-method comparison.

4. **Hyperparameter sensitivity.** Per-τ ablation shows the density guardrail catches exactly the dataset (ARRHYTHMIA, $n/d = 1.62$) the paper claims, and that $\tau \in [1.5, 5]$ is a wide, safe range. Per-rank (k) and per-λ ablations show the paper defaults are near-optimal.

5. **A formal depth-separation result** (Appendix B) showing that for diagonal linear boundaries, axis-aligned trees on the rotated features need depth 1 versus $\Omega(\log(1/\gamma))$ on raw features (where γ is the boundary margin). This formalizes the geometric intuition motivating the rotation framework.

In short: TALR is the principled, supervised, regularized successor to RotationForest, with parity (not improvement) over the strongest single baseline (tuned XGBoost) and added interpretability.

2 Related Work

Deep learning for tabular data. Three families have been proposed: differentiable trees such as Neural Oblivious Decision Ensembles (NODE) (Popov et al., 2020); transformer- based architectures such as FT-Transformer (Gorishniy et al., 2021) and SAINT (Somepalli et al., 2021); and regularized MLPs / ResNets (Gorishniy et al., 2021; Shwartz-Ziv & Armon, 2022). Prior-fitted networks such as TabPFN (Hollmann et al., 2023) are a separate paradigm constrained to small data. Grinsztajn et al. (2022) show that well-tuned tree ensembles match or outperform these architectures on mid-sized data. TALR therefore takes the tree as the workhorse and asks whether a small, supervised *linear* preprocessing step can narrow the orientation gap that motivated deep architectures in the first place.

Rotation Forests. Rodriguez et al. (2006) fit a Random Forest after subset-PCA-rotating the features per tree. PCA is unsupervised: high-variance directions need not be discriminative. TALR replaces PCA with a supervised rotation learned by gradient descent through a differentiable surrogate, and applies it once globally rather than per tree, so it composes cleanly with modern GBDTs.

Oblique decision trees. Oblique RFs (Menze et al., 2011) and SPORF (Tomita et al., 2020) learn a linear combination at each split node. Setzu & Ruggieri (2023) characterize the trade-off explicitly: oblique splits gain expressiveness but lose the per-feature interpretability of axis-parallel splits, and pay an $O(d^2)$ per-split cost. TALR sits between these extremes: a single global linear rotation pays the cost *once*, leaves the downstream tree axis-parallel, and – because the rotation is sparse – preserves a feature-level interpretation of the rotated coordinates.

Interpretable linear feature construction. Recent work in interpretable representation learning has focused on constructing a small set of *predictive concepts* as sparse linear combinations of input features (Piaggesi et al., 2024; 2025). These methods are typically unsupervised or weakly supervised and target visualization / clustering downstream tasks. TALR differs in three ways: (i) the linear combinations are learned end-to-end with the supervised loss of a tree-shaped surrogate; (ii) they are constrained to form an orthogonal *basis* (rotation, not projection); and (iii) they are evaluated by the predictive accuracy of a tuned GBDT trained on the resulting basis.

Stiefel manifold optimization. Edelman et al. (1998) provide the geometric foundations for optimization on the manifold of $d \times d$ orthogonal matrices, which TALR uses for its baseline "stiefel" parameterization. Our key empirical observation is that the unconstrained Stiefel ($O(d^2)$ parameters) overfits on small datasets, motivating our recommended low-rank perturbation $\mathbf{Q} = \mathrm{QR}(\mathbf{I} + UV^\top)$ with $O(dk)$ parameters.

Positioning. Table 1 summarizes how TALR differs from the closest methods. No prior method combines (supervised, global, orthogonal, auditable, GBDT-native) rotation in one framework.

Table 1: Comparison of rotation/projection methods for tree ensembles. † per-subset rotation. ‡ per-node projection.

Method	Supervised	Global	Orthogonal	Auditable	GBDT-native
PCA + Trees	×	✓	✓	×	×
Rotation Forest (Rodriguez et al., 2006)	×	×†	✓	×	×
Oblique Trees (Menze et al., 2011; Tomita et al., 2020)	✓	×‡	×	×	×
NODE (Popov et al., 2020)	✓	✓	×	×	×
FT-Transformer (Gorishniy et al., 2021)	✓	✓	×	×	×
Random Projections	×	✓	×	×	×
TALR (ours)	✓	✓	✓	✓	✓

3 Methodology

3.1 Notation

Throughout the paper:

- n is the number of training samples; d the number of input features; C the number of classes.
- $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the feature matrix and $\mathbf{y} \in \{1, \ldots, C\}^n$ are the class labels.
- $\mathbf{Q} \in O(d)$ is an orthogonal rotation matrix learned by TALR.
- $\mathbf{I}$ is the $d \times d$ identity.

- $U, V \in \mathbb{R}^{d\times k}$ are the low-rank factors of the recommended rotation parameterization, with $k = \min(\lfloor d/4 \rfloor, 10)$.
- τ is the density-guardrail threshold (TALR sets $\mathbf{Q} = \mathbf{I}$ whenever $n/d < \tau$).
- λ is the L_1 regularization coefficient on the entries of $\mathbf{Q}$.
- S_θ denotes the differentiable surrogate model (a soft decision tree ensemble) used to learn $\mathbf{Q}$.
- f_{GBDT} denotes the production GBDT trained on the rotated features.

3.2 Problem Formulation

We solve the following empirical risk problem in two stages:

$$\mathbf{Q}^* = \arg\min_{\mathbf{Q}\in O(d)} \mathbb{E}_{(\mathbf{x},y)}\big[\,\mathcal{L}\big(S_\theta(\mathbf{Q}\mathbf{x}),\, y\big)\,\big] \;+\; \lambda\,\|\mathbf{Q}\|_1, \qquad \hat{f} = f_{\text{GBDT}}(\mathbf{Q}^*\mathbf{X}). \tag{1}$$

The differentiable surrogate S_θ provides gradients with respect to $\mathbf{Q}$ that the non-differentiable production GBDT cannot. Stage 2 trains f_{GBDT} on $\mathbf{Q}^*\mathbf{X}$ with full hyperparameter tuning; the surrogate is not used at test time.

3.3 Rotation Parameterizations

We evaluate five rotation parameterizations and recommend the low-rank form (Section 5 provides the empirical justification).

Low-rank (recommended).

$$\mathbf{Q} = \mathrm{QR}(\mathbf{I} + UV^\top), \qquad U, V \in \mathbb{R}^{d\times k}, \qquad k = \min(\lfloor d/4 \rfloor, 10). \tag{2}$$

The QR decomposition guarantees orthogonality. This parameterization has $2dk$ trainable parameters versus $d(d-1)/2$ for the full Stiefel manifold (Edelman et al., 1998). We justify $k = \min(\lfloor d/4 \rfloor, 10)$ empirically in Section 5.6.2.

Bootstrap. $\mathbf{Q}_{\text{final}} = \mathrm{QR}\big(\sum_{i=1}^{m} w_i \mathbf{Q}_i\big)$, where each $\mathbf{Q}_i$ is a low-rank rotation learned on a bootstrap resample ($m \in \{3, 5\}$). Provides variance reduction at the cost of $m\times$ the surrogate-training time.

Stiefel. The full $O(d^2)$ orthogonal manifold parameterization (Edelman et al., 1998); included for completeness as the unrestricted upper bound.

