PREDICT BY REGIONS, EXPLAIN BY EXAMPLES: INSTANCE-LEVEL EXPLANATIONS FOR TABULAR PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular prediction models are widely used in real-world decision-making, making instance-level explanations increasingly important alongside predictive performance. Existing explanation methods largely focus on feature attribution, while example-based explanations often retrieve nearby training instances without ensuring that those examples reflect the structure actually used for prediction. We propose TABERA (Tabular Explainable Region-based Architecture), a tabular prediction framework that grounds prediction and instance-level evidence in the same learned predictive regions. TABERA assigns each input to a region represented by a learned centroid and constructs its prediction from a shared regional component together with an instance-specific correction, yielding an explicit decomposition into a regional baseline and an individual adjustment. The same region is then used to retrieve observed training instances as evidence, while the retrieved cases remain outside the predictive computation. We evaluate TABERA on 21 OpenML classification datasets against 13 representative tabular prediction models. TABERA achieves competitive predictive performance while learning regions with clear label-relevant structure. Across datasets, the regional component accounts for a substantial portion of the predictive decision, and the learned regions generally contain sufficient observed instances to support region-grounded retrieval. These results show that instance-level examples can be tied directly to the model's predictive structure without making retrieved cases part of the prediction itself, providing a practical bridge between tabular prediction and example-based explanation.

1 INTRODUCTION

Tabular data underlies many real-world decision-making systems, including credit assessment, insurance, healthcare, and industrial operations (Shwartz-Ziv & Armon, 2022; Borisov et al., 2024; Lee et al., 2024; 2025). In these applications, model predictions are often delivered to customers or used directly to support consequential decisions, making transparency at the level of individual predictions particularly important (Caruana et al., 2015; Goodman & Flaxman, 2017). Accordingly, *explainability* is an important requirement for deploying predictive models in many real-world tabular applications (Rudin, 2019).

Most existing explanations for tabular predictions focus on the features of the current input. For example, when a loan application is rejected, a feature-level explanation may indicate that the decision was driven by poor repayment history, low account balance, or unstable employment (Ribeiro et al., 2016; Lundberg & Lee, 2017; Sundararajan et al., 2017). Such explanations answer an important question: *which attributes of this case mattered?* However, there is another complementary way to understand the same decision: *where does this case lie among previously observed data?* A model could, for instance, show which previously observed applicants have similar profiles and how their outcomes relate to the current case (Aamodt & Plaza, 1994; Kim et al., 2014; Papernot & McDaniel, 2018). However, providing such instance-level explanations is not straightforward, and this direction remains comparatively underexplored in tabular learning.

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

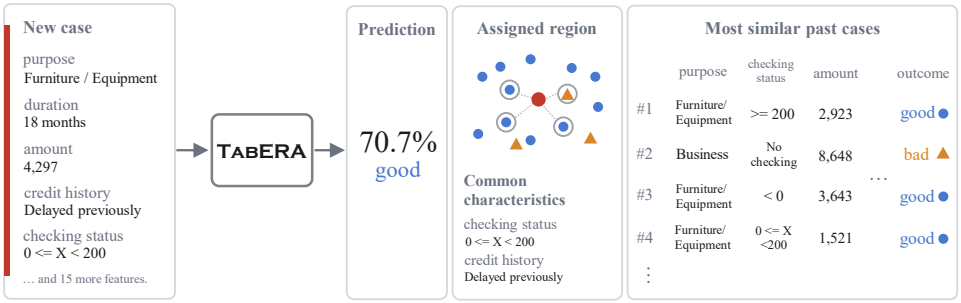


Figure 1: Illustration of TABERA on a credit prediction example. Rather than presenting only a prediction score, TABERA places the current case in the context of previously observed data by showing a group of similar cases, their common characteristics, and representative past examples. This provides concrete context for understanding how the current prediction relates to patterns observed in the training data.

The main difficulty is that similarity alone does not necessarily constitute a meaningful explanation. The nearest training samples under an arbitrary distance metric may look similar to a query, yet they may not reflect the structure that actually drives the model's prediction. For instance-level evidence to be informative, the selected cases should therefore be connected to the predictive mechanism itself, rather than retrieved independently after the prediction is made. This creates a key design challenge: *how can a tabular model expose concrete training examples that are aligned with the structure used for prediction?*

In this work, we propose TABERA (**Tab**ular **E**xplainable **R**egion-based **A**rchitecture), a tabular prediction framework that connects instance-level evidence to the model's predictive structure through learned regions, as illustrated in Figure 1. Each input is assigned to a region associated with a centroid, which provides a shared regional component for prediction. TabERA then augments the regional component with an instance-specific correction before producing the final prediction. The same assigned region is subsequently used to retrieve observed training instances as evidence, while the retrieved instances themselves do not participate in prediction. In this way, prediction and instance-level evidence are grounded in the same region structure while remaining computationally separate.

