Explain the Synth: Interpretable Evaluation of LLM Data Synthesis

Yue Yang[1,2*†], **Fan Yang**[1*], **Yu Bai**[1], **Hao Wang**[1†]
[1] Faculty of IT, Monash University, Australia [2] Maincode, Australia
{yue.yang1,fan.yang1,yu.bai3,hao.wang2}@monash.edu

Abstract

Large language models (LLMs) are increasingly used to generate synthetic data, in which tabular data constitute a fundamental data modality across a wide range of domains. Yet, current evaluation practices often provide limited insights into whether the synthetic data preserve real data-generating relationships or introduce plausible-looking artifacts. We present a conceptually simple, interpretable auditing framework that compares the explanatory structure induced by real versus synthetic data. The key idea is to use a transparent rule-based model as a shared explanatory language: we extract rules from real data to summarize how features relate to labels, then examine how this rule structure changes when explained using LLM-generated data. Importantly, these rules are derived by an independent rule auditor rather than by the generator itself. The resulting "explanation shift" reveals which relationships are preserved, weakened, removed, or newly introduced by the generator, offering actionable diagnostics beyond aggregate fidelity scores. We further provide a theoretical perspective that links explanation shift and cross-domain predictive gaps to distribution mismatch within an interpretable hypothesis class. Overall, our approach turns synthetic data evaluation into a human-auditable comparison of explanations, improving transparency for LLM-based tabular synthesis.

1 Introduction

Tabular data are widely used across high-stakes domains, such as healthcare, finance, energy systems, and public policy (Giuffrè and Shung, 2023; Shwartz-Ziv and Armon, 2022). Yet, access to high-quality tabular datasets is frequently constrained by privacy regulations, limited sample sizes, data missing, demographic imbalance, and organizational barriers to data sharing (Fonseca and Bacao, 2023). As a result, synthetic data generation has become a widely used approach for mitigating data access constraints and enabling model development and benchmarking (Miletic and Sariyar, 2024).

Large language models (LLMs) offer an attractive new paradigm for tabular synthesis. Unlike classical tabular generative models that require training specialized architectures, LLMs can be prompted with schema descriptions, constraints, and a small set of examples to produce synthetic rows in a training-free manner. Compared with traditional tabular synthesizers, such as CTGAN and TVAE (GAN/VAE-based) (Xu et al., 2019) and TabDDPM (diffusion-based), LLM-based methods like GReaT (Borisov et al., 2022) and TabuLa (Zhao et al., 2025) also offer a more flexible paradigm by modeling each row autoregressively as a sequence. This makes conditional generation and schema-constrained sampling straightforward. However, this flexibility introduces new risks: LLM-based generators can distort global statistics, amplify biases contained in prompts (Gallegos et al., 2024), or create spurious dependencies (Recasens et al., 2025) that appear plausible but do not reflect the real data-generating process. As a result, practitioners increasingly require not only *scores* of synthetic quality, but also *interpretable diagnostics* (Kapar et al., 2025) that explain *what* differs between real and synthetic distributions and *why* a synthetic dataset may fail in downstream tasks.

Existing synthetic-data evaluations largely fall into two categories. The first uses distributional similarity metrics (Stolte et al., 2024), which are useful for detection but often provide limited insights into the concrete feature–label relationships that were preserved or corrupted. The second evaluates downstream utility by training predictive models on synthetic data and testing on real data (or vice versa) (Xu et al., 2019). This approach provides an outcome-based score but typically lacks

[*] Equal contribution.
[†] Corresponding authors: Yue Yang and Hao Wang.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 43054–43077

interpretability and can confound distribution mismatch with model bias. Both categories may flag that synthetic data are problematic, but they do not readily explain *which* dependencies or shortcuts the generator introduced, nor do they offer a concise mechanism-level "diff" that can guide remediation (e.g., prompt redesign, constraint tightening, or post-generation filtering) (Rudin, 2019). However, unlike traditional explanation methods for deep learning models (Lundberg and Lee, 2017; Kulinski and Inouye, 2023; Adebayo et al., 2018; Lapuschkin et al., 2019), explaining LLM-generated synthetic data is difficult because generation is driven by high-dimensional, prompt-sensitive internal representations, so the same sample can arise from different hidden mechanisms and any produced rationale may be plausible but not faithful.

In this work, we propose an interpretability-driven auditing framework using rules to *explain and audit* LLM-generated synthetic tabular data through the lens of an interpretable hypothesis class. Our core idea is to treat an interpretable learner as a *probe* of the joint distribution and to compare the best explanations induced by real versus synthetic data. Concretely, we first train a gradient-boosted rule ensemble on the real dataset to obtain an interpretable explanation, which we call *Rule set A*. We then generate a synthetic dataset using an LLM conditioned on the same schema and task specification, and we evaluate the synthetic dataset using Rule set A as a baseline audit. Finally, we warm-start the rule learner from A and continue training on the synthetic dataset until convergence, producing *Rule set B*. The transition from A to B provides a direct, human-auditable summary of which feature–label relationships are preserved, weakened, removed, or newly introduced by the synthetic generator. Additionally, we establish a theoretical connection between rule convergence and synthetic-data quality, proving that faithful synthetic data produce explanations that transfer reliably to real data, while large cross-domain gaps reveal artifacts, such as spurious correlations, leakage, or distribution shift.

We summarize the contributions as follows:

- We turn LLM-generated synthetic-data evaluation into a human-auditable comparison of explanations, using rule models as a shared explanatory language.
- We propose four metrics that quantify utility transfer and synthetic drift. We also provide theoretical guarantees characterizing when drift can be fixed by reweighting existing rules and provide bounds for cross-domain utility transfer.
- We evaluate our framework across six tabular benchmarks and multiple LLM-based generators, showing why score-only evaluation can be misleading and demonstrating how rule differences localize concrete mechanism inversions.

Overall, our approach advances the transparency of LLM-based tabular synthesis by moving beyond opaque quality score-only evaluation to interpretable, auditable rule-based explanations.

2 Related Work

2.1 Tabular Data Generation Based on LLMs

LLMs have emerged as an effective approach to tabular synthesis. GReaT (Borisov et al., 2022) demonstrates that autoregressive models can capture complex feature dependencies by serializing tables into token sequences, matching the performance of specialized tabular data generators. This work inspires several extensions. REaLTabFormer (Solatorio and Dupriez, 2023) handles relational databases with foreign-key constraints, and HARMONIC (Wang et al., 2024) incorporates privacy-preserving mechanisms through instruction tuning. Unlike classical methods that require training a new model for each dataset, these LLM-based approaches can generate synthetic data from just a schema and a few examples (Fonseca and Bacao, 2023). This allows LLMs to be applied without dataset-specific training, since they reuse a pretrained language model instead of learning an explicit generative model for each table. Classical tabular generators, including Copulas-based methods (Patki et al., 2016), Bayesian networks (Zhang et al., 2017), GAN-based methods such as TVAE and CTGAN (Xu et al., 2019), and diffusion models, such as TabDDPM (Kotelnikov et al., 2023) and TabSyn (Zhang et al., 2023), typically learn joint, marginal, or conditional distributions through specialized architectures. In contrast, LLMs rely on implicit statistical regularities learned from data, without encoding such distributional constraints. While this enables flexible prompt-based synthesis, it permits plausible-looking correlations that do not correspond to true dependencies in the real data.

As a result, standard distributional similarity tests often fail to detect these subtle structural artifacts.

2.2 Quality Assessment of Synthetic Data

The existing comprehensive data evaluation frameworks are generally divided into two categories. The first approach measures the distributional similarity through distributional similarity metrics, including marginal distances, correlation test, and classifier two-sample test, such as C2ST (Lopez-Paz and Oquab, 2016). The second approach is to evaluate downstream utility through TSTR (train on synthetic, test on real), which is a popular paradigm in tabular synthesis evaluation (Xu et al., 2019). Variants include machine learning efficiency metrics and privacy measures, such as the distance to the nearest record. Recent efforts to unify these aspects within a privacy-utility-fidelity trinity framework (Hernandez et al., 2025) have led to platforms, such as SynthEval (Lautrup et al., 2025), which provide comprehensive benchmarking toolkits. However, both approaches lack diagnozability. A lower C2ST score or poor TSTR accuracy indicates a problem, but it cannot explain whether the problem stems from marginal drift, conditional dependency errors, or spurious dependencies. Recent studies have also questioned the reliability of distance-based privacy metrics (Yao et al., 2025). Our method complements these aggregated scores by providing interpretable, human-auditable diagnostics revealing how feature–label relationships differ between the real and synthetic data distributions.

2.3 Interpretable Rule Learning

Rule-based models have a long history in interpretable machine learning. Early work includes RIPPER (Cohen, 1995), which constructs rule sets by incrementally growing and pruning rules, then iteratively optimizing them to minimize error. RuleFit (Friedman and Popescu, 2008) combines tree-derived rules and Lasso regularization to generate sparse rule sets. Recent methods include Explainable Boosting Machines (EBM) (Nori et al., 2019), which extends generalized additive models through cyclic gradient boosting. SIRUS (Bénard et al., 2021) provides stable rule extraction from random forests. However, these methods often greedily add rules without enforcing structural orthogonality, making it difficult to distinguish which dependencies are genuinely new rather than redundant reweights of existing patterns. In this work, we adopt Orthogonal Gradient Boosting (OGB) (Yang et al., 2024), which explicitly constrains newly learned rules to remain orthogonal to the previously selected rules. This structural orthogonality is central to our proposed audit framework: when warm-starting the OGB on real data and continue training on synthetic data, newly added rules can capture residual structure in the original model rather than redundantly reweighting the existing dependencies (Yang et al., 2025, 2026).

2.4 Explanation Shift and Distribution Drift

Using model interpretations to monitor distribution changes is an emerging direction in machine learning reliability. Mougan et al. (2023) introduced the concept of "explanation shift" by comparing SHAP-based (Lundberg and Lee, 2017) feature attributions across distributions, developing the *skshift* library for deployment monitoring. Kulinski and Inouye (2023) proposed learning interpretable transportation maps via optimal transport relaxations to explain distribution shifts across tabular, text, and image data. Another work by Rabanser et al. (2019) provides a systematic empirical study of dataset shift detection methods. Additionally, explainable AI techniques have been applied to diagnose distributional artifacts in deployed models (Adebayo et al., 2018; Lapuschkin et al., 2019).

Unlike these methods that primarily address temporal drift or domain adaptation using SHAP values or optimal transport, we employ rule-based explanations for auditing LLM-generated synthetic tabular data. Our method yields directly auditable conditionals (e.g., threshold tests) that closely align with how practitioners inspect tabular relationships. Furthermore, we provide theoretical analysis linking rule convergence to synthetic quality through hypothesis-class discrepancy bounds.