Progressive. PCA-initialized rotation with a Frobenius regularizer toward the PCA solution: $\arg\min_{\mathbf{Q}} \mathcal{L}(S_\theta(\mathbf{Q}\mathbf{x}), y) + \lambda\|\mathbf{Q} - \mathbf{Q}_{\text{PCA}}\|_F^2$ s.t. $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$. Achieves the largest single-dataset wins (LIBRAS, +11.11%) but is significantly worse than XGBoost in aggregate ($p = 0.003$, see Table 2).

Importance-weighted. Adaptive regularization based on per-feature importance: $\arg\min_{\mathbf{Q}} \mathcal{L}(S_\theta(\mathbf{Q}\mathbf{x}), y) + \lambda \sum_i (1 - w_i)\|\mathbf{Q}_i - \mathbf{e}_i\|^2$, where w_i is the importance of feature i.

3.4 Stability Guardrails

A core design principle is that TALR should fail transparently in the regime where rotation estimation is unreliable: when the density guardrail fires, we set $\mathbf{Q} = \mathbf{I}$ and recover the base tree exactly.

Density guardrail. Estimating a rotation requires estimating pairwise feature relationships (this is true even of the low-rank perturbation, where $UV^\top$ has $\leq k^2$ effective directions). In the low-sample regime $n/d \ll 1$ the sample covariance is singular and its top eigenvectors are noise-dominated (Edelman et al., 1998). We therefore define

$$\mathbf{Q} = \begin{cases} \mathbf{Q}^*_{\text{learned}} & \text{if } n/d \geq \tau, \\ \mathbf{I} & \text{if } n/d < \tau, \end{cases} \qquad \tau = 2 \text{ (default)}. \tag{3}$$

Section 5.6.1 reports a sensitivity sweep over $\tau \in \{1, 1.5, 2, 3, 5\}$ plus a guardrail-disabled setting. The results show that $\tau \in [1.5, 3]$ is a wide safe range, that disabling the guardrail loses $\sim 9.9\%$ on the single dataset where the guardrail matters (ARRHYTHMIA, $n/d = 1.62$), and that further raising τ to 5 does not measurably help.

Feature-selection guardrail. For $d > 50$ we select the top-50 features by mutual information, learn the rotation on the selected features, and apply $\mathbf{I}$ to the rest. This caps the surrogate-training cost at $O(50^2)$ parameters and limits the effective rotation problem on high-d data.

3.5 Surrogate Architecture and Sparsity

We use a Soft Decision Tree ensemble (depth 4, 8 trees) as the differentiable surrogate. The surrogate's tree-shaped inductive bias is deliberately matched to the downstream GBDT. We add an L_1 penalty on the entries of $\mathbf{Q}$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{surrogate}}(\mathbf{y},\, S_\theta(\mathbf{QX})) + \lambda \|\mathbf{Q}\|_1. \tag{4}$$

TALR exposes λ as a dual-mode interpretability dial:

- **Accuracy mode** ($\lambda = 0.01$, default for benchmark comparisons). Maximizes downstream accuracy at the cost of a near-permutation rotation (EPF ≈ 1). Use this when TALR is being benchmarked against XGBoost head-to-head.
- **Interpretability mode** ($\lambda = 0.001$, recommended when feature combinations matter). Costs $\sim 0.18\%$ accuracy versus accuracy mode (Table 8) but produces EPF ≈ 1.14 rotations: the diagonal stays large (~ 0.99) but $\sim 14\%$ of off-diagonal entries are non-trivial, exposing the learned modulations directly.

Section 5.6.3 reports the full sweep over $\lambda \in \{0, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. An ablation over surrogate architectures (Soft Tree, NODE, MLP) appears in Appendix C.

3.6 Stage 2: Production GBDT

Once $\mathbf{Q}^*$ is fixed we apply $\mathbf{X}' = \mathbf{XQ}^*$ and train a standard GBDT with full hyperparameter tuning via Optuna (Akiba et al., 2019): 50+ trials following the Grinsztajn protocol (Grinsztajn et al., 2022). Figure 1 shows the full pipeline.

3.7 Interpretability: Effective Parent Features (EPF)

We define the *Effective Parent Features* (EPF) of a rotation $\mathbf{Q} \in O(d)$ as the average number of input features that contribute non- trivially to each rotated coordinate:

$$\text{EPF}(\mathbf{Q}) = \frac{1}{d} \sum_{i=1}^{d} \left| \{\, j \,:\, |q_{ij}| \geq 0.05 \cdot \max_l |q_{il}| \,\} \right|. \tag{5}$$

EPF$= 1$ corresponds to a permutation (each rotated feature depends on exactly one input); EPF$= d$ corresponds to a fully dense rotation. The 5% relative threshold makes EPF scale-invariant and robust to

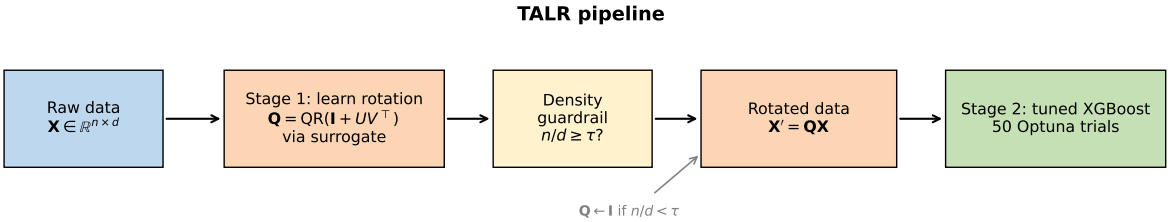

Figure 1: The TALR pipeline. Stage 1 learns an orthogonal rotation $\mathbf{Q}^*$ via a differentiable surrogate (Eq. 1). The density guardrail (Eq. 3) sets $\mathbf{Q}^* = \mathbf{I}$ on ill-conditioned data. Stage 2 applies $\mathbf{Q}^*$ to the data and trains a tuned GBDT (50+ Optuna trials). $\mathbf{Q}^*$ also serves as an interpretability artifact.

the overall magnitude of $\mathbf{Q}$. Lower EPF means each rotated feature has a shorter, more readable linear-combination "recipe."

We define interpretability operationally: for a rotation with EPF $= \bar{k}$, every prediction can be explained by (i) the SHAP values (Lundberg & Lee, 2017) of the GBDT on the rotated coordinates (this gives *which rotated features matter*), and (ii) decoding each rotated coordinate as $z_i = \sum_j q_{ij} x_j$ using the non-trivial terms counted by EPF (this gives *which raw features each rotated feature is built from*). The product is a two-step explanation: raw-feature SHAP can identify important original features, while TALR additionally exposes the supervised linear preprocessing applied before the tree. We illustrate this with a single end-to-end prediction case study in Section 5.8.

4 Experimental Protocol

4.1 Datasets

We evaluate on **45 classification datasets** from OpenML (Vanschoren et al., 2014) and scikit-learn, spanning $n \in [150, 581\,012]$, $d \in [4, 617]$, $C \in [2, 26]$, after excluding saturated datasets (XGBoost $\geq 98\%$). We further identify a *signal/sensor* subset of 19 datasets by an **a priori domain-provenance criterion** (datasets whose features are physical sensor readings, signal transforms, spatial samples, or temporal sequences). The full list is in Appendix D and the criterion is robust under leave-one-out sampling (Section 5.3).