We evaluate TABERA on 21 OpenML classification datasets against 13 representative tabular prediction models. Across the benchmark, TABERA achieves competitive predictive performance while learning regions that exhibit clear label-relevant structure. Our analyses further show that the learned regions provide a meaningful shared predictive context and support instance-level evidence grounded in the same structure used for prediction. Together, these results demonstrate that TABERA can combine strong tabular prediction with region-grounded instance-level explainability.

Our main contributions are summarized as follows.

- We study instance-level explanation for tabular prediction, focusing on how an individual prediction can be contextualized through previously observed training cases rather than only through feature-level information.
- We propose TABERA, which connects prediction and instance-level evidence through a shared region structure while keeping the predictive and retrieval computations separate.
- Across 21 OpenML classification datasets, TABERA achieves competitive predictive performance while learning label-relevant regions that support region-grounded instance evidence.

2 Related Work

2.1 Explainability for Tabular Prediction

Most existing approaches to explaining tabular predictions focus on the contribution of individual input features. Post-hoc methods such as LIME, SHAP, and Integrated Gradients estimate feature-level importance (Ribeiro et al., 2016; Lundberg & Lee, 2017; Sundararajan et al., 2017), while

intrinsically interpretable models expose feature-wise effects or feature-selection patterns through the predictive structure (Rudin, 2019; Lou et al., 2013; Agarwal et al., 2021; Arık & Pfister, 2021; Si et al., 2024). Despite their different mechanisms, these methods primarily explain *which features* matter for a given prediction. A complementary question—*how the instance relates to previously observed data*—has received comparatively less attention in tabular prediction.

2.2 Example- and Prototype-Based Explanations

Instance-level explanations have been studied through case-based reasoning, representative-example selection, influence estimation, and nearest-neighbor reasoning (Aamodt & Plaza, 1994; Bien & Tibshirani, 2011; Kim et al., 2014; 2016; Koh & Liang, 2017; Yeh et al., 2018; Papernot & McDaniel, 2018). Prototype-based neural models further incorporate representative patterns into the predictive architecture, sometimes linking latent prototypes to observed examples or interpretable concepts (Li et al., 2018; Chen et al., 2019; Nauta et al., 2021). In tabular learning, PTaRL constructs a prototype-based projection space to calibrate representations, while ProtoGate employs a non-parametric prototype-based prediction mechanism (Ye et al., 2024; Jiang et al., 2024). TabERA takes a different approach: learned centroids serve as regional representatives in the predictive computation, while observed training instances from the corresponding region are presented separately as instance-level evidence.

2.3 Retrieval-Based Tabular Learning

Recent tabular models have increasingly exploited relationships among training instances for prediction. NPT and SAINT model inter-sample interactions through attention (Kossen et al., 2021; Somepalli et al., 2021), while TabR, ModernNCA, retrieval-based TabPFN variants, and ProtoGate explicitly use nearby training examples or their labels in the predictive computation (Gorishniy et al., 2024; Ye et al., 2025; Thomas et al., 2024; Hollmann et al., 2023; Jiang et al., 2024). These methods show that local sample structure can be highly informative for tabular prediction, but retrieval primarily serves a predictive role. TabERA bridges these directions by grounding prediction and retrieved explanatory examples in the same learned region, while keeping the retrieved instances outside the predictive computation.

3 Method

Our model, TabERA (Tabular Explainable Region-based Architecture), is designed to connect the predictive structure of a tabular model to observed training cases. To this end, TabERA establishes a shared region structure that supports both prediction and instance-level evidence. Its design follows three principles: (1) *prediction-aligned region structure*, where the learned regions directly participate in prediction; (2) *explicit predictive decomposition*, where each prediction is separated into a region-shared component and instance-specific variation; and (3) *region-grounded instance evidence*, where training examples associated with the same region are exposed as evidence without participating in prediction. Together, these principles place each prediction in the context of both its broader predictive region and concrete observed cases.

Overview. Given an input x, an encoder f_θ produces its representation $q = f_\theta(x) \in \mathbb{R}^D$. The representation space is partitioned into regions, each associated with a centroid, and $g(x)$ denotes the region assigned to x.

The assigned region serves two distinct roles. For prediction, its centroid provides a regional component that is combined with an instance-specific correction to form the prediction representation. For explanation, the same region assignment is used after training to retrieve relevant training examples. Only the prediction path contributes to gradient-based learning, while retrieval is performed separately and does not affect either the prediction or the training objective.