3 Preliminary

3.1 Exponential Family Distribution

Let P denote the *real* joint distribution of a pair of random variables (X, Y) and Q denote the *synthetic* (LLM-generated) joint distribution of (X, Y). Assume the output Y conditional on the input X follows a (regular) exponential family with canonical parameter θ:

$$p(Y = y \mid \theta) = h(y) \exp\big(y\theta - A(\theta)\big), \quad (1)$$

where $A(\theta)$ is the log-partition function, and h is a base measure (McCullagh, 2019). It covers Gaus-

sian (MSE up to constants), Bernoulli (logistic), Poisson, Gamma (canonical), etc. For the multi-class setting, one may instead use the multinomial exponential family with a softmax link; see Remark A. In canonical-link generalized linear models (GLMs), the *linear predictor* (a.k.a. natural parameter) is $\theta(x) = f(x)$, where f is a linear model.

Up to y-only constants, the per-sample loss $\ell(\cdot, \cdot)$ for canonical exponential family is

$$\ell(f(x), y) = A(f(x)) - yf(x). \tag{2}$$

Lemma 3.1. *For the canonical exponential family loss in Eq. (2), let $z = f(x)$, then:*

1. *$\ell(z, y)$ is convex in z since $A(z)$ is convex.*
2. *The gradient and Hessian are*

$$\begin{aligned} \frac{\partial}{\partial z}\ell(z, y) &= A'(z) - y, \\ \frac{\partial^2}{\partial z^2}\ell(z, y) &= A''(z) \geq 0. \end{aligned} \tag{3}$$

3. *$A'(z) = \mathbb{E}[Y \mid \theta = z]$ and $A''(z) = \mathrm{Var}(Y \mid \theta = z)$ for regular families.*

3.2 Orthogonal Gradient Boost Additive Rule Ensemble

We use an additive rule (Friedman and Popescu, 2008) with K rules

$$f_K(x) = \beta_0 + \sum_{j=1}^{Ks} \beta_j q_j(x), \quad q_j(x) \in \{0, 1\}, \tag{4}$$

where q_j denotes the query (condition, i.e., the "if" part) of the j-th rule, and β_j is the weight (the "then" part) of the j-th rule. Given a dataset $\{(x_i, y_i)\}_{i=1}^n$ drawn i.i.d. from the distribution over (X, Y), define $\hat{\boldsymbol{f}} := (f(x_1), \ldots, f(x_n))^\top$, $\boldsymbol{y} := (y_1, \ldots, y_n)^\top$, and query output vectors $\boldsymbol{q}_j := (q_j(x_1), \ldots, q_j(x_n))^\top$.

For $\lambda \geq 0$, define regularized empirical risk

$$\begin{aligned} \widehat{R}_\lambda(f) &= \frac{1}{n}\sum_{i=1}^n \ell(f(x_i), y_i) + \frac{\lambda}{2n}\|\boldsymbol{\beta}\|_2^2 \\ &= \frac{1}{n}\sum_{i=1}^n \left(A(\hat{f}_i) - y_i\hat{f}_i\right) + \frac{\lambda}{2n}\|\boldsymbol{\beta}\|_2^2, \end{aligned} \tag{5}$$

where $\boldsymbol{\beta}^{(t)} = (\beta_1^{(t)}, \ldots, \beta_t^{(t)})^T$.

Let $\Phi_t = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_t]$ and $\boldsymbol{g} = \nabla\widehat{R_\lambda}(f)$. At iteration t, the query q_t is selected by maximizing the OGB objective function (Yang et al., 2024):

$$\mathrm{obj}_{\mathrm{OGB}}(\boldsymbol{q}) := \left|\boldsymbol{g}_\perp^\top \boldsymbol{q}_\perp\right| / \left(\|\boldsymbol{q}_\perp\|_2 + \varepsilon\right), \tag{6}$$

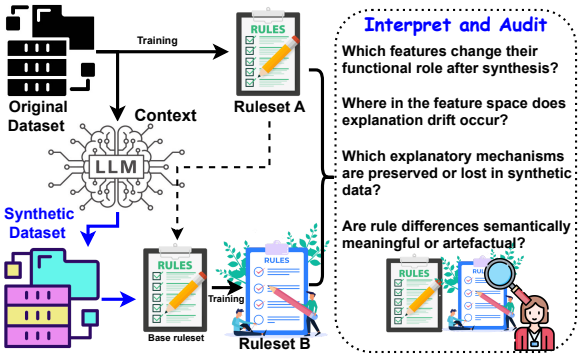


Figure 1: A mechanism-aware auditing pipeline.

where $\boldsymbol{g}_\perp$ and $\boldsymbol{q}_\perp$ are the projection of $\boldsymbol{g}$ and $\boldsymbol{q}$ onto the complement of $\mathrm{range}(\Phi_t)$. This objective function guarantees that the selected query is not a linear combination of existing queries. After selecting query q_t, we refit all coefficients:

$$(\beta_0^{(t)}, \boldsymbol{\beta}^{(t)}) \in \arg\min_{\beta_0 \in \mathbb{R}, \boldsymbol{\beta} \in \mathbb{R}^t} R_\lambda\left(\beta_0 \mathbf{1} + \Phi_t \boldsymbol{\beta}\right). \tag{7}$$

4 Methodology

We consider a supervised tabular learning task with a real (observed) labeled dataset $D_P = \{(x_i, y_i)\}_{i=1}^n$, $x_i \in \mathbb{R}^d$, $y_i \in \mathcal{Y}$, drawn i.i.d. from an unknown real distribution P over (X, Y). We assume access to an LLM-based generator $\mathcal{G}$ that produces a synthetic labeled dataset $D_Q = \{(x_j^{(Q)}, y_j^{(Q)})\}_{j=1}^m \sim Q$, conditioned on a table schema $\mathcal{S}$ (e.g., feature names, types, and ranges) and optional generation constraints $\mathcal{C}$ (e.g., validity checks, domain rules, and label definitions). Our goal is to *explain and audit* whether the synthetic distribution Q preserves the predictive structure present in P, and to pinpoint which feature–label dependencies are preserved, weakened, removed, or newly introduced by the LLM generator. To obtain human-auditable explanations, we restrict attention to an interpretable hypothesis class $\mathcal{H}$ of additive rule ensembles f_K with K rules. Each query q_k is a conjunction of simple predicates over features (e.g., threshold tests and category membership), so that $q_k(x) = 1$ corresponds to an explicit "if" condition being satisfied.

4.1 Core Pipeline: Rule set A → Rule set B

Step 1: Learn a real-data explanation (Rule set A). As shown in Figure 1, we fit a rule-based predictor on the real dataset D_P using Orthogonal Gradient Boosting (OGB) with correc-

tive refitting, yielding: (i) a set of learned queries $A = \{q_1^{(A)}, \ldots, q_{K_A}^{(A)}\}$, (ii) corresponding coefficients $\boldsymbol{\beta}^{(A)}$, and (iii) the real-data explanation model $f_A(x) = \beta_0^{(A)} + \sum_{k=1}^{K_A} \beta_k^{(A)} q_k^{(A)}(x)$. We interpret A as a compact, human-readable description of the feature–label relationships supported by real data, within the hypothesis class $\mathcal{H}$.

Step 2: Generate synthetic data with an LLM. We use the generator $\mathcal{G}$ to produce a synthetic dataset D_Q that conforms to the schema $\mathcal{S}$ and constraints $\mathcal{C}$, thereby inducing a synthetic distribution Q over (X, Y). Since our focus is on explainability rather than generator design, $\mathcal{G}$ is treated as a black-box and may be instantiated by any LLM-based tabular data generator.

Step 3: Learn a synthetic explanation (Rule set B) with warm-start from A. Starting from Rule set A (and optionally its coefficients), we continue training the rule learner on the synthetic data D_Q to obtain a converged synthetic explanation: $B = \{q_1^{(B)}, \ldots, q_{K_B}^{(B)}\}$, $f_B(x) = \beta_0^{(B)} + \sum_{k=1}^{K_B} \beta_k^{(B)} q_k^{(B)}(x)$. Intuitively, B captures the best rule-expressible feature–label dependencies induced by the synthetic distribution Q.

5 Rule-based Evaluation Metrics for LLM-Generated Synthetic Data

We assess synthetic data quality using four complementary metrics that together characterize utility transfer, diagnostic drift, and mechanism drift. We also provide theoretical analysis establishing guarantees for these proposed metrics.

5.1 Evaluation Metrics

(i) Cross-domain transfer gap (CDTG; utility fidelity). Let D_P^{test} denote a held-out real test set. We define the empirical cross-domain gap as

$$\widehat{\Delta}_{\text{cross}} := \widehat{R}(f_B; D_P^{\text{test}}) - \widehat{R}(f_A; D_P^{\text{test}}),$$

where $\widehat{R}(f; D) = \frac{1}{|D|} \sum_{(x,y)\in D} \ell(f(x), y)$. A small $\widehat{\Delta}_{\text{cross}}$ indicates that the synthetic-derived explanation transfers well to real data. See Theorem 5.3 for detailed proof.

(ii) Orthogonal Gradient Energy (OGE; diagnostic drift). Let $\Phi_A^{(Q)} \in \{0,1\}^{m \times K_A}$ be the output matrix of queries in A evaluated on the synthetic sample, and let $S_A := \text{range}(\Phi_A^{(Q)}) \subseteq \mathbb{R}^m$ be the induced span. We define the empirical synthetic gradient vector evaluated at f_A:

$$\big[\boldsymbol{g}_Q(f_A)\big]_j = \frac{1}{m} \frac{\partial}{\partial z} \ell(z, y_j^{(Q)})\Big|_{z=f_A(x_j^{(Q)})},$$

and thus derive the OGE as

$$\text{OGE}(Q; A) := \big\|\Pi_{S_A^\perp} \boldsymbol{g}_Q(f_A)\big\|_2,$$

where $\Pi_{S_A^\perp}$ denotes orthogonal projection onto $S_A^\perp$. A large $\text{OGE}(Q; A)$ indicates that the synthetic loss gradient contains systematic components that cannot be removed by reweighting existing real-data rules, implying the need for new explanatory directions. See Theorem 5.2 for detailed proof.

(iii) Rule-level support shifts (RLSS; mechanism drift). We evaluate the real-data explanation f_A on synthetic samples and compute rule-level summaries. For any query q, define its empirical support length on D_Q: $\widehat{p}_Q(q) := \sum_{j=1}^m q(x_j^{(Q)})$, with a small $\varepsilon > 0$ for numerical stability. We compute analogous quantities $\widehat{p}_P(\cdot), \widehat{\mu}_P(\cdot)$ on real validation/test splits and then we compute support shifts:

$$\Delta_{\text{supp}}(q) := \big|\widehat{p}_P(q) - \widehat{p}_Q(q)\big|.$$

These quantities localize *which* dependencies drift and provide actionable diagnostics for refining prompts, constraints, or post-generation filtering.

(iv) Explanation similarity from A→B. (xSim) We summarize distribution shift in a human-auditable form by comparing the two query sets:

$$A \cap B \text{ (preserved)}, A \setminus B \text{ (removed)}, B \setminus A \text{ (added)},$$

and by quantifying coefficient drift on shared rules. We further report a structural similarity score, namely the Jaccard index: $J(A, B) := \frac{|A \cap B|}{|A \cup B|}$. We consider queries that have the same support for the same dataset as identical. For queries which are not exactly identical, we calculate the overlap rate of the support (coverage) of the queries. If the overlap rate of two queries is within a tolerance (e.g., >90%), we say that they are similar.