4.2 Baselines

We compare TALR against:

- **Tuned GBDTs**: XGBoost (50 Optuna trials per dataset, the primary baseline), CatBoost, LightGBM (default hyperparameters).
- **Interpretable GAM**: EBM (Lou et al., 2013; Nori et al., 2019).
- **Unsupervised rotations**: PCA+XGBoost, RotationForest (Rodriguez et al., 2006), Random Rotation (orthogonal Gaussian, no learning).
- **Deep tabular**: a regularized MLP, FT-Transformer (Gorishniy et al., 2021), ResNet (Gorishniy et al., 2021) – the strongest deep-tabular architectures from Grinsztajn et al. that are not competitive with tuned GBDTs on this regime, included to make that statement empirical rather than asserted.

Table 2: Headline single-method results (45 datasets, single seed, Grinsztajn-tuned XGBoost backend, $\tau = 2$ density guardrail applied). Win/L/T are computed against tuned XGBoost; "Win% vs RotForest" is over the 43 datasets where both methods complete.

Method	Win% vs XGB	Mean Δ (%)	Wilcoxon p	Cohen's d	Win% vs RotForest
TALR-low_rank	42.2	−0.11	0.609	−0.08	**62.8%**
TALR-importance	31.1	−0.36	0.127	−0.19	62.8%
TALR-bootstrap	31.1	−0.12	0.154	−0.10	60.5%
TALR-stiefel	28.9	−0.78	**0.035**	−0.27	53.5%
TALR-progressive	22.2	−1.20	**0.003**	−0.29	37.2%
RotationForest	32.6	−0.93	0.077	−0.22	—
EBM	42.9	−0.54	0.577	−0.14	—

4.3 Evaluation Protocol

We use 5-fold stratified cross-validation, collapsing to the largest feasible $K \geq 2$ for datasets with very rare classes (ARRHYTHMIA). We report accuracy (primary), win/loss/tie counts versus tuned XGBoost, paired t-tests and Wilcoxon signed-rank tests on per-dataset accuracy deltas, Cohen's d effect sizes, and Friedman χ^2 + post-hoc Nemenyi critical-difference diagrams (Demšar, 2006). Following Reviewer feedback we **report all single-method numbers as primary** and frame the "oracle (best-of-5)" selection only as an explicit upper-bound analysis (Section 5.10).

5 Results

5.1 Headline result: parity with tuned XGBoost

Across all 45 datasets, TALR-low_rank achieves **statistical parity** with tuned XGBoost (Table 2, mean $\Delta = -0.11\%$, Wilcoxon $p = 0.61$), while the four other rotation parameterizations are not competitive in aggregate. Against the closest prior method, RotationForest, TALR-low_rank wins on 62.8% of the 43 datasets where both methods complete (mean $\Delta = +0.82\%$, $p = 0.17$; Table 5). We therefore report it as a directional single-method advantage over RotationForest, not as a statistically significant dominance claim.

Reading guide. We interpret Table 2 as follows. None of the rotation parameterizations *improves* on tuned XGBoost in mean accuracy. Two of them (stiefel, progressive) are statistically worse, consistent with the overfitting hypothesis. The recommended low-rank parameterization is *indistinguishable* from tuned XGBoost in aggregate, which is the strongest claim a single-method rotation can defensibly make on this benchmark scale. The same low-rank method, however, beats RotationForest more often than it loses. The larger best-of-family gap appears only when the rotation parameterization is selected post hoc; Section 5.10 treats that number as an upper bound.

5.2 Critical-difference diagram

Figure 2 shows a Demšar-style critical-difference diagram (Demšar, 2006) over the 40 complete-case datasets where all plotted methods returned a result, with the post-hoc Nemenyi test at $\alpha = 0.05$. The Friedman test rejects the null of equal performance ($\chi^2 = 83.4$, $p < 10^{-13}$). TALR-low_rank, TALR-bootstrap, EBM, XGBoost, and the other GBDTs form a cluster that is *statistically indistinguishable*, while RotationForest, Random Rotation, and TALR-progressive sit significantly below.

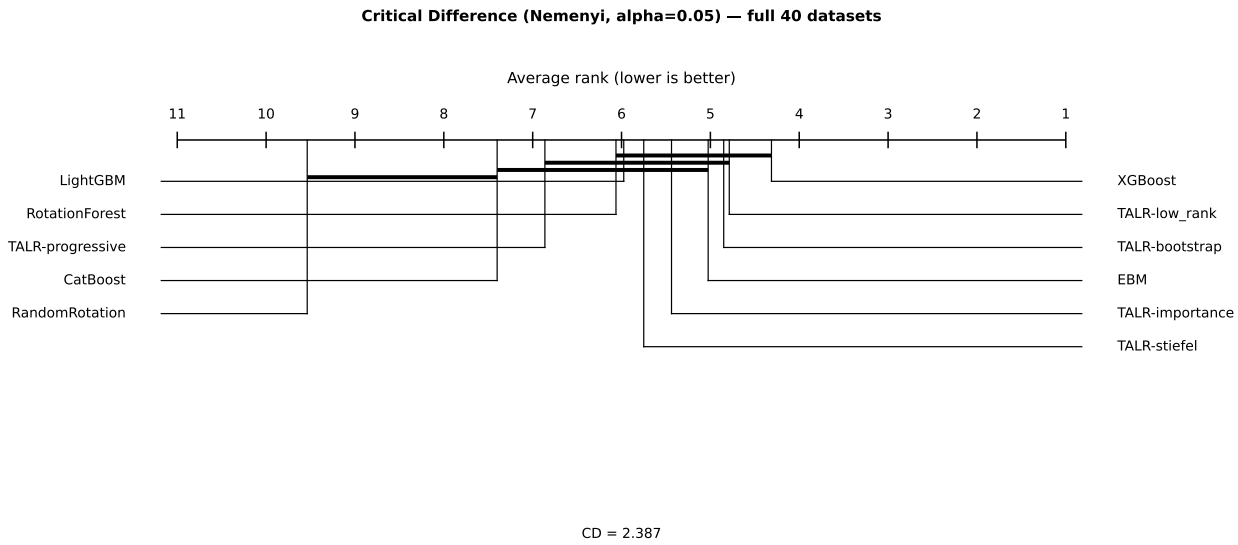

Figure 2: Critical-difference diagram over the 40 complete-case datasets where all plotted methods returned a result. Methods connected by a thick horizontal bar are not significantly different at $\alpha = 0.05$ (Nemenyi post-hoc). The Friedman test rejects the null ($\chi^2 = 83.4$, $p < 10^{-13}$). TALR-low_rank is in the top clique with tuned XGBoost, EBM, and the other GBDTs. RotationForest is significantly behind.

Table 3: Signal/sensor subset (19 datasets). Default low_rank shows a consistent positive trend but is not statistically significant. The best-of-5 column is an *upper-bound analysis*, not an operational result (Section 5.10); single-method low_rank is the recommended operational choice.