3.1 Prediction-Aligned Region Structure

We first organize the representation space into regions that directly participate in prediction. Given N training examples, we set $P = \lfloor \sqrt{N} \rfloor$ as a fixed rule across datasets, avoiding dataset-specific tuning of the region count. We maintain P unit-norm centroids $C = \{c_1, \ldots, c_P\}$, where $c_p \in \mathbb{R}^D$.

For a representation $q = f_\theta(x)$, the assigned region is determined by the centroid with the highest cosine similarity:

$$g(x) = \arg\max_p \frac{q^\top c_p}{\|q\|_2 \|c_p\|_2}. \tag{1}$$

Since the centroids are normalized, this assignment induces an angular partition of the representation space.

The centroids are initialized from randomly selected training representations and subsequently updated by exponential moving average (EMA). For each region p,

$$m_p \leftarrow \rho m_p + (1-\rho) \sum_{i:g(x_i)=p} q_i, \qquad c_p \leftarrow \frac{m_p}{\|m_p\|_2}, \tag{2}$$

where the representations used for the update are detached from the gradient graph. Thus, the centroids track the evolving representation space without being directly optimized by gradient descent.

During training, TABERA uses hard region assignments in the forward pass and a straight-through estimator (STE) (Bengio et al., 2013) to provide a differentiable surrogate for the discrete assignment. This preserves discrete region selection while allowing gradients to propagate through the assignment operation.

3.2 EXPLICIT PREDICTIVE DECOMPOSITION

The assigned centroid provides a common regional reference for all instances within the same region. To preserve instance-specific variation beyond this regional component, we augment the centroid with a correction derived from each instance's representation.

Instance-specific correction. For an input x, let $a = g(x)$ denote its assigned region and c_a the corresponding centroid. We compute the normalized direction from c_a to its representation q as $r = (q - c_a)/\max(\|q - c_a\|_2, \epsilon)$, where $\epsilon > 0$ is a small constant for numerical stability, and remove the component parallel to c_a:

$$v = r - (r^\top c_a) c_a. \tag{3}$$

Since $\|c_a\|_2 = 1$, v is orthogonal to c_a and captures instance-specific variation relative to the assigned centroid.

We then construct the prediction representation as

$$h = c_a + \beta v, \qquad \beta = \sigma(\beta_{\text{raw}}) \in (0, 1), \tag{4}$$

where β_{raw} is a learned scalar controlling the strength of the instance-specific correction. Thus, c_a provides the shared regional component, while βv adjusts the representation for each instance.

Logit decomposition. The resulting representation is passed to a shared linear prediction head, $z = Wh + b$. Substituting Eq. 4 yields

$$z = \underbrace{(Wc_a + b)}_{\text{regional baseline}} + \underbrace{\beta W v}_{\text{instance-specific correction}}. \tag{5}$$

Thus, the model output admits an exact decomposition into a regional baseline shared by instances in the same region and an instance-specific correction. The region used for subsequent instance-level retrieval therefore corresponds to a component that directly participates in prediction.

3.3 REGION-GROUNDED INSTANCE EVIDENCE

We use the region-shared component of the prediction to provide broader context for each individual instance. While the instance-specific correction captures variation within a region, the regional baseline is shared by all instances assigned to that region. We therefore retrieve observed training instances from the same region as instance-level evidence, grounding the retrieved examples in a component that directly participates in prediction.

During training, we store detached representations of training instances together with their region assignments. After training, these stored representations are used for instance-level retrieval. For an input x, we select the k most similar training instances within its assigned region using cosine similarity between representations. If fewer than k instances are available, candidates are added from other regions in descending order of centroid cosine similarity, followed by the remaining training instances if necessary.

The retrieved instances are used only as evidence and do not participate in prediction or optimization. Thus, they provide observed examples associated with the shared predictive context of the input without changing the model output.

Finally, TABERA is trained using only the supervised cross-entropy objective,

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}}(z, y). \tag{6}$$

The encoder, prediction head, and β_{raw} are optimized by gradient descent, while the centroids are updated by EMA as described in Section 3.1.

4 RESULTS

4.1 EXPERIMENTAL SETUP

We evaluate TABERA on 21 OpenML classification datasets (Vanschoren et al., 2014) using established training and evaluation conventions (Lee et al., 2025). For each dataset, we report the average performance over five splits.