5.2 Theoretical Analysis

We provide theoretical guarantees that our evaluation captures utility transfer and mechanism drift. After full correction, there is no descent direction within the span of selected rules, so projecting the gradient onto the orthogonal complement targets a new explanatory direction. We prove that a

small OGE(Q;A) means the real-data-derived explanation A is (approximately) stationary under the synthetic distribution, indicating preserved rule-expressible structure; a large OGE(Q;A) signals systematic drift not fixable by reweighting existing rules, requiring new rules. Finally, via cross-domain transfer, if real and synthetic distributions are close in our interpretable hypothesis class, then the best synthetic explanation performs nearly as well as the best real-data explanation on real data.

Define the gradient of the *data-fit term* w.r.t. $\hat{\boldsymbol{f}}$:

$$\boldsymbol{g}^{(t)} := \nabla_{\hat{\boldsymbol{f}}} \left(\frac{1}{n} \sum_{i=1}^{n} \ell(\hat{f}_i, y_i) \right) \Bigg|_{\hat{\boldsymbol{f}}=\hat{\boldsymbol{f}}^{(t)}}.$$

For exponential family using (3), we have $g_i^{(t)} = \frac{1}{n}\Big(A'(\hat{f}_i^{(t)}) - y_i\Big)$.

Theorem 5.1. *Assume $\ell(\cdot, y)$ is differentiable and convex, and let $f^{(t)}$ be obtained by corrective refit (7) with query matrix Φ_t.*

1. *If $\lambda = 0$, then $\forall j \le t$, the gradient is orthogonal to the selected query span:*

$$\Phi_t^\top \boldsymbol{g}^{(t)} = \mathbf{0} \iff (\boldsymbol{q}_j)^\top \boldsymbol{g}^{(t)} = 0. \quad (8)$$

2. *If $\lambda > 0$, then*

$$\Phi_t^\top \boldsymbol{g}^{(t)} + \lambda \boldsymbol{\beta}^{(t)}/n = \mathbf{0}. \quad (9)$$

Equivalently, the gradient is orthogonal up to the regularization term.

See Appendix B for detailed proof. The Theorem 5.1 indicates that after full correction, there is no descent direction within the span of already-selected rules. Thus projecting the gradient onto the orthogonal complement targets genuinely new explanatory directions.

Theorem 5.2. *Assume $\ell(\cdot, y)$ is convex and differentiable, and consider updates to f_A restricted to the real-rule span S_A, i.e., predictors of the form $f_A + \sum_{j=1}^{K} \delta_j q_j$ evaluated on D_Q. Then:*

1. *If $\mathrm{OGE}(Q; A) = 0$, the synthetic gradient $\boldsymbol{g}_Q(f_A)$ lies in S_A, and thus the steepest descent direction of the synthetic empirical risk at f_A can be representable by reweighting rules in A.*

2. *If $\mathrm{OGE}(Q; A) > 0$, the synthetic empirical risk at f_A admits a non-zero gradient component orthogonal to S_A. Consequently, any update using only reweighting of rules in A cannot eliminate this orthogonal component; reducing synthetic risk further requires adding at least one rule whose output vector has a non-trivial component in $S_A^{\perp}$.*

See Appendix B for detailed proof. A small $\mathrm{OGE}(Q; A)$ indicates that the real-data-derived explanation A is close to stationary under the synthetic distribution, suggesting that the synthetic generator preserves the rule-expressible predictive structure of the real data. A large $\mathrm{OGE}(Q; A)$ indicates systematic synthetic drift that cannot be resolved by reweighting existing real-data rules, necessitating new, non-redundant explanatory rules.

Theorem 5.3. *Let $f_P^\star \in \arg\min_{f\in\mathcal{H}} \mathcal{R}_P(f)$ and $f_Q^\star \in \arg\min_{f\in\mathcal{H}} \mathcal{R}_Q(f)$. Define the Hypothesis-class discrepancy as $\mathrm{disc}_{\mathcal{H}}(P, Q) := \sup_{f\in\mathcal{H}} \big|\mathcal{R}_P(f) - \mathcal{R}_Q(f)\big|$. Define the cross-risk gap $\Delta_{\mathrm{cross}} := \mathcal{R}_P(f_Q^\star) - \mathcal{R}_P(f_P^\star)$. Then*

$$\Delta_{\mathrm{cross}} \le 2\,\mathrm{disc}_{\mathcal{H}}(P, Q). \quad (10)$$

See Appendix B for detailed proof. Theorem 5.3 shows that synthetic fidelity can be assessed via cross-domain transfer: if real and synthetic distributions are close with respect to our interpretable hypothesis class, then the optimal synthetic explanation must perform nearly as well as the optimal real-data explanation on real data.

6 Experiments and Results

We evaluate our method on six benchmark tabular datasets (Adult, Abalone, Buddy, California, Diabetes, and German) and compare with five baselines: HARMONIC (Wang et al., 2024), GREAT (Borisov et al., 2022), REAL (training directly on the original data as an oracle upper bound) (Solatorio and Dupriez, 2023), CTGAN (Xu et al., 2019), and TVAE (Xu et al., 2019). Adult, Abalone, Diabetes, and German credit are obtained from the UCI Machine Learning Repository (Asuncion and Newman, 2007), while California Housing is sourced from the scikit-learn real-world datasets (Pedregosa et al., 2011). The first three (HARMONIC, GREAT, REAL) are LLM-based settings, while CTGAN and TVAE are representative GAN/VAE-style tabular synthesizers. For each dataset and synthesis method, we evaluate synthetic fidelity using the four metrics defined earlier: *(i)* the cross-domain transfer gap, which measures whether explanations learned

Table 1: Results on 6 datasets and 5 methods across four evaluation metrics. Each subtable reports one metric.

(a) CDTG ($\rightarrow 0$)

Method	Adult	Abalone	Buddy	California	Diabetes	German
Harmonic	-0.051	2.733	-0.042	0.765	-0.084	-0.055
Great	-0.008	4.913	-0.017	0.354	-0.214	-0.085
Real	-0.024	0.510	-0.015	0.041	-0.052	-0.030
TVAE	-0.017	2.473	0.005	0.183	0.026	-0.020
CTGAN	-0.031	2.319	-0.012	1.080	-0.117	-0.140

(b) OGE ($\downarrow$)

Method	Adult	Abalone	Buddy	California	Diabetes	German
Harmonic	4.337	169.1	8.260	52.43	6.227	6.308
Great	11.90	368.6	8.933	222.8	7.318	7.382
Real	2.550	140.5	1.290	38.42	5.069	4.349
TVAE	14.13	176.9	11.97	276.7	4.938	4.184
CTGAN	2.260	169.0	5.575	55.56	6.732	8.321

(c) RLSS ($\downarrow$)

Method	Adult	Abalone	Buddy	California	Diabetes	German
Harmonic	2.302	3.572	2.214	1.654	1.741	1.781
Great	2.779	2.494	0.505	2.752	1.971	2.769
Real	1.914	2.683	0.471	0.700	1.595	2.323
TVAE	2.249	1.678	0.655	3.521	1.733	2.649
CTGAN	1.937	2.059	1.391	1.750	1.285	2.286

(d) xSim ($\uparrow$)

Method	Adult	Abalone	Buddy	California	Diabetes	German
Harmonic	0.152	0.081	0.290	0.053	0.081	0.026
Great	0.462	0.081	0.818	0.290	0.053	0.081
Real	0.250	0.176	0.667	0.600	0.143	0.111
TVAE	0.600	0.111	0.739	0.176	0.111	0.081
CTGAN	0.143	0.111	0.739	0.250	0.081	0.026

from synthetic data generalize to held-out real data, *(ii)* orthogonal gradient energy (OGE) which detects mechanism drift that cannot be corrected by reweighting real-data rules, *(iii)* rule-level support which quantify how frequently each dependency appears and how strongly it relates to the label, and *(iv)* an explicit explanation similarity from Rule set A (real) to Rule set B (synthetic).

6.1 Evaluation Results

As shown in Table 1, across datasets, the metrics reveal a consistent fidelity–instability pattern: GReaT often incurs the largest OGE, indicating substantial mechanism drift that cannot be corrected by reweighting real rules, even when its transfer gap is competitive on some datasets (e.g., Diabetes). In contrast, REAL achieves strong performance on the drift/similarity-based metrics (OGE, RLSS, xSim), serving as a reliable reference for explanation stability. Traditional generators (CTGAN/TVAE) remain strong baselines and frequently achieve the best or near-best results on specific datasets—particularly on Adult (TVAE best xSim; CTGAN best OGE) and Diabetes/German (TVAE best OGE)—highlighting that synthetic fidelity is dataset-dependent and reinforcing the need to evaluate both cross-domain transfer and explanation stability.

6.2 From Scores to Explanations: Auditing Synthetic Data

These metrics are informative, but their raw values should not be interpreted in isolation. Strong utility or transfer does not guarantee "realism": a generator may match performance for a particular predictor family while still shifting marginals, correlations, tail behavior, or rare subgroups. This risk is amplified for LLM-based tabular generators, which can produce plausible outputs while subtly altering the data-generating mechanism. Hence, we complement scalar scores with interpretability-based analysis, inspecting explanations to identify which dependencies are preserved versus distorted. To aid interpretation, we visualize the learned rules

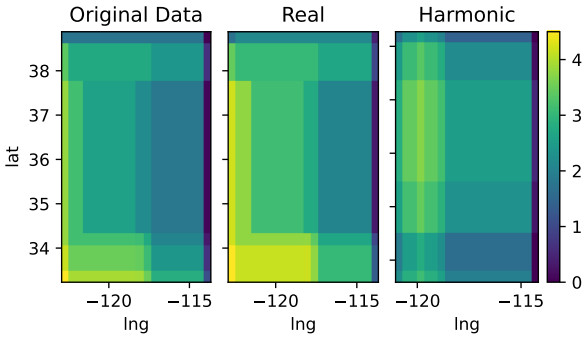


Figure 2: The outputs of the additive rule ensembles trained on the original California dataset and the data synthesized by Real and HARMONIC.

over the California dataset by mapping their longitude–latitude coverage onto heatmaps for three rule sets in Figure 2: one learned from the original data, one learned from the RealTabFormer synthetic data, and one learned from the HARMONIC synthetic data. The longitude–latitude heatmaps indicate that the rule set learned from the original data closely matches that learned from RealTabFormer, while the rule set learned from HARMONIC differs substantially. We further visualize the differences between the policies learned from the original data and those learned from the RealTabFormer and HARMONIC datasets in Figure 3. In the difference heatmaps, deeper red and blue colors indicate larger

deviations from the original policy, with red denoting positive differences and blue denoting negative differences. For latitude $< 34°$, the HARMONIC difference map is predominantly blue, indicating a negative deviation from the original policy. A

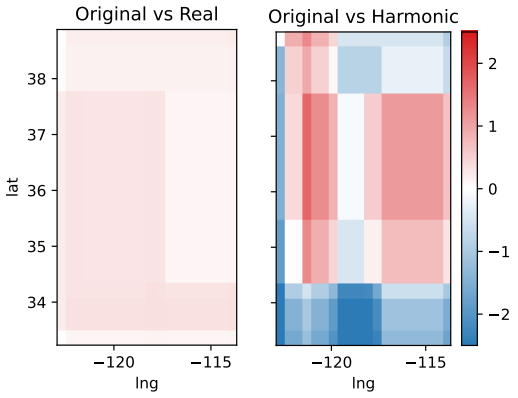


Figure 3: Difference between the outputs of the additive rule ensembles trained on the original California dataset and the data synthesized by Real and HARMONIC.

natural question arising from the observed discrepancies is whether the synthetic dataset generated by HARMONIC is *incorrect*. To answer this question, we leverage our rule-based analysis, which enables a fine-grained inspection of how explanatory mechanisms differ across datasets.