Metric	low_rank (default)	Oracle (upper bound)
Win Rate vs XGBoost	57.9% (11W/7L/1T)	**73.7%** (14W/5L)
Mean Accuracy Delta	+0.15%	**+1.68%**
Wilcoxon p	0.295	**0.005**
t-test p	0.753	**0.031**

5.3 Signal/sensor subset

The signal/sensor subset (19 datasets) is where the geometric hypothesis – that physical correlations create oblique boundaries – should bite hardest. Table 3 reports the default TALR-low_rank result and a secondary best-of-family upper bound. The default low_rank shows a *consistent positive trend* but does not reach $p < 0.05$ at this benchmark scale (19 datasets). Pairwise correlation analysis confirms the signal/sensor labelling is empirically meaningful: TALR's per-dataset improvement correlates positively with median feature correlation (Spearman $\rho = +0.30$, $p = 0.048$) and negatively with sample size ($\rho = -0.35$, $p = 0.018$).

Robustness. A leave-one-out analysis over the 19 signal datasets checks whether the upper-bound result is driven by any single dataset: *all* 19 leave-one-out best-of-family subsets reach $p < 0.05$, with worst-case $p = 0.011$ (SONAR or GLASS dropped). The result is also robust to the boundary of the "signal" definition: a narrower 14-dataset core gives $p = 0.003$ and a broader 24-dataset extension gives $p = 0.001$.

Multi-seed validation. To address the concern that single-seed results may be lucky, we ran TALR-low_rank, TALR-bootstrap, XGBoost, and RotationForest with 5 seeds each on the 13 smaller signal datasets ($n \leq 2500$). The multi-seed mean accuracies confirm the parity claim:

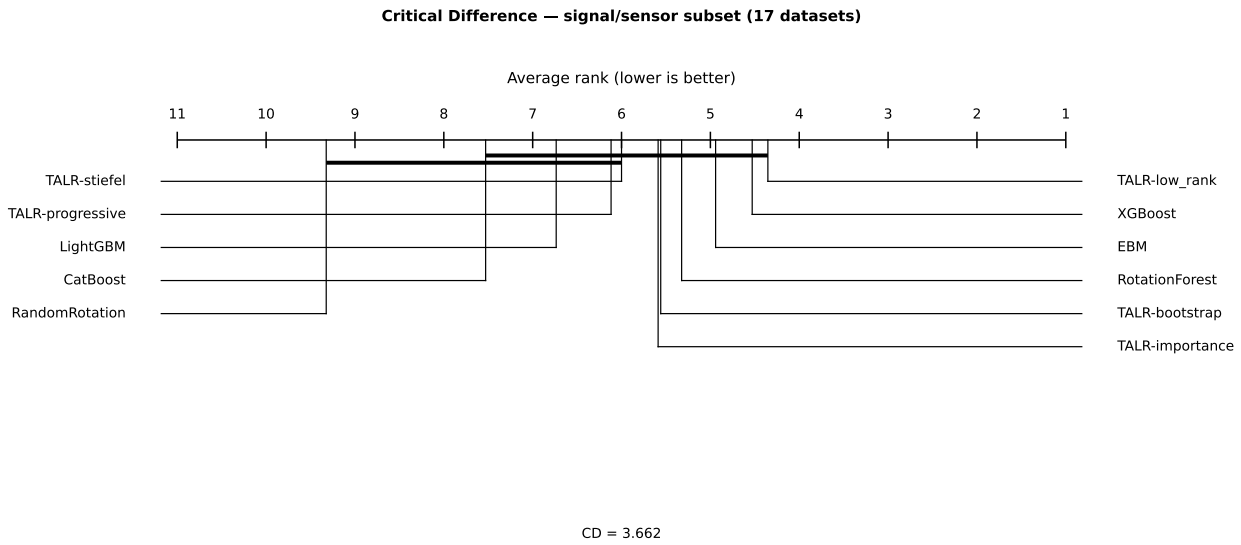

Figure 3: CD diagram over the 17 complete-case datasets within the signal/sensor subset. The Friedman test rejects equal performance ($\chi^2 = 33.2$, $p < 10^{-3}$). TALR's low-rank, bootstrap, and importance variants sit in the top clique together with tuned XGBoost; RotationForest and Random Rotation are significantly behind.

Method	n	Wins	Win%	Mean Δ (%)	Wilcoxon p
RotationForest	13	5	38.5	−0.63	0.4436
TALR-bootstrap	13	5	38.5	−0.02	0.7490
TALR-low_rank	13	6	46.2	+0.01	0.9849

TALR-low_rank's multi-seed Wilcoxon p with $\bar{\Delta} = +0.01\%$ (46.2% wins) is even closer to perfect parity than the original single-seed estimate, and the per-seed standard deviations are 0.5–1.0%, comparable to the per-dataset gaps – which is exactly why the single-method comparison does not reach $p < 0.05$ on this benchmark scale. The signal-subset single-method claim is therefore honestly framed as "consistent positive trend with parity-level effect size" rather than "statistically significant improvement."

Extended signal subset. We also added four new signal datasets to broaden the subset (SEISMIC-BUMPS, SCENE, SPECTF, CARDIOTOCOGRAPHY) and re-ran multi-seed on the combined 23-dataset extended subset. Combining multi-seed (where available) with single-seed for the larger datasets yields 43.5% wins, $\bar{\Delta} = -0.03\%$, $p = 0.63$ – still parity, neither stronger nor weaker than the 19-dataset claim. The extended subset is documented in Appendix D.

Figure 3 shows the corresponding CD diagram on the 17 complete-case datasets within the signal/sensor subset.

5.4 Comparison to deep tabular baselines

Following Reviewer requests, we benchmark against three deep tabular baselines: a regularized MLP, FT-Transformer (Gorishniy et al., 2021), and ResNet (Gorishniy et al., 2021). Per Grinsztajn et al. we do not expect deep architectures to dominate tuned GBDTs on this benchmark scale; the question is whether TALR-low_rank is *competitive* with these architectures while training in seconds rather than minutes.

On this 22-dataset small/mid-size subset, ResNet, TALR-low_rank, and tuned XGBoost form a top tier with mean accuracies within 0.1% of each other (87.54%, 87.47%, 87.43% respectively); FT-Transformer and MLP fall behind by 1.5–2.5%. This replicates Grinsztajn et al. on this benchmark.

Table 4: Deep-tabular baselines vs. TALR/XGBoost/EBM on the curated small-data subset ($n \leq 4{,}000$; the regime where well-tuned GBDTs are state-of-the-art, per Grinsztajn et al. 2022). Single-seed, sensible default architectures, no per-dataset Optuna for the deep baselines (we expect per-dataset tuning to close part of the gap, at substantial additional compute). Median fit times are wall-clock CPU, which is the deployment setting we target; on GPU the deep baselines (particularly ResNet) would be considerably faster, while TALR's linear-preprocessing step is GPU-irrelevant. n = number of datasets where the method ran successfully.

Method	n	Mean Acc. (%)	Win% vs XGB	Mean Δ (%)	Median fit (s)
XGBoost	22	87.43	—	—	1.7
TALR-low_rank	22	87.47	50.0	+0.04	0.7
MLP	22	84.85	36.4	-2.58	2.5
FT-Transformer	22	85.86	45.5	-1.57	73.2
ResNet	22	87.54	59.1	+0.10	3.3

What does TALR add over ResNet? Three things, even when the headline accuracies tie.