We compare against 13 baselines spanning four perspectives on tabular prediction and explainability. First, *tree-based* models (Chen & Guestrin, 2016; Ke et al., 2017; Prokhorenkova et al., 2018) provide strong predictive baselines whose explanations are typically expressed through feature importance. Second, *general neural* models (Gorishniy et al., 2021; 2022), including MLP, EmbedMLP, MLP-PLR, and ResNet, do not include an explicit mechanism for feature- or instance-level explanation. Third, *feature-interaction* models (Gorishniy et al., 2021; Yan et al., 2023; Somepalli et al., 2021), including FT-Transformer, T2G-Former, and SAINT, explicitly model interactions among tabular attributes through attention-based architectures, with SAINT additionally incorporating interactions across samples. Finally, *instance-relational* models (Ye et al., 2025; Gorishniy et al., 2024), ModernNCA and TabR, exploit relationships among training instances for prediction, making them the closest baselines to our instance-level perspective; however, the neighboring or retrieved instances are part of the predictive mechanism rather than a separate evidence pathway.

For TABERA, numerical features are represented using piecewise linear embeddings (PLE) (Gorishniy et al., 2022), while categorical features are one-hot encoded; the resulting features are then passed through a multi-layer encoder. We use a fixed retrieval size of $k = 8$ across all datasets. The model is trained with AdamW (Loshchilov & Hutter, 2019) for up to 100 epochs with early stopping based on validation loss. Hyperparameter optimization follows the same protocol and search budget used for the baseline models in (Lee et al., 2025). Full implementation details and hyperparameter ranges are provided in the Appendix. All experiments are conducted on an NVIDIA GeForce RTX 5080 GPU.

4.2 PREDICTIVE PERFORMANCE

As shown in Table 1, TABERA achieves an average accuracy of 0.850, tied for the second-highest result overall, and an F1-score of 0.791, tied for the best result. Although TABERA is primarily designed to provide instance-level evidence, separating retrieval from the predictive path does not appear to require a substantial trade-off in predictive performance. TabERA performs on par with strong tree-based models, which remain highly competitive for tabular prediction, and outperforms all feature-interaction baselines based on attention architectures. It also remains competitive with instance-relational approaches that explicitly exploit relationships among training instances during prediction. Most notably, compared with TabR, the closest retrieval-based baseline to our setting, TABERA achieves higher performance on both Accuracy and F1-score. These results suggest that instance-level retrieval can be separated from the predictive computation while retaining strong predictive performance.

Table 1: Predictive performance across 21 OpenML classification datasets, averaged over five splits. F1 denotes positive-class F1 for binary classification and support-weighted F1 for multiclass classification. Best and second-best results are shown in **bold** and underline, respectively.

Model Group	Model	Accuracy ↑	F1-Score ↑
Tree-based	XGBoost	0.846	0.763
	LightGBM	<u>0.850</u>	<u>0.783</u>
	CatBoost	0.831	0.759
	Random Forest	0.848	0.779
General Neural	MLP	0.829	0.747
	EmbedMLP	0.842	0.768
	MLP-PLR	0.839	0.757
	ResNet	0.847	**0.791**
Feature-Interaction	FT-Transformer	0.838	0.768
	T2G-Former	0.820	0.737
	SAINT	0.844	0.780
Instance-Relational	ModernNCA	**0.853**	**0.791**
	TabR	0.843	0.773

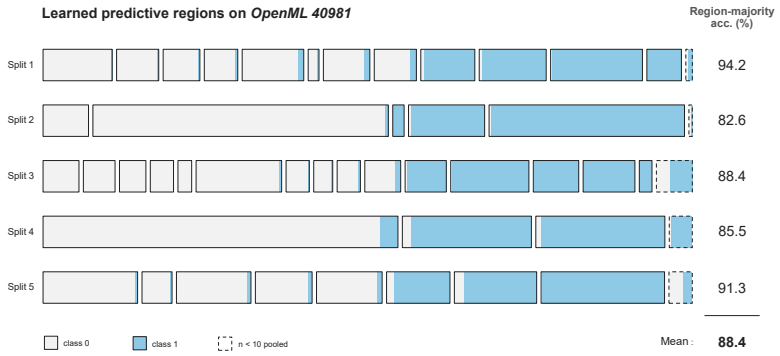


Figure 2: Label composition of predictive regions learned on OpenML 40981 across five predefined splits. Block width is proportional to region size, and color indicates the training-label composition within each region. Regions with fewer than 10 training instances are pooled for visualization. The values on the right show region-majority test accuracy for each split.

4.3 Structure of Learned Predictive Regions

TabERA shows its largest relative improvement on OpenML 40981, achieving an average accuracy of 0.884 and F1-score of 0.869, compared with 0.872 and 0.853 for the strongest baseline. We use this dataset to examine how the learned regions organize the predictive representation space.