In the ruleset learned from the original data, latitude around $34°$ emerges as a stable and interpretable geographic signal through multiple positive rules. Specifically, there are three rules associate with that latitude:

1. $+0.4285$ if $\texttt{households} \leq 3.4194 \wedge \texttt{latitude} \leq 33.62 \wedge \texttt{median\_income} \geq 1.9063 \wedge \texttt{total\_rooms} \geq 3.8044$. This rule captures low-density, middle-income Southern California neighborhoods with sufficient housing capacity. Such regions are typically suburban or coastal residential areas, where location provides a strong baseline advantage that is reinforced by adequate income and housing size;

2. $+0.3753$ if $\texttt{households} \leq 3.8783 \wedge \texttt{latitude} \leq 34.26 \wedge -118.5 \leq \texttt{longitude} \leq -117.88$ This rule highlights the Greater Los Angeles and nearby metropolitan corridor, where geographic location alone acts as a dominant explanatory factor. Low household density within this high-demand region is strongly associated with positive outcomes, even without explicit income constraints;

3. $+0.3399$ if $\texttt{households} \leq 2.5062 \wedge \texttt{housing\_median\_age} \geq 25 \wedge \texttt{latitude} \leq 34.1 \wedge \texttt{total\_bedrooms} \geq 0.9663$.

This rule identifies established Southern California neighborhoods with older housing stock and very low household density. Such areas often correspond to mature, well-developed residential zones that benefit from long-term location premiums.

Together, these rules encode a coherent and interpretable mechanism in which Southern California acts as a globally positive geographic regime, with household density, income, and housing characteristics serving as moderating factors.

In contrast to the original dataset, the ruleset learned from the HARMONIC synthetic data contains a *negative* rule that overlaps substantially with the same latitude range: -0.2492 if $\texttt{households} \geq 1.9196 \wedge \texttt{latitude} \leq 35.362 \wedge 550 \leq \texttt{population} \leq 2158.3 \wedge \texttt{total\_bedrooms} \geq 1.0165$.

This rule penalizes moderately dense, mid-population Southern California regions with relatively high bedroom counts. Unlike the original rules, latitude no longer provides a positive baseline; instead, it appears only within a narrow, population-conditioned penalty. This inversion of the geographic signal explains the systematic negative deviations observed for $\texttt{latitude} < 34°$ in the HARMONIC difference heatmap.

From an evaluation metrics-based perspective, the HARMONIC synthetic data do not exhibit a well-aligned spatial distribution on the California dataset, a closer inspection of the learned rules reveals that the generated data are not unreasonable. In particular, HARMONIC preserves local statistical dependencies and predictive utility, but fails to retain the global explanatory mechanisms present in the original data. As a result, passing predictive evaluations alone does not guarantee explanation fidelity, underscoring the importance of mechanism-aware analysis when assessing synthetic datasets.

7 Discussion and Future Work

In this work, we introduce a mechanism-aware framework that uses interpretable explanations as the unit of comparison, enabling us to separate apparent utility transfer from genuine fidelity of underlying dependencies. By analyzing how explanations trained on real data change under synthetic distributions, our metrics expose when performance is preserved by reweighting familiar patterns versus when the generator induces drift that requires genuinely new explanatory structure. Em-

pirically, we show that this explanation-centric view turns evaluation into diagnosis: it not only flags failures based on score-based tests, but also localizes what changed and why.

More generally, we ask which mechanisms the generator relies on, and how those mechanisms shift across domains, prompts, or data sources. Our results motivate evaluation pipelines that treat explanations—rules in a human-auditable hypothesis class—as first-class objects. This makes drift detectable, comparable, and actionable, providing a practical bridge between black-box performance and trustworthy deployment. Future work can expand beyond tabular, and integrate risk-sensitive auditing so that explanation fidelity becomes a standard counterpart to utility in LLM evaluation.

Limitations

Our analysis should not be interpreted as validating any specific generator (e.g., HARMONIC) as producing "correct" or fully realistic data. The proposed framework is an auditing tool, not a certification mechanism: it identifies and explains distributional shifts through learned rules. Also, the proposed framework targets statistical/mechanistic fidelity, not downstream social harms (privacy leakage, memorization, fairness, or representation bias), which require additional evaluation dimensions.

Also, our proposed metrics are intended to quantify how closely the generated data matches the original data, and the human interpretability of the extracted rules provides an additional lens to inspect why and how the generated data differs. We do not explicitly link these two via a meta-evaluation metric; In many test cases, the learned rule sets are highly dataset-specific, and meaningful human judgment would require domain expertise (e.g., for the diabetes dataset). Identifying a single meta-evaluation criterion can be challenging. We therefore position a large-scale human meta-evaluation as future work. That said, the interpretability component of our method provides a practical mechanism for sanity-checking and debugging: when the quantitative scores suggest a potential mismatch, the corresponding rule-level explanations allow users/domain experts to inspect where the mismatch arises and whether it reflects genuine mechanism drift, spurious dependencies, or other generator artifacts.

References

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31.

Arthur Asuncion and David Newman. 2007. Uci machine learning repository.

Clément Bénard, Gérard Biau, Sébastien Da Veiga, and Erwan Scornet. 2021. Sirus: Stable and interpretable rule set for classification.

Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. 2022. Language models are realistic tabular data generators. *arXiv preprint arXiv:2210.06280*.

William W Cohen. 1995. Fast effective rule induction. In *Proceedings of the twelfth international conference on machine learning*, pages 115–123.

Joao Fonseca and Fernando Bacao. 2023. Tabular and latent space synthetic data generation: a literature review. *Journal of Big Data*, 10(1):115.

Jerome H Friedman and Bogdan E Popescu. 2008. Predictive learning via rule ensembles.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, 50(3):1097–1179.

Mauro Giuffrè and Dennis L Shung. 2023. Harnessing the power of synthetic data in healthcare: innovation, application, and privacy. *NPJ digital medicine*, 6(1):186.

Mikel Hernandez, Pablo A Osorio-Marulanda, Mikel Catalina, Lorea Loinaz, Gorka Epelde, and Naiara Aginako. 2025. Comprehensive evaluation framework for synthetic tabular data in health: fidelity, utility and privacy analysis of generative models with and without privacy guarantees. *Frontiers in Digital Health*, 7:1576290.

Jan Kapar, Niklas Koenen, and Martin Jullum. 2025. What's wrong with your synthetic tabular data? using explainable ai to evaluate generative models. In *World Conference on Explainable Artificial Intelligence*, pages 19–43. Springer.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. 2023. Tabddpm: Modelling tabular data with diffusion models. In *International conference on machine learning*, pages 17564–17579. PMLR.

Sean Kulinski and David I Inouye. 2023. Towards explaining distribution shifts. In *International Conference on Machine Learning*, pages 17931–17952. PMLR.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2019. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1096.

Anton D Lautrup, Tobias Hyrup, Arthur Zimek, and Peter Schneider-Kamp. 2025. Syntheval: a framework for detailed utility and privacy evaluation of tabular synthetic data. *Data Mining and Knowledge Discovery*, 39(1):6.

David Lopez-Paz and Maxime Oquab. 2016. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Peter McCullagh. 2019. *Generalized linear models*. Routledge.

Marko Miletic and Murat Sariyar. 2024. Challenges of using synthetic data generation methods for tabular microdata. *Applied Sciences*, 14(14):5975.

Carlos Mougan, Klaus Broelemann, David Masip, Gjergji Kasneci, Thanassis Thiropanis, and Steffen Staab. 2023. Explanation shift: How did the distribution shift impact the model? *arXiv preprint arXiv:2303.08081*.

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

Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. 2016. The synthetic data vault. In *2016 IEEE international conference on data science and advanced analytics (DSAA)*, pages 399–410. IEEE.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and 1 others. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. 2019. Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems*, 32.

Pol G Recasens, Alberto Gutierrez, Jordi Torres, Josep Berral, Anisa Halimi, Kieran Fraser, and 1 others. 2025. In-context bias propagation in llm-based tabular data generation. *arXiv preprint arXiv:2506.09630*.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215.

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

Aivin V Solatorio and Olivier Dupriez. 2023. Realtabformer: Generating realistic relational and tabular data using transformers. *arXiv preprint arXiv:2302.02041*.

Marieke Stolte, Franziska Kappenberg, Jörg Rahnenführer, and Andrea Bommert. 2024. Methods for quantifying dataset similarity: a review, taxonomy and comparison. *Statistic Surveys*, 18:163–298.

Yuxin Wang, Duanyu Feng, Yongfu Dai, Zhengyu Chen, Jimin Huang, Sophia Ananiadou, Qianqian Xie, and Hao Wang. 2024. Harmonic: Harnessing llms for tabular data synthesis and privacy protection. *Advances in Neural Information Processing Systems*, 37:100196–100212.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32.

Fan Yang, Pierre Le Bodic, Michael Kamp, and Mario Boley. 2024. Orthogonal gradient boosting for simpler additive rule ensembles. In *International Conference on Artificial Intelligence and Statistics*, pages 1117–1125. PMLR.

Yue Yang, Fan Yang, Yu Bai, and Hao Wang. 2025. Self-interpretable reinforcement learning via rule ensembles. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '25, page 2235–2243, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Yue Yang, Fan Yang, Yu Bai, and Hao Wang. 2026. Neural+symbolic approaches for interpretable actor-critic reinforcement learning. In *The Fourteenth International Conference on Learning Representations*.

Zexi Yao, Nataša Krčo, Georgi Ganev, and Yves-Alexandre de Montjoye. 2025. The dcr delusion: Measuring the privacy risk of synthetic data. In *European Symposium on Research in Computer Security*, pages 469–487. Springer.

Hengrui Zhang, Jiani Zhang, Balasubramaniam Srinivasan, Zhengyuan Shen, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. 2023. Mixed-type tabular data synthesis with score-based diffusion in latent space. *arXiv preprint arXiv:2310.09656*.

Jun Zhang, Graham Cormode, Cecilia M Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2017. Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41.

Zilong Zhao, Robert Birke, and Lydia Y Chen. 2025. Tabula: Harnessing language models for tabular data synthesis. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 247–259. Springer.