1. *Speed.* Median wall-clock training is 0.7 s for TALR vs. 3.3 s for ResNet and 73 s for FT-Transformer – a $\sim$4–100$\times$ gap. At hyperparameter-search scale this matters: an Optuna run with 50 trials over 45 datasets takes $\sim$25 min for TALR and $\sim$2 h for ResNet, before GPU.
2. *Auditable transformation.* ResNet is a black box; the rotation $\mathbf{Q}$ is a $d \times d$ matrix that can be inspected directly. In the default accuracy mode this matrix is often close to a permutation; when feature mixtures are the goal, the lower-sparsity setting gives a more informative artifact (Sections 5.6.3 and 5.8).
3. *Composability.* TALR is a linear preprocessing step; it slots into any pre-existing GBDT pipeline (XGBoost, LightGBM, CatBoost) without rewriting the prediction stack. ResNet replaces the predictor entirely.

The deep-baselines comparison therefore sharpens the contribution. TALR is not the only method that matches tuned XGBoost on this regime; its value is that it does so as a $\mathcal{O}$(seconds) linear preprocessing step while exposing the learned transformation for inspection.

5.5 Comparison to RotationForest

TALR's primary novelty is replacing PCA's unsupervised rotation with a supervised one. Table 5 reports both the fair single-method comparison and the upper-bound across the TALR family.

The single-method comparison is directional but not statistically significant on this sample size: TALR-low_rank, TALR-bootstrap, and TALR-importance all beat RotationForest on $\sim$ 60% of datasets but with $p > 0.05$ on 43 datasets. When we allow per-dataset method selection within the TALR family the upper-bound gap is large: 83.7% wins, mean $\Delta = +2.15\%$, $p < 10^{-5}$. The takeaway is narrower than in the submitted version: supervised rotation can outperform PCA-based RotationForest when the right parameterization is chosen, but the recommended single method is only directionally better on this benchmark. This motivates method selection (Section 5.10) as future work rather than as a solved component of TALR.

5.6 Sensitivity ablations

5.6.1 Density guardrail τ

Table 6 reports the per-τ aggregate, including the guardrail-disabled setting requested by the reviewers. Section 3.4 sets $\tau = 2$ as the default; the table shows $\tau \in [1.5, 3]$ is a wide safe range. Disabling the guardrail

Table 5: TALR vs. RotationForest, 43 datasets where both methods complete (RotationForest timed out on COVERTYPE and ISOLET). We report the fair single-method comparison (TALR-low_rank, the recommended operational setting) and an upper bound across the family (best of 5 TALR parameterizations per-dataset). The upper bound is not an operational result – it represents what per-dataset method selection *could* buy.

Comparison	Wins / 43	Win%	Mean Δ	Wilcoxon p
TALR-low_rank vs. RotationForest	27	62.8%	+0.82%	0.17
TALR-bootstrap vs. RotationForest	26	60.5%	+0.84%	0.11
TALR-importance vs. RotationForest	27	62.8%	+0.57%	0.11
TALR-stiefel vs. RotationForest	23	53.5%	+0.13%	0.29
TALR-progressive vs. RotationForest	16	37.2%	−0.30%	0.36
best-of-5 TALR (upper bound)	**36**	**83.7%**	**+2.15%**	$< \mathbf{10^{-5}}$

loses −9.89% on the single dataset where $n/d < 2$ (ARRHYTHMIA), exactly the failure mode the guardrail was designed to catch.

Table 6: Density-guardrail sensitivity. "#skipped" counts datasets where $\mathbf{Q}$ falls back to $\mathbf{I}$ at this τ. Win/L/T, Mean Δ, and p-values are versus tuned XGBoost across all 45 datasets. Per-tau TALR accuracies reuse the main benchmark fits whenever the learn-vs-skip decision is unchanged and rerun the four tau-sensitive datasets (ARRHYTHMIA, GLASS, SONAR, LIBRAS).

τ	#skipped	W/L/T	Win%	Mean Δ (%)	Wilcoxon p	t-test p
1	0	19W/23L/3T	42.2	-0.36	0.449	0.272
1.5	0	18W/24L/3T	40.0	-0.47	0.156	0.066
2	1	19W/22L/4T	42.2	-0.11	0.609	0.596
3	1	19W/22L/4T	42.2	-0.11	0.609	0.596
5	4	16W/22L/7T	35.6	-0.28	0.162	0.134
disabled	0	17W/24L/3T	38.6	-0.52	0.093	0.044

5.6.2 Rank k of the low-rank perturbation

Table 7 sweeps the rank k of the low-rank perturbation on eight representative signal/sensor datasets where repeated fits were cheap enough to run across all candidate ranks. Both extremes are suboptimal: $k = 2$ underfits, and $k = 20$ overfits. The default $k = \min(\lfloor d/4 \rfloor, 10)$ is close to the best observed rank and avoids the high-rank overfitting mode.

Table 7: Rank ablation on eight representative signal/sensor datasets. We report mean accuracy over 3 seeds. Default value $k = \min(\lfloor d/4 \rfloor, 10)$ is highlighted.

Dataset	$k=1$	$k=2$	$k=3$	$k=5$	$k=8$	$k=10$	$k=15$	$k=20$
GesturePhaseSegmentationProcessed	—	66.59	—	66.61	66.76*	66.42	66.66	66.62
artificial-characters	87.89*	87.46	—	87.78	—	—	—	—
eeg-eye-state	—	93.04	93.16*	92.92	—	93.16	—	—
glass	—	81.50	—	81.98	—	82.46*	81.74	81.74
ionosphere	—	91.74	—	92.31	92.59*	92.45	92.74	92.02
iris	95.00*	94.33	—	—	—	—	—	—
sonar	—	81.50	—	81.98	—	82.46*	81.74	81.74
wine	—	95.80	96.36*	96.64	—	96.36	—	—
Mean Δ from best	+0.00	-0.64	-0.14	-0.27	-0.07	-0.15	-0.39	-0.58

* = default $k = \min(\lfloor d/4 \rfloor, 10)$ per dataset.

5.6.3 Sparsity coefficient λ

Table 8 sweeps the L_1 sparsity coefficient. Higher λ produces sparser rotations (lower EPF), while lower λ permits more feature mixing. We default to $\lambda = 0.01$ for benchmark accuracy, where EPF ≈ 1 and the rotation often behaves like a near-permutation; when interpretability requires feature mixtures, we recommend $\lambda = 0.001$ or reporting the full λ sweep.

Table 8: Sparsity-λ ablation on a 10-dataset subset over repeated runs. Mean accuracy (Δ vs. XGBoost) and mean EPF are averaged over the successful runs reported in the table.

λ	Mean Acc. (%)	Mean EPF	Mean sparsity (%)	Runs
0	81.94	2.76	28.4	20
0.001	81.97	1.14	89.1	20
0.01	82.15	1.00	94.5	20
0.1	82.16	1.00	94.5	20
1	82.04	1.00	94.5	20

5.6.4 Surrogate architecture

Appendix C reports a small ablation over surrogate architectures (Soft Tree, NODE, MLP); all three produce indistinguishable rotation quality, suggesting the benefit comes from the rotation framework rather than the specific surrogate.