Figure 2 shows the training-label composition of the regions learned independently in each of the five data splits. Across splits, many regions are strongly concentrated toward a single class, indicating that the learned partition repeatedly captures label-relevant structure. We further quantify this tendency using *region-majority accuracy*: each test instance is assigned the majority training label of its corresponding region. This yields an average accuracy of 88.4%, compared with 56.2% from the global training-set majority label. The analysis suggests that the predictive regions learned by TabERA capture structure that remains informative for held-out instances.

5 Discussion

5.1 Prediction Structure and Instance Evidence

We next examine how the three design principles of TabERA manifest across the benchmark. Specifically, we analyze whether (1) the learned regions capture label-relevant predictive structure, (2) the regional component and instance-specific correction play distinguishable roles in prediction,

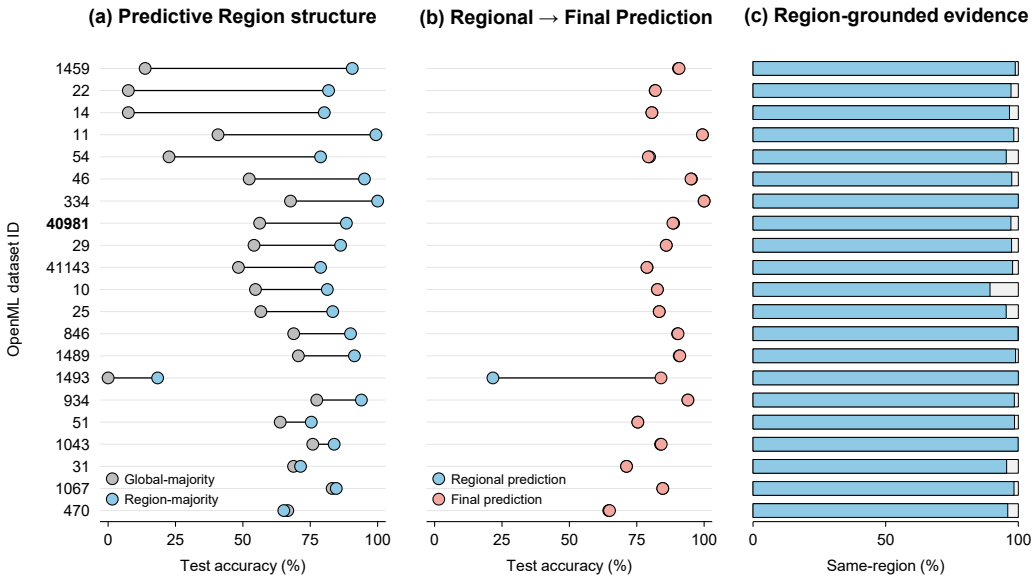


Figure 3: Analysis of TabERA's predictive structure across 21 datasets. (a) Global-majority and region-majority accuracy, measuring the label-relevant structure captured by the learned regions. (b) Regional and final prediction accuracy, showing the effect of the instance-specific correction. On most datasets, the difference between the two is small, resulting in substantial overlap between the corresponding points. (c) Same-region retrieval share with $k = 8$, showing the extent to which instance-level evidence can be retrieved from the assigned region without fallback.

and (3) the learned regions provide sufficient support for instance-level retrieval. Figure 3 summarizes the results.

Prediction-aligned region structure. We first examine whether the learned regions capture label-relevant predictive structure across datasets. To isolate the information provided by the region assignment, we compare two simple majority-rule predictions. *Global-majority accuracy* assigns every test instance the most frequent label in the full training set, whereas *region-majority accuracy* assigns the majority training label of the region to which each test instance is mapped. The difference between the two therefore measures how much label-relevant structure is introduced by conditioning on the learned region. As shown in Figure 3(a), region-majority accuracy consistently exceeds global-majority accuracy across datasets. Averaged over the 21 datasets, global-majority accuracy is 50.35%, whereas region-majority accuracy reaches 81.82%, a gain of 31.47 percentage points; region-majority accuracy is higher on 20 of the 21 datasets. Thus, the learned region assignments provide substantially more label-relevant structure than the global class distribution, supporting their role as prediction-aligned groups rather than arbitrary partitions of the representation space.

Regional and final predictions. We next examine how the instance-specific correction changes the prediction relative to the shared regional component. We define the *regional prediction* using only the regional baseline Wc_a+b, and compare it with the final prediction after adding the instance-specific correction. As shown in Figure 3(b), the regional prediction achieves an average accuracy of 82.04%, compared with 85.03% for the final prediction. At the instance level, the predicted class changes after applying the correction for only 3.96% of test instances on average. These results indicate that the regional component accounts for most final decisions, while the instance-specific correction provides additional adjustment for a smaller subset of instances.