A Additional Theoretical Preliminaries

Remark (Multi-class exponential family)**.** *For K-class classification, let $f(x) \in \mathbb{R}^K$ be logits and use multinomial NLL: $\ell(f, y) = \log \sum_{k=1}^{K} e^{f_k} - f_y$. This is an exponential-family NLL with vector natural parameter. All monotonicity and cross-gap results remain unchanged. The orthogonality condition generalizes to $Q_t^\top G = 0$ where $G \in \mathbb{R}^{n \times K}$ stacks gradients per class.*

Lemma A.1. *For the canonical exponential family loss in Eq. (2), let $z = f(x)$, then:*

1. *$\ell(z, y)$ is convex in z since $A(z)$ is convex.*
2. *The gradient and Hessian are*

$$\begin{aligned} \frac{\partial}{\partial z}\ell(z, y) &= A'(z) - y, \\ \frac{\partial^2}{\partial z^2}\ell(z, y) &= A''(z) \geq 0. \end{aligned} \tag{11}$$

3. *$A'(z) = \mathbb{E}[Y \mid \theta = z]$ and $A''(z) = \mathrm{Var}(Y \mid \theta = z)$ for regular families.*

Proof. (1) is standard: A is convex as a log-partition function, hence $A(z) - yz$ is convex in z. (2) follows by differentiation of Equation (2). (3) is a standard property of exponential families: derivatives of A correspond to cumulants. □

OGB Objective and a Useful Upper Bound: Let $Q_{t-1} = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{t-1}]$ and denote the orthogonal decomposition

$$\begin{aligned} \boldsymbol{q} = \boldsymbol{q}_\parallel + \boldsymbol{q}_\perp, \quad \boldsymbol{q}_\parallel &\in \mathrm{range}(Q_{t-1}), \\ \boldsymbol{q}_\perp &\perp \mathrm{range}(Q_{t-1}). \end{aligned}$$

Similarly, let $\boldsymbol{g} = \boldsymbol{g}_\parallel + \boldsymbol{g}_\perp$ be the decomposition of the gradient with respect to $\mathrm{range}(Q_{t-1})$. The OGB objective (with stabilizer $\varepsilon > 0$) is

$$\mathrm{obj}_{\mathrm{ogb}}(\boldsymbol{q}) \;:=\; \frac{\left|\boldsymbol{g}_\perp^\top \boldsymbol{q}_\perp\right|}{\|\boldsymbol{q}_\perp\|_2 + \varepsilon}. \tag{12}$$

Proposition 1 (Upper bound on the OGB objective)**.** *For any candidate $\boldsymbol{q}$,*

$$\mathrm{obj}_{\mathrm{ogb}}(\boldsymbol{q}) \;\leq\; \|\boldsymbol{g}_\perp\|_2.$$

Proof. Since $\boldsymbol{q}_\perp \perp \mathrm{range}(Q_{t-1})$ and $\boldsymbol{g}_\parallel \in \mathrm{range}(Q_{t-1})$, we have $\boldsymbol{g}^\top \boldsymbol{q}_\perp = (\boldsymbol{g}_\perp)^\top \boldsymbol{q}_\perp$. By Cauchy–Schwarz,

$$|\boldsymbol{g}^\top \boldsymbol{q}_\perp| = |(\boldsymbol{g}_\perp)^\top \boldsymbol{q}_\perp| \leq \|\boldsymbol{g}_\perp\|_2 \, \|\boldsymbol{q}_\perp\|_2.$$

Divide by $\|\boldsymbol{q}_\perp\|_2 + \varepsilon \geq \|\boldsymbol{q}_\perp\|_2$ to obtain

$$\mathrm{obj}_{\mathrm{ogb}}(\boldsymbol{q}) = \frac{|\boldsymbol{g}^\top \boldsymbol{q}_\perp|}{\|\boldsymbol{q}_\perp\|_2 + \varepsilon} \leq \frac{\|\boldsymbol{g}_\perp\|_2 \|\boldsymbol{q}_\perp\|_2}{\|\boldsymbol{q}_\perp\|_2 + \varepsilon} \leq \|\boldsymbol{g}_\perp\|_2.$$

□

Theorem A.2. *Let $\{f^{(t)}\}$ be the sequence produced by repeatedly adding rules (any selection strategy) and performing corrective refit (7), we have:*

1. *$R_\lambda(f^{(t+1)}) \leq R_\lambda(f^{(t)})$ for all t.*
2. *If $R_\lambda(f) \geq \underline{R} > -\infty$ for all f in the considered class (true for exponential-family NLL), then $\{R_\lambda(f^{(t)})\}$ converges.*

Proof. Let $\mathcal{F}_t := \{\beta_0 + \sum_{j=1}^{t} \beta_j q_j(\cdot)\}$ be the function class spanned by selected rules at iteration t. Then $\mathcal{F}_t \subseteq \mathcal{F}_{t+1}$. By definition of corrective refit,

$$R_\lambda(f^{(t)}) = \min_{f \in \mathcal{F}_t} R_\lambda(f),$$
$$R_\lambda(f^{(t+1)}) = \min_{f \in \mathcal{F}_{t+1}} R_\lambda(f).$$

Since the minimization is over a superset at $t+1$, the minimum cannot increase: $R_\lambda(f^{(t+1)}) \leq R_\lambda(f^{(t)})$. For (2), monotone non-increasing sequences bounded below converge. For exponential-family NLL, $A(z) - yz$ is lower bounded for regular families on $\mathbb{R}$ (and with ℓ_2 regularization R_λ is coercive), hence bounded below. □

B Proof for the Theorem

Theorem B.1. *Assume $\ell(\cdot, y)$ is differentiable and convex. Let $f^{(t)}$ be obtained by corrective refit (7) with rule matrix Q_t.*

1. *If $\lambda = 0$, then the gradient is orthogonal to the selected rule span:*

$$Q_t^\top \boldsymbol{g}^{(t)} = \mathbf{0} \iff (\boldsymbol{q}_j)^\top \boldsymbol{g}^{(t)} = 0, \ \forall j \leq t. \tag{13}$$

2. *If $\lambda > 0$, then*

$$Q_t^\top \boldsymbol{g}^{(t)} + \frac{\lambda}{n} \boldsymbol{\beta}^{(t)} = \mathbf{0}. \tag{14}$$

Equivalently, the gradient is orthogonal up to the regularization term.

Proof. Write $\hat{\boldsymbol{f}} = \beta_0 \mathbf{1} + Q_t \boldsymbol{\beta}$ and consider the objective in $(\beta_0, \boldsymbol{\beta})$:

$$\Phi(\beta_0, \boldsymbol{\beta}) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(\beta_0 + (Q_t \boldsymbol{\beta})_i, y_i\big) + \frac{\lambda}{2n} \|\boldsymbol{\beta}\|_2^2.$$

Since ℓ is differentiable, stationarity at the (global) minimizer gives

$$\nabla_{\boldsymbol{\beta}} \Phi(\beta_0^{(t)}, \boldsymbol{\beta}^{(t)}) = \mathbf{0}.$$

By chain rule,

$$\nabla_{\boldsymbol{\beta}} \left(\frac{1}{n} \sum_{i=1}^{n} \ell(\hat{f}_i, y_i) \right) = Q_t^\top \boldsymbol{g}^{(t)}.$$

Also, $\nabla_{\boldsymbol{\beta}} \left(\frac{\lambda}{2n} \|\boldsymbol{\beta}\|_2^2 \right) = \frac{\lambda}{n} \boldsymbol{\beta}^{(t)}$. Hence

$$Q_t^\top \boldsymbol{g}^{(t)} + \frac{\lambda}{n} \boldsymbol{\beta}^{(t)} = \mathbf{0}.$$

If $\lambda = 0$ we recover (13). This holds for any differentiable convex loss, and in particular for the exponential-family NLL (2). □

Theorem B.2. *Assume $\ell(\cdot, y)$ is convex and differentiable, and consider updates to f_A restricted to the real-rule span S_A, i.e., predictors of the form $f_A + \sum_{j=1}^{K} \delta_j q_j$ evaluated on D_Q. Then:*

1. *If $\mathrm{OGE}(Q; A) = 0$, the synthetic gradient $\boldsymbol{g}_Q(f_A)$ lies in S_A, and thus the steepest descent direction of the synthetic empirical risk at f_A is representable by reweighting rules in A.*

2. *If* $\mathrm{OGE}(Q;A) > 0$*, the synthetic empirical risk at* f_A *has a non-zero gradient component orthogonal to* S_A*. Consequently, any update using only reweighting of rules in* A *cannot eliminate this orthogonal component; reducing synthetic risk further requires adding at least one rule whose output vector has a non-trivial component in* $S_A^{\perp}$*.*

Proof. We first verify that the gradient is well-defined under the losses of interest. For canonical exponential-family NLL $\ell(z,y) = A(z) - yz$, Lemma 3.1 states that $\ell(\cdot, y)$ is convex and differentiable with $\partial_z \ell(z,y) = A'(z) - y$, hence $\boldsymbol{g}_Q(f_A)$ exists and is finite.

Let $\widehat{\mathcal{R}}_Q(f) = \frac{1}{m}\sum_{i=1}^{m} \ell(f(x_i^{(Q)}), y_i^{(Q)})$. For any perturbation that *only reweights* the existing rules in A, $\Delta f(x) = \sum_{j=1}^{K} \delta_j q_j(x)$, the induced prediction-vector change on D_Q is $\Delta\hat{\boldsymbol{f}} = Q_A^{(Q)}\boldsymbol{\delta} \in S_A$ by definition of S_A. By differentiability, the first-order expansion gives $\widehat{\mathcal{R}}_Q(f_A + \Delta f) = \widehat{\mathcal{R}}_Q(f_A) + \langle \boldsymbol{g}_Q(f_A), \Delta\hat{\boldsymbol{f}}\rangle + o(\|\Delta\hat{\boldsymbol{f}}\|_2)$.

Decompose $\boldsymbol{g}_Q(f_A) = \Pi_{S_A}\boldsymbol{g}_Q(f_A) + \Pi_{S_A^{\perp}}\boldsymbol{g}_Q(f_A)$. If $\mathrm{OGE}(Q;A) = 0$, then $\Pi_{S_A^{\perp}}\boldsymbol{g}_Q(f_A) = \mathbf{0}$, so $\boldsymbol{g}_Q(f_A) \in S_A$ and the negative gradient direction $-\boldsymbol{g}_Q(f_A)$ is representable by some $\Delta\hat{\boldsymbol{f}} \in S_A$, proving (1).