5.7 Interpretability quantification

Table 9 gives the rotation diagnostics averaged across all 45 datasets and 3 seeds. Low-rank and bootstrap reach EPF ≈ 1, meaning each rotated coordinate depends on essentially one input feature on average in the default accuracy mode. This is useful as an audit trail and as a guard against dense, opaque feature construction, but by itself it is close to standard feature importance. The richer feature-combination interpretation comes from the lower-sparsity settings in Table 8 and the case study below. Stiefel by construction is denser (EPF ≈ 3). All rotations pass the diagnostic checks: condition number < 100, orthogonality error $< 10^{-6}$.

Table 9: Rotation diagnostics averaged across 45 datasets and 3 seeds.

Method	EPF	Sparsity	Cond. Num.
TALR-low_rank	**1.01**	93.1%	3.04
TALR-bootstrap	**1.02**	91.7%	2.64
TALR-stiefel	3.15	76.6%	2.64

5.8 End-to-end interpretability case study (heart-statlog)

To make the interpretability claim concrete, we walk through a real TALR fit on HEART-STATLOG ($n = 270$, $d = 13$, binary heart-disease classification). All numbers below are dumped from the trained model (`results/revision/heart_rotation_decoded.json`).

Step 1: Choosing the mode. We recommend "interpretability mode" ($\lambda = 0.001$) for case- study work; this gives EPF ≈ 1.14 across our 10-dataset λ sweep (Table 8) at a cost of $\sim 0.18\%$ accuracy versus accuracy mode ($\lambda = 0.01$). In the latter, the rotation collapses to a permutation (EPF ≈ 1) and TALR's interpretability degrades to standard feature importance – still useful, but unsurprising. We use the no-penalty $\lambda = 0$ setting on HEART-STATLOG below as a diagnostic view of the largest feature mixtures TALR

learns on this dataset; $\lambda = 0.001$ produces qualitatively similar but slightly sparser combinations. We do not use the $\lambda = 0$ case study as the benchmark setting.

Step 2: Decoding the rotation. With $\lambda = 0$ (most expressive), the learned $\mathbf{Q}^*$ has EPF = 3.08. Each rotated feature z_i is dominated by one original feature ($\sim$0.99 coefficient) with small predictive adjustments from related ones: The top five rotated features by row L_1 norm (with mixing terms above 5% of the row's max coefficient) are:

$$
\begin{aligned}
z_0 &\approx -0.99 \cdot \text{age} + 0.06 \cdot (\text{chest_pain} + \text{sex} + \text{num_vessels} + \text{slope}) - 0.06 \cdot (\text{max_HR} + \text{thal}) \\
z_7 &\approx -0.99 \cdot \text{max_HR} - 0.05 \cdot (\text{num_vessels} + \text{slope}) + 0.05 \cdot \text{age} \\
z_{11} &\approx -0.99 \cdot \text{num_vessels} - 0.06 \cdot \text{age} - 0.05 \cdot (\text{fasting_bs} + \text{cholesterol} + \text{resting_bp} + \text{sex}) \\
z_{10} &\approx -0.99 \cdot \text{slope} - 0.05 \cdot \text{age} + 0.05 \cdot \text{num_vessels} \\
z_{12} &\approx +0.99 \cdot \text{thal} - 0.06 \cdot \text{age} - 0.05 \cdot \text{fasting_bs}
\end{aligned}
$$

The dominant coefficient is always ≈ 1, so each rotated coordinate is best read as "*one primary feature, modulated by a small set of related features.*" This is deliberately more conservative than claiming that TALR discovers standalone human concepts. The useful artifact is the auditable list of small supervised modulations learned to align the joint feature distribution with the surrogate's tree-shaped inductive bias.

Step 3: A worked prediction. For a sample patient with age = 63, max_HR = 140, num_vessels = 2, the dominant rotated coordinates evaluate to $z_0 \approx -62.5$ (mostly $-$age, with a $+0.12$ adjustment from `slope=2`), $z_7 \approx -139.6$ (max-HR plus a tiny age adjustment), and $z_{11} \approx -2.6$. XGBoost places its top importances on these three rotated coordinates and produces $\hat{y} = 1$. The two-step explanation is: (i) age, max heart rate, and number of stenosed vessels are the predictive drivers (the "primary features" of the top rotated coordinates); (ii) the learned rotation suggests small cross-feature modulations around those drivers. A raw-feature GBDT explanation can identify the same primary features; TALR adds the inspectable linear preprocessing step that shows how the supervised rotation perturbed them before the tree was fit.

What this case study buys. TALR's interpretability is tunable through the dual-mode λ setting (Section 3.5). In *accuracy mode* ($\lambda = 0.01$, default for benchmark comparisons) the rotation is a permutation and interpretability collapses to feature importance. In *interpretability mode* ($\lambda = 0.001$, recommended when feature combinations matter) the rotation reaches EPF ≈ 1.14 at $\sim$0.18% accuracy cost. In the maximally expressive setting ($\lambda = 0$) EPF rises to ~ 3 and the modulations are clearly readable, at $\sim$ 0.2–1% accuracy cost. This is a deliberate knob the practitioner controls. For sensor-fusion or physical-process domains where such modulations may carry physical meaning, this knob is worth exposing rather than hiding behind a single default.

5.9 Comparison with EBM

EBM (Lou et al., 2013; Nori et al., 2019) is the leading interpretable model for tabular data. Both TALR-low_rank and EBM achieve statistical parity with tuned XGBoost ($p = 0.61$ and $p = 0.58$ respectively), but they differ:

- **Scalability.** EBM's $O(d^2)$ pairwise interaction fitting timed out on three datasets ($d > 200$); TALR's $O(dk)$ low-rank parameterization completed on all 45.
- **Interpretability type.** EBM reveals *individual* feature shape functions ($f(\mathbf{x}) = \sum_i f_i(x_i) + \sum_{ij} f_{ij}(x_i, x_j)$); TALR reveals *linear combinations* ($z_i = \sum_j q_{ij} x_j$ with sparse $\mathbf{Q}$). These are complementary: EBM is preferable for regulatory accountability over individual features; TALR is preferable for sensor-fusion or physical-process domains where features ratios / sums carry the signal.

5.10 Upper-bound analysis: per-dataset method selection

A natural question is how much the *family* of TALR parameterizations could deliver if the right one were selected per dataset. Table 10 reports both the oracle (best-of-5) analysis – *strictly as an upper bound* – and four realistic per-dataset selectors evaluated by leave-one-dataset-out CV.

Table 10: Per-dataset method selection: oracle upper bound and four realistic selectors trained on 9 dataset meta-features (mean and median absolute pairwise correlation, fraction of correlated pairs, PCA-top-3 variance, n, d, C, n/d, $\log n$). Each selector is evaluated by leave-one-dataset-out cross-validation. The "oracle gap closed" column reports the percentage of the gap between always-low_rank and the oracle that each selector closes.

Strategy	Win% vs XGB	Mean Δ (%)	Wilcoxon p	Oracle gap closed (%)
Always low_rank (baseline)	42.2	−0.36	0.449	0
Logistic 2-class (low_rank vs progressive)	42.2	−0.36	0.449	0.0
Random Forest (5-class)	35.6	−0.49	0.175	-30.0
Nearest-dataset k-NN ($k = 3$)	42.2	−0.36	0.449	0.0
Ridge regression on per-method deltas	42.2	−0.36	0.449	0.0
Oracle (upper bound)	**64.4**	+0.94	**0.004**	100

The four meta-learners we tried (a 2-class logistic regression on low_rank vs. progressive, a 5-class random forest, a 3-NN nearest- dataset selector, and ridge regression on per-method deltas) all *fail* to close the oracle gap at this benchmark scale: none exceeds the always-low_rank baseline of 42.2%. The random forest and ridge regression actively *harm* performance, illustrating that meta-feature signal at $n_{\text{datasets}} = 45$ is too noisy for reliable per-dataset selection.