Region support for instance retrieval. Finally, we examine whether the learned regions provide sufficient support for instance-level retrieval. TabERA sets the number of regions using the fixed rule $P = \lfloor\sqrt{N}\rfloor$, avoiding dataset-specific tuning, and does not explicitly constrain the number of training instances assigned to each region. This could produce sparsely populated regions that frequently require retrieval to expand beyond the assigned region. As shown in Figure 3(c), with

$k = 8$, 97.48% of retrieved instances are drawn from the assigned region on average across datasets, while fallback to other regions is required for only 3.54% of test instances. These results indicate that the fixed region-count rule provides sufficient retrieval support in practice without additional region-size tuning or balancing.

5.2 Comparison with Retrieval-Based Prediction

Among our baselines, TabR (Gorishniy et al., 2024) is the closest comparison to TabERA in that it explicitly retrieves training instances. The key distinction lies in how retrieval is used: in TabR, retrieved instances participate directly in the predictive computation, whereas TabERA separates instance retrieval from the predictive path and uses retrieved cases only as evidence. As shown in Table 1, TabERA maintains competitive predictive performance under this separation and outperforms TabR on both Accuracy and F1-score.

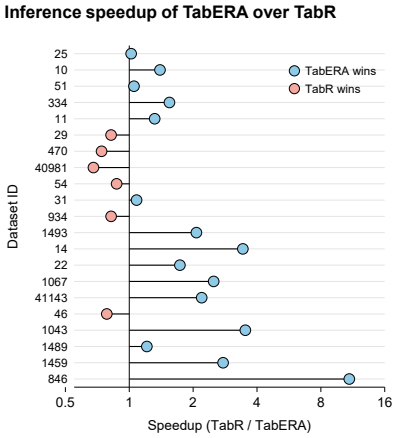


Figure 4: Inference efficiency of TabERA and TabR across 21 OpenML datasets. Speedup is computed as $T_{\text{TabR}}/T_{\text{TabERA}}$, such that values above 1 indicate faster inference with TabERA and values below 1 indicate faster inference with TabR. Datasets are ordered by training-set size, with larger datasets appearing lower on the y-axis.

We further examine the computational consequence of this design through inference efficiency. Because TabERA performs prediction without accessing the training-instance memory, retrieval is required only when instance-level evidence is requested. Figure 4 compares the inference speed of the two models across the 21 datasets, ordered by increasing training-set size. TabERA is faster on 15 of the 21 datasets, with a median speedup of $1.41\times$. While TabR is faster on several smaller datasets, the relative advantage of TabERA generally becomes more pronounced as the training-set size increases. This pattern is consistent with the computational distinction between the two approaches: TabR accesses training instances as part of prediction, whereas TabERA does not. Together, these results indicate that separating retrieval from the predictive path can preserve strong predictive performance while providing an inference-efficiency advantage, particularly on larger datasets.

5.3 Instance-Level Explanation Example

A central goal of TabERA is to provide instance-level explanations that remain directly tied to the structure used for prediction. Conventional post-hoc explanations may identify important features or retrieve nearby training samples after a prediction has already been made, but such information need not reflect the representation that actually determined the model's decision. In contrast, TabERA grounds both prediction and explanation in the same learned region. This allows each prediction to be explained from three complementary views: the characteristic profile of the assigned predictive region, the position of the current instance within that region, and observed training cases drawn from the same region. Together, these views provide a coherent account of the predictive context rather than an independently constructed post-hoc neighborhood.

Figure 5 illustrates this explanation mechanism using two test instances from the `credit-g` binary classification task. Figure 5(a) shows Sample 18, which TabERA predicts as `good` with probability 0.941. The assigned predictive region contains 122 training instances, of which 117 (96%) are also labeled `good`, indicating a strong class tendency within the region. The region profile further reveals characteristics that are common in this predictive context: `property_magnitude=real estate` appears in 80% of the region and `checking_status=no checking` in 66%, both matching the current instance. At the same time, the within-region comparison identifies where the instance is atypical. Its `credit_amount` of $3{,}447$ exceeds the regional median of $1{,}404$, while `purpose=retraining` and `credit_history=all paid` occur in only 2 and 3 regional instances, respectively. Finally, the explanation presents eight observed training cases from the same