If $\mathrm{OGE}(Q;A) > 0$, then $\Pi_{S_A^{\perp}}\boldsymbol{g}_Q(f_A) \neq \mathbf{0}$. For any $\Delta\hat{\boldsymbol{f}} \in S_A$, orthogonality implies $\langle \Pi_{S_A^{\perp}}\boldsymbol{g}_Q(f_A), \Delta\hat{\boldsymbol{f}}\rangle = 0$, so the directional derivative within the span S_A depends only on $\Pi_{S_A}\boldsymbol{g}_Q(f_A)$. Thus updates that only reweight rules in A cannot affect the orthogonal component $\Pi_{S_A^{\perp}}\boldsymbol{g}_Q(f_A)$. In particular, after performing a *full corrective refit* restricted to the rule span S_A, Theorem B.1 implies stationarity within that span (no descent direction remains in S_A), yet the non-zero orthogonal component persists. Therefore, reducing synthetic risk further requires expanding the span by adding at least one rule with non-trivial projection onto $S_A^{\perp}$, proving (2). □

Theorem B.3 (Cross-risk gap bound (any loss))**.** *Define the cross-risk gap*

$$\Delta_{\mathrm{cross}} := \mathcal{R}_P(f_Q^\star) - \mathcal{R}_P(f_P^\star).$$

Then

$$\Delta_{\mathrm{cross}} \le 2\,\mathrm{disc}_{\mathcal{H}}(P,Q). \tag{15}$$

Proof. Add and subtract $\mathcal{R}_Q(\cdot)$:

$$\begin{aligned}\Delta_{\mathrm{cross}} &= \mathcal{R}_P(f_Q^\star) - \mathcal{R}_P(f_P^\star)\\ &= \big(\mathcal{R}_P(f_Q^\star) - \mathcal{R}_Q(f_Q^\star)\big)\\ &\quad + \big(\mathcal{R}_Q(f_Q^\star) - \mathcal{R}_Q(f_P^\star)\big)\\ &\quad + \big(\mathcal{R}_Q(f_P^\star) - \mathcal{R}_P(f_P^\star)\big).\end{aligned}$$

The middle term is ≤ 0 since $f_Q^\star$ minimizes $\mathcal{R}_Q$ over $\mathcal{H}$. Thus

$$\begin{aligned}\Delta_{\mathrm{cross}} &\le \big|\mathcal{R}_P(f_Q^\star - \mathcal{R}_Q(f_Q^\star)\big| +\\ &\qquad \big|\mathcal{R}_Q(f_P^\star) - \mathcal{R}_P(f_P^\star)\big|\\ &\le 2\,\mathrm{disc}_{\mathcal{H}}(P,Q),\end{aligned}$$

because both $f_Q^\star$ and $f_P^\star$ belong to $\mathcal{H}$. □

C Prompt for LLM-based Method

HARMONIC prompt (California Housing)[th] **k-shot JSON completion (template, $k = 5$).** Provide 5 real rows (JSON dictionaries; feature order may be shuffled), then request one additional approximate sample in the same JSON format.

```
Here are 5 tabular data about California
    Housing, each containing 8 feature
    columns
and 1 label column (MedHouseVal). I will
    transmit the data to you in JSON format.
Please generate an approximate sample based
    on these 5 examples.

Example one: {"MedInc": \ph{...},
    "HouseAge": \ph{...}, "AveRooms":
    \ph{...}, "AveBedrms": \ph{...},
            "Population": \ph{...},
                "AveOccup": \ph{...},
                "Latitude": \ph{...},
                "Longitude": \ph{...},
            "MedHouseVal": \ph{...}}

Example two: {"AveOccup": \ph{...},
    "Longitude": \ph{...}, "Population":
    \ph{...}, "MedInc": \ph{...},
            "AveRooms": \ph{...},
                "Latitude": \ph{...},
                "HouseAge": \ph{...},
                "AveBedrms": \ph{...},
            "MedHouseVal": \ph{...}}

Example three:{"HouseAge": \ph{...},
    "MedHouseVal": \ph{...}, "Latitude":
    \ph{...}, "MedInc": \ph{...},
            "AveBedrms": \ph{...},
                "AveRooms": \ph{...},
                "AveOccup": \ph{...},
                "Population": \ph{...},
            "Longitude": \ph{...}}

Example four: {"Latitude": \ph{...},
    "AveRooms": \ph{...}, "AveBedrms":
    \ph{...}, "HouseAge": \ph{...},
            "MedInc": \ph{...}, "AveOccup":
                \ph{...}, "Longitude":
                \ph{...}, "Population":
                \ph{...},
            "MedHouseVal": \ph{...}}

Example five: {"Longitude": \ph{...},
    "Latitude": \ph{...}, "MedInc":
    \ph{...}, "HouseAge": \ph{...},
            "AveRooms": \ph{...},
                "AveBedrms": \ph{...},
                "Population": \ph{...},
                "AveOccup": \ph{...},
            "MedHouseVal": \ph{...}}

Generate one sample:
```

Optional (fine-tuning variant): append a reference line `OUTPUT: {...}` as the target completion.

GReaT prompt (California Housing)[th] **Row serialisation (template).** Each record is written as a single text sequence with clauses of the form `<feature> is <value>`. The target can be included as another clause.

```
MedInc is \ph{medinc}, HouseAge is
    \ph{house_age}, AveRooms is
    \ph{ave_rooms},
AveBedrms is \ph{ave_bedrms}, Population is
    \ph{population}, AveOccup is
    \ph{ave_occup},
Latitude is \ph{latitude}, Longitude is
    \ph{longitude}, MedHouseVal is
    \ph{med_house_val}.
```

Conditional generation (prefix). Provide any subset of clauses; the model completes the rest.

```
Latitude is 37.88, Longitude is -122.23,
    MedInc is 8.3252,
```

D "Why" Rules

Why rule-based explanations? We adopt a rule-based explanation layer because it provides *actionable* and *comparable* diagnostics for synthetic tabular data. Unlike distributional similarity scores (e.g., two-sample tests) that only indicate whether a shift exists, learned rules explicitly describe *which* feature–label relationships are present. This enables a direct A→B "explanation diff": preserved rules reveal dependencies the generator maintains, removed rules highlight real-world structure that is lost, and newly added rules expose synthetic artifacts (e.g., shortcut correlations or leakage) that can mislead downstream models. Moreover, rules are human-auditable artifacts that domain experts can inspect and validate, making the evaluation transparent rather than purely metric-driven. Finally, our rule lens is generator-agnostic (it treats the LLM synthesizer as a black box) and complements traditional metrics: we can pair global detection measures with rule-level mechanism drift to both *detect* and *explain* synthetic mismatch.

- **Explainable auditing of LLM-generated tabular data.** We introduce an explanation-based evaluation framework that compares rule explanations learned from real and synthetic data, yielding a human-auditable A→B diff of preserved, missing, and spurious dependencies.
- **Mechanism-level diagnostics beyond scalar scores.** Our rule diff localizes synthetic drift at the level of feature–label relationships, providing actionable insights for prompt/constraint refinement and post-generation filtering.
- **Generator-agnostic evaluation.** The proposed auditing layer applies to any LLM-based tabular synthesis method, decoupling explainability analysis from generator design choices.
- **Theoretical grounding within an interpretable hypothesis class.** We provide a formal justification linking cross-domain performance gaps to distribution mismatch measured through the rule hypothesis class, supporting explanation transfer as a principled synthetic-quality criterion.

E Mechanism-Aware Auditing of LLM-Generated Data

Although our experiments focus on LLM-generated *tabular* datasets, the core problem we address is central to the ACL community: *how to evaluate and explain artifacts produced by large language models beyond predictive utility*. In modern NLP pipelines, LLMs are increasingly used to (i) generate training data (instruction tuning, self-training, data augmentation), (ii) produce structured outputs (information extraction, dialogue states, semantic parses), and (iii) synthesize datasets for privacy, cost, or coverage reasons. In all these settings, "passing" downstream metrics does not guarantee that the generated data preserve the intended linguistic or causal mechanisms; it may instead encode prompt-sensitive shortcuts, spurious correlations, or distributional artifacts that harm robustness, fairness, and scientific validity.

A recurring ACL concern is that standard evaluation can mask *why* a model succeeds or fails, especially under distribution shift. Our framework elevates *explanations as first-class objects*: we compare a compact, human-auditable rule set learned from real data (A) with the corresponding rule set induced by synthetic data (B), and we quantify their *explanation diff*. This aligns with ACL's broader agenda in interpretability and responsible NLP: diagnosing whether a system is learning the *right* generalizations, not merely achieving high scores.

Orthogonal Gradient Energy (OGE) provides a mechanism-aware test for whether synthetic artifacts can be "repaired" by reweighting explanations already supported by real data, or whether they require *new* explanatory structure. This directly mirrors common NLP failure modes: an LLM-generated dataset may preserve surface-level utility while shifting the underlying decision rationale (e.g., relying on style markers, demographic proxies, or prompt artifacts). By construction, OGE separates "fixable by reweighting" drift from "irreducible mechanism drift", which is precisely the kind of diagnostic signal that complements standard leaderboard-style evaluation.

Applicability to NLP tasks. Our method is not tied to any particular modality. It applies whenever we can define: (i) a differentiable loss on labeled examples, and (ii) a feature representation that supports human inspection. For NLP, this includes settings such as:

- Data augmentation / self-instruct: auditing whether synthetic instruction–response pairs introduce unintended heuristics or bias.
- Information extraction & semantic parsing: auditing whether generated structured labels preserve the intended feature–label dependencies (e.g., entity-type cues, syntactic triggers).
- Dialogue state tracking / slot filling: checking whether synthetic dialogues preserve causal relationships between user intents, slots, and context.
- Evaluation set generation: verifying that synthetic test items do not drift towards trivial cues that inflate measured performance.

The community is increasingly emphasizing *faithfulness, robustness, and accountability* in LLM-based systems. Our contribution is an evaluation-and-explanation pipeline that provides (a) quantitative metrics for mechanism drift and (b) qualitative, human-auditable rule-level evidence of what changed. In this sense, our work is not only about synthetic tabular data; it is about a general methodology for auditing LLM-generated datasets and structured outputs when interpretability and distribution shift are first-order concerns.

F Additional Evaluation

We also evaluate synthetic data effectiveness from the traditional perspectives: how well the synthetic samples match the statistical properties of the real data. We report data mismatch (DM) to check datatype compatibility (DM =0 indicates no datatype mismatch), Wasserstein distance (WD) to quantify distributional differences across columns (WD =0 indicates identical distributions), and Correlation Similarity (CS) to measure agreement in column-wise pairwise correlations (CS =1 indicates identical correlations). These metrics provide standard, preliminary sanity checks widely used in prior synthetic tabular data work.

Across four datasets, RTF provides the most reliable synthetic fidelity (lowest WD consistently) while maintaining strong likelihood-based utility, whereas GReaT exhibits severe distributional failures on AD and BU (WD explosion), highlighting the need for robust auditing beyond task-level scores.

Table 2: Effectiveness results on four datasets across five evaluation metrics (mean±std over 3 seeds). We report only the five baselines HARMONIC, TVAE, CTGAN, REAL, and GREAT.