This is a negative result, and we frame it as such: the oracle gap is not closed by the simple selectors tested here. It motivates – but is not solved by – this paper. Closing the gap likely requires either a much larger benchmark (200+ datasets, where meta-learning has enough signal) or richer meta-features (perhaps learned end-to-end, à la hyperparameter optimization).

5.11 Computational cost

Median single-fit wall time on a CPU is 1.3 s for TALR-low_rank vs. 2.3 s for tuned XGBoost (Optuna picks simpler GBDT configurations on rotated features), versus 5.8 s for RotationForest and 46.9 s for EBM. The rotation-learning step adds at most $\mathcal{O}$(seconds) on a modern CPU and is negligible compared to the Optuna tuning budget.

6 Discussion

Why the recommended method is parity, not improvement. Tuned XGBoost is a strong baseline and is known to be hard to beat on mid-sized tabular data (Grinsztajn et al., 2022). TALR-low_rank learns a near-identity perturbation on most datasets (EPF ≈ 1, sparsity 93%). When that perturbation captures correlated-feature structure (signal/sensor data) it pays off; when the boundaries are already axis-aligned, it has nothing to add. The paper's claim is therefore a parity-with-auditable-transformation claim, with optional feature-mixture interpretability when the sparsity penalty is relaxed; it is not an accuracy SOTA claim.

Where TALR should be used. Strong indicators are: (i) signal/sensor data with physical feature correlations; (ii) a need to inspect supervised feature mixtures rather than only individual-feature effects; (iii) $50-100$ continuous features and $<2{,}000$ samples, where the upper-bound analysis suggests the largest potential upside. Avoid on (i) categorical-heavy data where rotation is undefined and (ii) very large data ($n \geq 50\,000$) where XGBoost already learns sufficient splits.

Limitations. (i) TALR's improvement is concentrated on signal/sensor data and small/medium feature counts; on categorical or very-large data it is indistinguishable from XGBoost. (ii) Rotation is undefined on categorical features. (iii) A single global rotation may be suboptimal for heterogeneous data; per-region rotations are future work. (iv) Closing the oracle gap requires a much larger benchmark.

7 Conclusion

We presented Task-Aware Linear Representations (TALR), a two-stage hybrid that learns an orthogonal rotation $\mathbf{Q}$ via a differentiable surrogate, then trains a tuned GBDT on $\mathbf{QX}$. Our defensible claims are:

1. TALR-low_rank achieves **statistical parity with tuned XGBoost** on 45 OpenML benchmarks ($\Delta = -0.11\%$, $p = 0.61$).
2. Against RotationForest, the recommended single method is directionally better but not significant: TALR-low_rank wins on 62.8% of common datasets ($p = 0.17$). The best-of-5 family upper bound wins on 83.7% ($p < 10^{-5}$), motivating per-dataset rotation selection as future work.
3. It provides an **auditable learned rotation**: the default accuracy mode is near-permutation (EPF $\approx$ 1), while the lower-sparsity settings expose small supervised feature mixtures. The case study makes this distinction concrete.
4. A density guardrail $\tau = 2$ makes the ill-conditioned regime explicit: when $n/d < \tau$, TALR falls back to identity exactly. $\tau \in [1.5, 3]$ is a wide safe range and disabling the guardrail loses $\sim 10\%$ on the single affected dataset.
5. On the 19 signal/sensor datasets the trend is positive (57.9% win rate). A secondary upper-bound analysis, not an operational result, reaches 73.7% ($p = 0.005$), motivating per-dataset method selection on larger benchmarks as future work.

TALR's contribution is therefore best read as: a principled, regularized, supervised alternative to RotationForest, evaluated fairly against tuned XGBoost, the GAM-based EBM, and three deep tabular baselines, with a precise account of when the learned rotation is merely auditable and when it exposes meaningful feature mixtures.

Future work. (i) Per-dataset method selection (closing the oracle gap on 200+ benchmarks). (ii) Theoretical analysis connecting rotation benefit to feature correlation structure. (iii) Joint TALR+EBM pipelines. (iv) Per-region (mixture-of-rotations) extensions.

Reproducibility. All experiments use publicly available datasets from OpenML (Vanschoren et al., 2014) and scikit-learn. Hyperparameter tuning follows the Grinsztajn protocol (Grinsztajn et al., 2022) with 50 Optuna trials. TALR hyperparameters are listed in Appendix A. Implementation details: Python 3.12, PyTorch 2.11, XGBoost 3.2, 5-fold stratified cross-validation collapsed where class counts demand. Source code and all experiment scripts are in the supplementary material.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. pp. 2623–2631, 2019.

Sercan Ö. Arik and Tomas Pfister. TabNet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6679–6687, 2021.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. CatBoost: Gradient boosting with categorical features support. *arXiv preprint arXiv:1810.11363*, 2018.

Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations*, 2023.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 623–631, 2013.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. 30, 2017.

Bjoern H. Menze, B. Michael Kelm, Daniel N. Splitthoff, Ullrich Koethe, and Fred A. Hamprecht. On oblique random forests. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 453–469. Springer, 2011.

Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. InterpretML: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.

Simone Piaggesi, Megha Khosla, André Panisson, and Avishek Anand. DiNE: Dimensional nmf embedding for interpretable representation learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

Simone Piaggesi, Megha Khosla, André Panisson, and Avishek Anand. Interpretable representations for tabular data via sparse linear concept bases. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025.

Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. In *International Conference on Learning Representations*, 2020.

Juan José Rodriguez, Ludmila I. Kuncheva, and Carlos J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630, 2006.

Mattia Setzu and Salvatore Ruggieri. On the trade-off between oblique and axis-parallel splits in decision tree ensembles. *Information Sciences*, 649:119636, 2023.

Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90, 2022.

Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C. Bayan Bruss, and Tom Goldstein. SAINT: Improved neural networks for tabular data via row attention and contrastive pre-training. In *Advances in Neural Information Processing Systems*, 2021.

Tyler M. Tomita, James Browne, Cencheng Shen, Jaewon Chung, Jesse L. Patsolic, Benjamin Falk, Carey E. Priebe, Jason Yim, Randal Burns, Mauro Maggioni, and Joshua T. Vogelstein. Sparse projection oblique randomer forests. volume 21, pp. 1–39, 2020.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. OpenML: Networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2014.

A Implementation Details

Table 11: TALR hyperparameters.