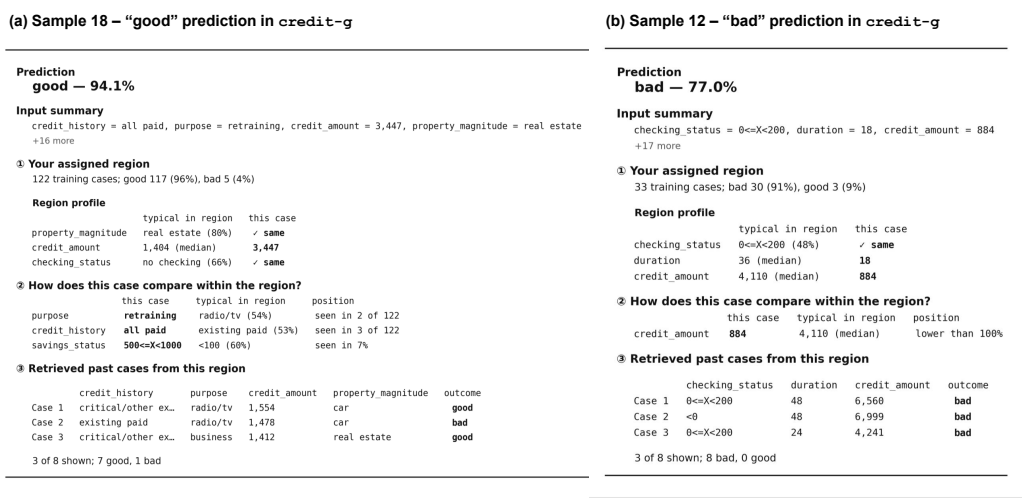


Figure 5: Instance-level explanation examples from OpenML 31 (`credit-g`). (a) Sample 18, predicted as `good`, and (b) Sample 12, predicted as `bad`. For each instance, TABERA presents the assigned predictive region, the instance's position within that region, and retrieved training cases from the same region.

region, seven of which are labeled `good`. Taken together, the explanation shows not only why the assigned region supports the predicted class, but also how the current instance aligns with and departs from the learned regional pattern.

Figure 5(b) illustrates the same explanation mechanism for the opposite class. Sample 12 is predicted as `bad` with probability 0.770, and 30 of the 33 training instances (91%) in its assigned region share that label. Unlike Sample 18, only `credit_amount` satisfies the criterion for an unusual within-region position. This difference highlights that TABERA does not impose a fixed set of explanatory features; instead, the displayed explanation adapts to the structure of the predictive region assigned to each instance.

Overall, these examples illustrate how TABERA turns the learned predictive region into a structured instance-level explanation. The region profile summarizes the local decision context, the within-region comparison shows how the current instance conforms to or deviates from that context, and the retrieved training cases provide concrete observations associated with the same region. Crucially, all three views are derived from the region assignment already used for prediction, so the explanation remains aligned with the model's predictive structure rather than introducing a separate post-hoc notion of similarity.

6 CONCLUSION

We introduced TABERA, a tabular prediction framework that connects individual predictions to observed training cases through a shared predictive region structure. TABERA combines a region-shared component with an instance-specific correction, yielding an explicit decomposition of the model output, and retrieves training instances from the corresponding region as evidence without using them in the predictive computation. Across 21 classification datasets, TABERA maintains competitive predictive performance while learning regions that capture label-relevant structure and provide meaningful context for individual predictions. A limitation of the current framework is that it focuses on instance-level evidence without explicitly incorporating feature-level attribution. In addition, the interpretability of the retrieved evidence depends on the quality of the learned region structure. Combining region-grounded instance evidence with feature-level explanations is a promising direction for future work.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, manuscript organization, and refinement of the presentation of the paper. All AI-assisted text was reviewed and revised by the authors, who verified the technical content, claims, and references. Generative AI was not used to generate experimental results or to make final scientific conclusions. The authors take full responsibility for the content of this work.

ETHICS STATEMENT

This work uses publicly available tabular benchmark datasets and does not involve human-subject experiments or the collection of new personal data. TabERA is intended to improve transparency in tabular prediction by providing instance-level evidence; however, retrieved examples may raise privacy concerns when applied to sensitive real-world data. Such deployments should therefore consider appropriate anonymization and access-control mechanisms.

REPRODUCIBILITY STATEMENT

We provide the model specification, training procedure, evaluation protocol, preprocessing details, and hyperparameter settings in the main paper and appendix. The source code will be released publicly on GitHub upon publication to facilitate reproduction of our results.

REFERENCES

Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1):39–59, 1994. doi: 10.3233/AIC-1994-7104.