(a) DM (↓)

Method	German	Adult	Diabetes	Buddy
HARMONIC	0.14±0.00	0.00±0.00	0.07±0.05	0.00±0.00
TVAE	0.14±0.00	0.00±0.00	0.10±0.00	0.00±0.00
CTGAN	0.14±0.00	0.00±0.00	0.07±0.05	0.00±0.00
REAL	0.00±0.00	0.00±0.00	0.10±0.00	0.00±0.00
GReaT	0.14±0.00	0.06±0.00	0.07±0.05	0.03±0.04

(b) WD (↓)

Method	German	Adult	Diabetes	Buddy
HARMONIC	0.82±0.03	0.49±0.01	0.07±0.00	0.23±0.02
TVAE	0.71±0.02	0.07±0.01	0.26±0.01	0.05±0.00
CTGAN	0.72±0.02	0.06±0.01	0.08±0.00	0.06±0.02
REAL	0.69±0.01	0.03±0.00	0.09±0.02	0.04±0.00
GReaT	0.91±0.22	3.83±0.09	0.13±0.00	2189.43±1012.52

(c) CS (↑)

Method	German	Adult	Diabetes	Buddy
HARMONIC	0.96	0.99	0.97	0.98
TVAE	0.98	0.97	0.97	0.99
CTGAN	0.90	0.99	0.99	1.00
REAL	0.98	0.99	0.99	0.97
GReaT	0.98	0.95	0.87	1.00

G Computational Resources

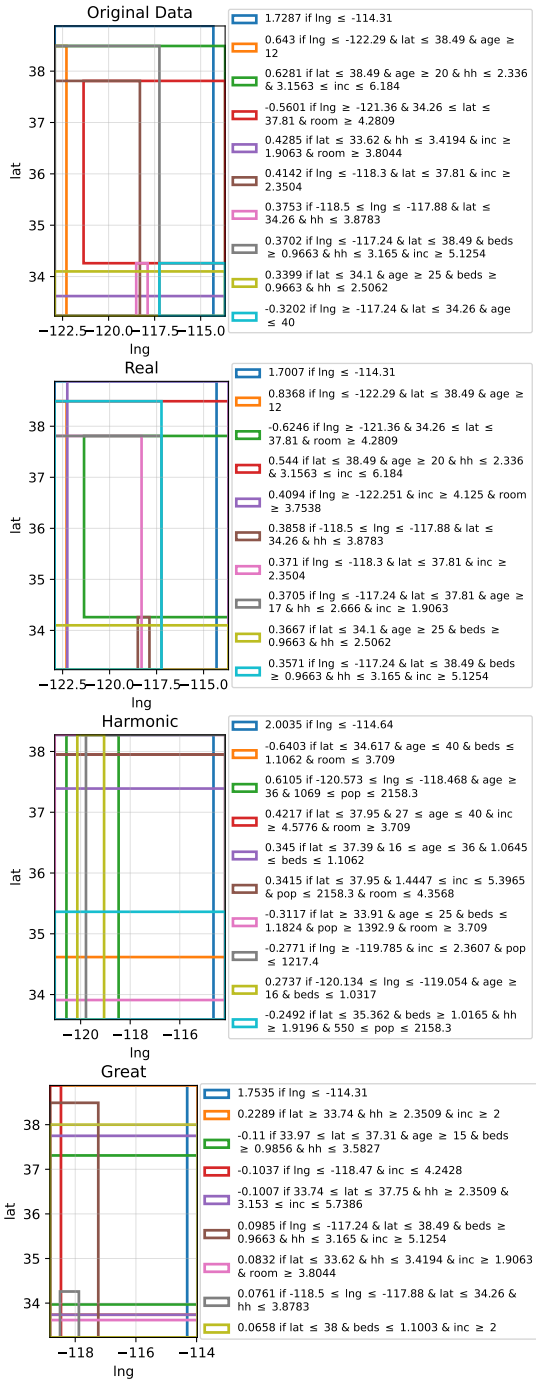
The evaluation experiments are conducted on a computer with processor "3.1GHz 6-Core Intel Core i5" and a memory of "72GB 2133 MHz DDR4".

H Training Details

Our goal is to audit *mechanism fidelity* using a compact, human-auditable ruleset, while ensuring the learned surrogate remains sufficiently predictive to be meaningful. Accordingly, we cap the maximum ruleset size at $K \leq 20$. This upper bound is motivated by interpretability: beyond ~ 15–20 additive rules, practitioners struggle to reliably read, compare, and reason about rule-level differences, especially when auditing multiple datasets and generators. In our setting, $K \leq 20$ provides enough capacity to capture dominant dependencies (i.e., the main explanatory mechanisms) while keeping the rule comparisons (support/lift shifts and ruleset differences) tractable for human inspection. Because our analysis compares rule mechanisms learned on real versus synthetic data, the rule learner must act as a *faithful surrogate* for the task signal. We therefore require the fitted rule model to reach at least 85% test accuracy (measured on held-out real data). This constraint guards against drawing conclusions from an underfit or unstable surrogate: if the rule model fails to capture the underlying decision boundary, observed "mechanism drift" could reflect surrogate misspecification rather than genuine distributional/mechanistic changes induced by the generator. In practice, the 85% threshold balances (i) sufficient task fidelity for meaningful auditing, and (ii) retaining sparsity so the resulting rules remain interpretable.

I More Experiments

Figures 4 to 13 show the visualization of the additive rule ensembles and their outputs trained on the original Adult, Abalone, Diabetes, German and California datasets and the corresponding synthetic data.



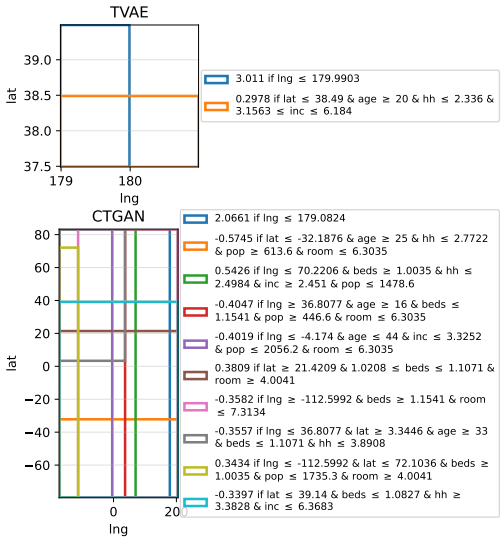


Figure 4: Visualization of the additive rule ensembles trained on original and synthetic data of California.

J Example of Rule set

Additive rule ensembles for California dataset generated by the original data.

+1.7287 if longitude≤-114.31

+0.9284 if median_income≥6.184 & total_rooms≥3.8044

+0.643 if housing_median_age≥12 & latitude≤38.49 & longitude≤-122.29

+0.6281 if households≤2.336 & housing_median_age≥20 & latitude≤38.49 & 3.1563≤median_income≤6.184

-0.5601 if 34.26≤latitude≤37.81 & longitude≥-121.36 & total_rooms≥4.2809

+0.4285 if households≤3.4194 & latitude≤33.62 & median_income≥1.9063 & total_rooms≥3.8044

+0.4253 if median_income≥3.1563 & total_rooms≥6.9697

+0.4142 if latitude≤37.81 & longitude≤-118.3 & median_income≥2.3504

+0.3753 if households≤3.8783 & latitude≤34.26 & -118.5≤longitude≤-117.88

+0.3702 if households≤3.165 & latitude≤38.49 & longitude≤-117.24 & median_income≥5.1254 & total_bedrooms≥0.9663

+0.3399 if households≤2.5062 & housing_median_age≥25 & latitude≤34.1 & total_bedrooms≥0.9663

-0.3202 if housing_median_age≤40 & latitude≤34.26 & longitude≥-117.24

-0.3073 if households≥2.0738 & median_income≤3.986

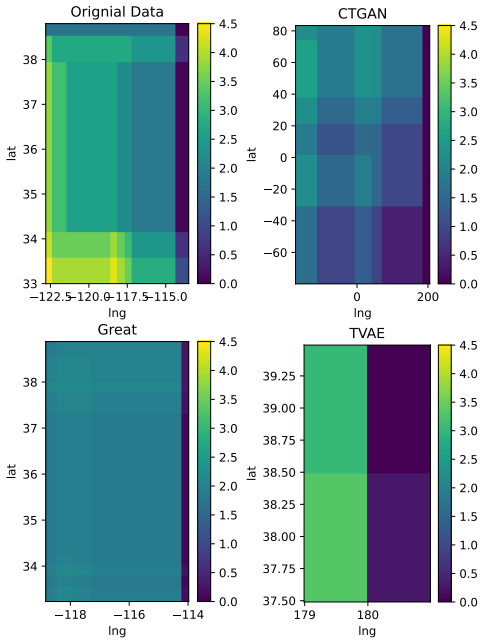


Figure 5: The outputs of the additive rule ensembles trained on the original California data and the data synthesized by CTGAN, GReaT and TVAE.

+0.2964 if households≤2.666 & housing_median_age≥17 & latitude≤37.81 & longitude≤-117.24 & median_income≥1.9063

+0.2849 if 34≤latitude≤37.48 & longitude≤-117.88

-0.2731 if latitude≥36.64 & -121.98≤longitude≤-118.3

-0.2701 if median_income≤2.7486 & total_rooms≥3.8044

+0.2613 if households≤2.978 & median_income≥4.4606 & population≥503.8

-0.2401 if households≥2.0738 & latitude≥33.86 & longitude≥-119.888 & population≤2527.4 & total_rooms≤6.2821

+0.1771 if median_income≥3.558 & population≤2527.4 & total_rooms≥5.5333

==========

Rule ensemble for synthesis data generated by REAL:

+1.7007 if longitude≤-114.31

+0.8368 if housing_median_age≥12 & latitude≤38.49 & longitude≤-122.29

+0.7458 if median_income≥6.184 & total_rooms≥3.8044

+0.6954 if housing_median_age≥52 & median_income≥2.7196 & population≤2737.1

-0.6246 if 34.26≤latitude≤37.81 & longitude≥-121.36 & total_rooms≥4.2809

+0.544 if households≤2.336 & housing_median_age≥20 & latitude≤38.49 & 3.1563≤median_income≤6.184

+0.4094 if longitude≥-122.251 & median_income≥4.125 & total_rooms≥3.7538

+0.3929 if median_income≥3.1563 & total_rooms≥6.9697

+0.3858 if households≤3.8783 & latitude≤34.26 & -118.5≤longitude≤-117.88

+0.371 if latitude≤37.81 & longitude≤-118.3 & median_income≥2.3504

+0.3705 if households≤2.666 & housing_median_age≥17 & latitude≤37.81 & longitude≤-117.24 & median_income≥1.9063

+0.3667 if households≤2.5062 & housing_median_age≥25 & latitude≤34.1 & total_bedrooms≥0.9663

+0.3571 if households≤3.165 & latitude≤38.49 & longitude≤-117.24 & median_income≥5.1254 & total_bedrooms≥0.9663

+0.3495 if 34≤latitude≤37.48 & longitude≤-117.88

-0.3492 if median_income≤3.1535 & total_rooms≤7.0783

+0.3254 if households≤3.4194 & latitude≤33.62 & median_income≥1.9063 & total_rooms≥3.8044

+0.3002 if households≤2.978 & median_income≥4.4606 & population≥503.8

-0.263 if households≥2.0738 & latitude≥33.86 & longitude≥-119.888 & population≤2527.4 & total_rooms≤6.2821

-0.2629 if latitude≥34.19 & longitude≤-118.31 & median_income≤5.4797 & population≤1694.8

-0.26 if households≥1.9689 & median_income≤6.7892 & population≥906 & total_rooms≥3.7538

==========

Rule ensemble for synthesis data generated by Harmonic:

+2.0035 if longitude≤-114.64

-0.6403 if housing_median_age≤40 & latitude≤34.617 & total_bedrooms≤1.1062 & total_rooms≤3.709

+0.6105 if housing_median_age≥36 & -120.573≤longitude≤-118.468 & 1069≤population≤2158.3

+0.4217 if 27≤housing_median_age≤40 & latitude≤37.95 & median_income≥4.5776 & total_rooms≥3.709