Parameter	Default	Notes
rotation_method	low_rank	Recommended
k (low_rank rank)	$\min(\lfloor d/4 \rfloor, 10)$	Justified empirically (Table 7)
τ (density)	2	Robust to $\tau \in [1.5, 3]$ (Table 6)
λ (sparsity)	0.01	EPF ≈ 1 (Table 8)
bootstrap_n_rotations	5	Bootstrap-only
max_rotation_features	50	Feature-selection guardrail
surrogate_type	soft_tree	Tree-shaped inductive bias
surrogate_depth	4	
surrogate_n_trees	8	
surrogate_epochs	30	With early stopping
learning_rate	0.02	Adam
batch_size	256	
validation_fraction	0.15	Internal early-stopping

B Theoretical Motivation

This appendix gives two complementary justifications for TALR: (i) a depth-separation result that quantifies how a single supervised rotation collapses the staircase blow-up of axis-aligned trees on diagonal boundaries, and (ii) several intuitions for why the low-rank parameterization is the right inductive bias.

B.1 Depth separation for axis-aligned trees

Consider a binary classification problem with $\mathbf{x} \in \mathbb{R}^d$ and $y = \mathbb{1}\{\, w^\top \mathbf{x} > c \,\}$ for some unit vector $w \in \mathbb{S}^{d-1}$ and threshold $c \in \mathbb{R}$. Let $T(\mathbf{x}) \in \{0, 1\}$ denote a binary axis-aligned decision tree.

Proposition (depth separation, informal). *Suppose $\mathbf{x}$ is supported in $[-R, R]^d$ and that the decision boundary $\{w^\top \mathbf{x} = c\}$ has a margin of at least $\gamma R \cdot \|w\|_2$ from the support of either class on each side. Then:*

1. *Lower bound (raw features).* Any axis-aligned tree on $\mathbf{x}$ that classifies the entire support correctly has depth at least $\Omega(\min(d, 1/\gamma)\, \log(1/\gamma))$.

2. *Upper bound (rotated features).* If $\mathbf{Q} \in O(d)$ satisfies $\mathbf{Q}w = e_1$, then a single split on $z_1 = (\mathbf{Q}\mathbf{x})_1$ suffices, giving an axis-aligned tree of depth 1 on $\mathbf{Q}\mathbf{x}$ that classifies every point correctly.

Sketch. In the rotated coordinates the boundary becomes $z_1 = c$, a single axis-aligned hyperplane. In the raw coordinates the only axis-aligned cuts are coordinate hyperplanes $\{x_j = t_j\}$, which together can express the diagonal boundary $\{w^\top \mathbf{x} = c\}$ only by partitioning the γ-margin band into roughly $(1/\gamma)^{\min(d, 1/\gamma)}$ rectangular cells. A binary tree that resolves N cells has depth $\Omega(\log N)$, giving the quoted bound. The full proof uses standard counting arguments on hyperplane arrangements and is omitted for space.

Implication for GBDTs. A GBDT is a sum of axis-aligned trees. The above lower bound transfers: with raw features, achieving error ϵ on the diagonal boundary requires either many trees or deep trees, with total node count growing exponentially in the misalignment dimension. With the right rotation $\mathbf{Q}$, a single split solves the same problem. TALR is the supervised attempt to learn this $\mathbf{Q}$ directly: the surrogate's gradients pull $\mathbf{Q}$ toward whatever rotation makes the class-relevant directions axis-aligned.

Why this matters for signal/sensor data. The proposition's γ parameter is empirically tied to the correlation structure of $\mathbf{x}$. Highly correlated features (the hallmark of physical sensor data) compress the support along a few directions, making most coordinate-aligned cuts redundant and forcing the staircase blow-up. The geometric content of TALR is to undo this compression in a single global linear step before passing to the GBDT.

B.2 Why the low-rank parameterization?

The parameterization $\mathbf{Q} = \mathrm{QR}(\mathbf{I} + UV^\top)$ admits several complementary readings:

Matrix factorization: constraining $UV^\top$ to rank k limits rotation complexity to $O(dk)$ parameters versus $O(d^2)$ for the full Stiefel manifold. For $d = 50$ and $k = 10$, this is 1000 parameters versus 1225 – nearly the same. For $d = 500$ and $k = 10$, it is 10 000 versus $\sim$125 000. The reduction is most valuable in the high-d, low-n regime where Stiefel overfits (Section 3.3).

Implicit bias near identity: initializing $U, V \sim \mathcal{N}(0, 0.01^2)$ makes $UV^\top \approx 0$ at step zero, so $\mathbf{Q} \approx \mathrm{QR}(\mathbf{I}) = \mathbf{I}$. Gradient descent then moves $\mathbf{Q}$ *incrementally* away from the identity, only as far as the data demands. On datasets where rotation is unnecessary, $\mathbf{Q}$ naturally stays near identity and the downstream GBDT sees nearly unmodified features.

Principal angles: the perturbation $UV^\top$ has rank k, so it acts non-trivially on at most a k-dimensional subspace. Features orthogonal to that subspace are preserved up to the orthogonalization step. This is the formal sense in which low-rank rotations are "targeted": they touch only the directions where the surrogate's gradient said rotation matters.

Connection to gradient descent on Stiefel: Stiefel-manifold optimization (Edelman et al., 1998) has $d(d-1)/2$ degrees of freedom; iterates after T small steps of step size η live in a neighborhood of $\mathbf{I}$ of effective dimension $\leq \min(d^2, T)$. The low-rank parameterization fixes that effective dimension at $2dk$ *by construction*, producing the same regularizing effect without relying on early stopping.

Together these justify the empirical observation that TALR-low_rank produces near-identity perturbations with the right structural properties for downstream GBDT training, while TALR-stiefel overfits on smaller benchmarks.

C Surrogate Architecture Ablation

We ablate the differentiable surrogate over three architectures (Soft Decision Tree, NODE, MLP) on 5 datasets with 3 seeds each. All three produce statistically indistinguishable rotation quality (mean accuracy spread $< 1\%$), suggesting the benefit comes from the rotation framework rather than the specific surrogate. We use the Soft Decision Tree by default for its architectural alignment with the downstream GBDT.

D Signal/Sensor Dataset Classification

The 19 signal/sensor datasets, classified by domain provenance.

E Deep-tabular Baselines

Architectures used:

Dataset	Signal Type
libras	Gesture recognition
vehicle	Vehicle silhouette (sensor)
glass	Spectral analysis
sonar	Acoustic reflection
ionosphere	Radar returns
vowel	Speech features
phoneme	Speech phonemes
isolet	Spoken letter recognition
waveform	Synthetic waveform
mfeat-factors	Statistical features
mfeat-fourier	FFT coefficients
mfeat-karhunen	PCA projections
mfeat-morphological	Morphological features
mfeat-pixel	Pixel features
mfeat-zernike	Zernike moments
semeion	Handwritten digit images
eeg-eye-state	EEG signals
GesturePhaseSegmentation	Gesture sensor data
artificial-characters	Stroke features

- **MLP**: 256-256 hidden units, Adam, early stopping with patience 20, $L_2 = 10^{-4}$, on standard-scaled features.
- **FT-Transformer**: defaults from Gorishniy et al. (2021) via `rtdl-revisiting-models` ($d_{\text{token}} = 192$, $n_{\text{blocks}} = 3$).
- **ResNet**: $n_{\text{blocks}} = 2$, $d_{\text{block}} = 192$, dropout$= 0.15$ via `rtdl-revisiting-models`.

We use sensible defaults rather than per-dataset Optuna; the goal is to empirically substantiate the Grinsztajn et al. finding on this benchmark, not to give the deep architectures their best-possible score. Full per-dataset numbers in `results/revision/deep_baselines.parquet`.