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Benjamin Lengerich, Rich Caruana, and Geoffrey E. Hinton. Neural additive models: Interpretable machine learning with neural nets. In *Advances in Neural Information Processing Systems 34*, pp. 4699–4711, 2021.

Sercan Ö. Arık and Tomas Pfister. TabNet: Attentive interpretable tabular learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):6679–6687, 2021. doi: 10.1609/aaai.v35i8.16826.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, 5(4):2403–2424, 2011.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(6):7499–7519, 2024. doi: 10.1109/TNNLS.2022.3229161.

Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for HealthCare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1721–1730, 2015.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K. Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems 32*, pp. 8928–8939, 2019.

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. doi: 10.1145/2939672.2939785.

Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, 2017. doi: 10.1609/aimag.v38i3.2741. arXiv:1606.08813.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Advances in Neural Information Processing Systems 34*, pp. 18932–18943, 2021.

Yury Gorishniy, Ivan Rubachev, and Artem Babenko. On embeddings for numerical features in tabular deep learning. In *Advances in Neural Information Processing Systems 35*, 2022.

Yury Gorishniy, Ivan Rubachev, Nikolay Kartashev, Daniil Shlenskii, Akim Kotelnikov, and Artem Babenko. TabR: Tabular deep learning meets nearest neighbors. In *International Conference on Learning Representations*, 2024. arXiv:2307.14338.

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.

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 *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *PMLR*, pp. 21844–21878, 2024. arXiv:2306.12330.

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 30*, pp. 3149–3157, 2017.

Been Kim, Cynthia Rudin, and Julie Shah. The Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in Neural Information Processing Systems 27*, pp. 1952–1960, 2014. arXiv:1503.01161.

Been Kim, Rajiv Khanna, and Oluwasanmi Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems 29*, pp. 2288–2296, 2016.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Jannik Kossen, Neil Band, Clare Lyle, Aidan N. Gomez, Thomas Rainforth, and Yarin Gal. Self-attention between datapoints: Going beyond individual input-output pairs in deep learning. In *Advances in Neural Information Processing Systems 34*, 2021. arXiv:2106.02584.

Kyungeun Lee, Ye Seul Sim, Hye-Seung Cho, Moonjung Eo, Suhee Yoon, Sanghyu Yoon, and Woohyung Lim. Binning as a pretext task: Improving self-supervised learning in tabular domains. *arXiv preprint arXiv:2405.07414*, 2024.

Kyungeun Lee, Moonjung Eo, Hye-Seung Cho, Dongmin Kim, Ye Seul Sim, Seoyoon Kim, Min-Kook Suh, and Woohyung Lim. MultiTab: A comprehensive benchmark suite for multi-dimensional evaluation in tabular domains. *arXiv preprint arXiv:2505.14312*, 2025.

Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

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*, 2013.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30*, 2017.

Meike Nauta, Ron van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Nicolas Papernot and Patrick McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. CatBoost: Unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems 31*, 2018.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016. doi: 10.1145/2939672.2939778.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019. doi: 10.1038/s42256-019-0048-x.

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

Jacob Si, Wendy Yusi Cheng, Michael Cooper, and Rahul G. Krishnan. InterpreTabNet: Distilling predictive signals from tabular data by salient feature interpretation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *PMLR*, pp. 45353–45405, 2024.

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. *arXiv preprint arXiv:2106.01342*, 2021.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Valentin Thomas, Junwei Ma, Rasa Hosseinzadeh, Keyvan Golestan, Guangwei Yu, Maksims Volkovs, and Anthony Caterini. Retrieval & fine-tuning for in-context tabular models. In *Advances in Neural Information Processing Systems 37*, 2024. arXiv:2406.05207.

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

Jiahuan Yan, Jintai Chen, Yixuan Wu, Danny Z. Chen, and Jian Wu. T2G-Former: Organizing tabular features into relation graphs promotes heterogeneous feature interaction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Han-Jia Ye, Huai-Hong Yin, De-Chuan Zhan, and Wei-Lun Chao. Revisiting nearest neighbor for tabular data: A deep tabular baseline two decades later. In *International Conference on Learning Representations*, 2025. arXiv:2407.03257.

Hangting Ye, Wei Fan, Xiaozhuang Song, Shun Zheng, He Zhao, Dandan Guo, and Yi Chang. PTaRL: Prototype-based tabular representation learning via space calibration. In *The Twelfth International Conference on Learning Representations*, 2024.

Chih-Kuan Yeh, Joon Sik Kim, Ian E. H. Yen, and Pradeep Ravikumar. Representer point selection for explaining deep neural networks. In *Advances in Neural Information Processing Systems 31*, pp. 9291–9301, 2018. arXiv:1811.09720.