-0.3886 if households≤2.2097 & & housing_median_age≥23 &

1.9583≤median_income≤4.5776 & population≤1392.9

-0.3676 if households≥1.9196 & housing_median_age≤23 & 1.0317≤total_bedrooms≤1.1824 & total_rooms≤5.0182

+0.345 if 16≤housing_median_age≤36 & latitude≤37.39 & 1.0645≤total_bedrooms≤1.1062

+0.3415 if latitude≤37.95 & 1.4447≤median_income≤5.3965 & population≤2158.3 & total_rooms≤4.3568

-0.3117 if housing_median_age≤25 & latitude≥33.91 & population≥1392.9 & total_bedrooms≤1.1824 & total_rooms≥3.709

+0.2906 if households≤2.9718 & median_income≥1.4447 & 1.0829≤total_bedrooms≤1.1824 & total_rooms≥3.709

-0.2771 if longitude≥-119.785 & median_income≤2.3607 & population≤1217.4

+0.2737 if housing_median_age≥16 & -120.134≤longitude≤-119.054 & total_bedrooms≤1.0317

-0.2492 if households≥1.9196 & latitude≤35.362 & 550≤population≤2158.3 & total_bedrooms≥1.0165

-0.2253 if households≥1.9196 & 20≤housing_median_age≤36 & longitude≤-119.785 & population≤1069

-0.2238 if households≤3.2168 & housing_median_age≥27 & latitude≥34.617 & longitude≥-120.573 & total_bedrooms≤1.1824

+0.217 if households≤2.632 & longitude≤-118.05 & median_income≥1.9583 & population≥185.8 & total_bedrooms≤1.1824

-0.1933 if latitude≥35.875 & median_income≤2.7404

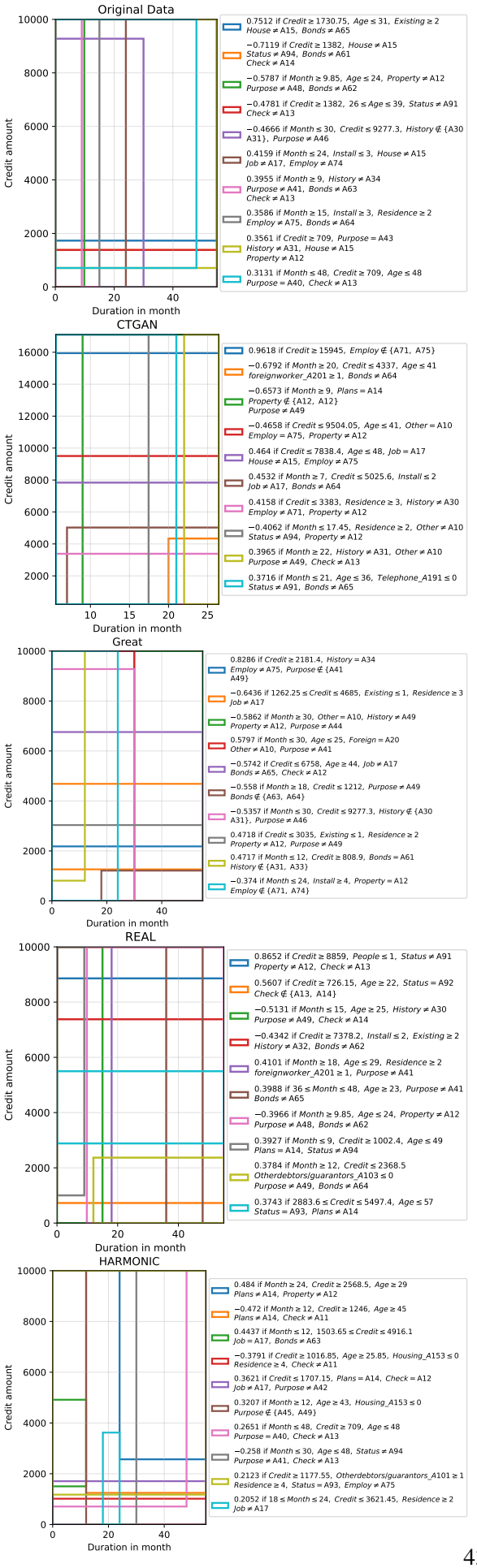
-0.1772 if households≤3.4642 & 20≤housing_median_age≤36 & latitude≥34.18 & total_rooms≤5.621

-0.1705 if median_income≤3.1542

-0.0713 if households≥2.5199 & housing_median_age≤27 & population≥185.8 & 3.709≤total_rooms≤7.0137

K Use of AI Tools for Grammar Checking and Editing

We used AI-based writing assistance tools solely for *language-level* support, including grammar correction, spelling, punctuation, and minor phrasing edits to improve clarity and readability. These tools were *not* used to generate new scientific content, design experiments, derive theoretical results, write proofs, create or modify data, or produce model outputs. All technical claims, methodological descriptions, equations, and conclusions were authored and verified by the authors. After applying AI-suggested edits, the authors reviewed and manually validated all changes to ensure accuracy, preserve the intended meaning, and maintain an appropriate academic tone.



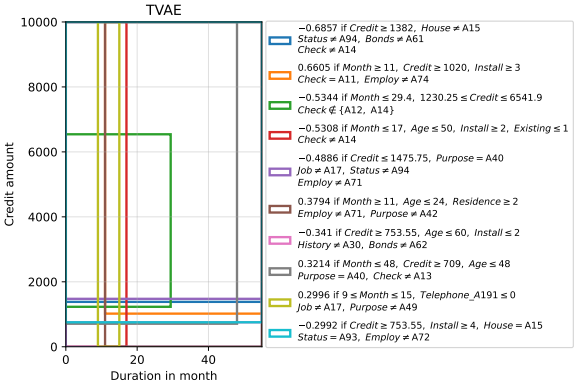


Figure 6: Visualization of the additive rule ensembles trained on original and synthetic data of German.

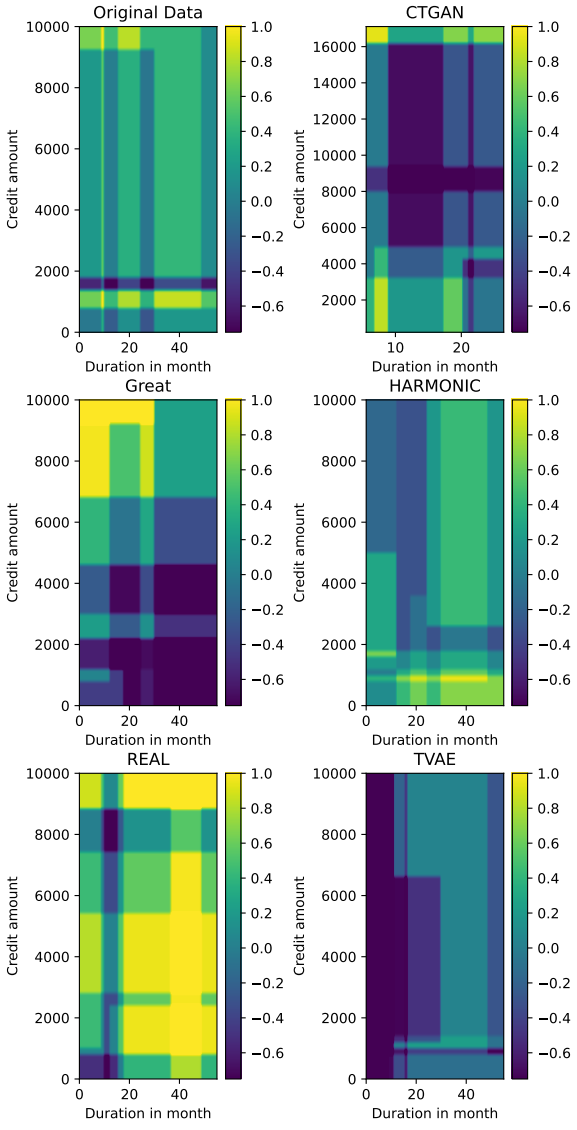
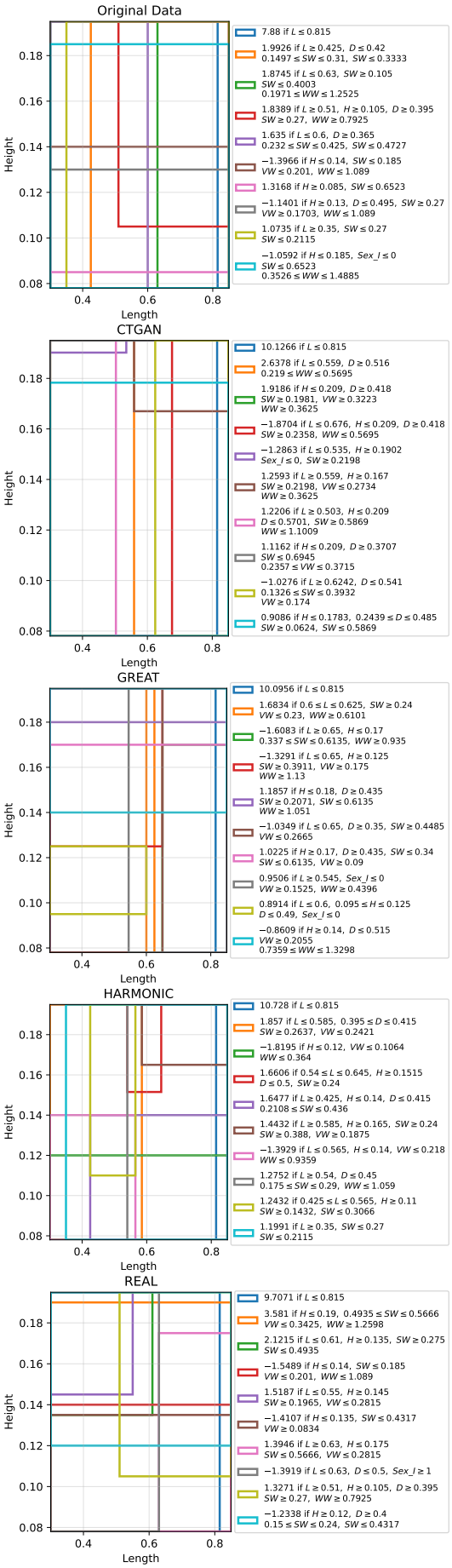


Figure 7: The outputs of the additive rule ensembles trained on original and synthetic data of German.



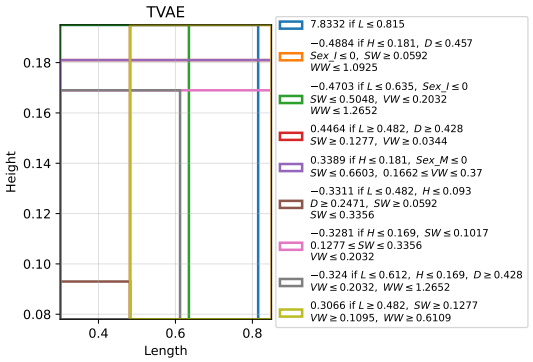


Figure 8: Visualization of the additive rule ensembles trained on original and synthetic data of Abalone.

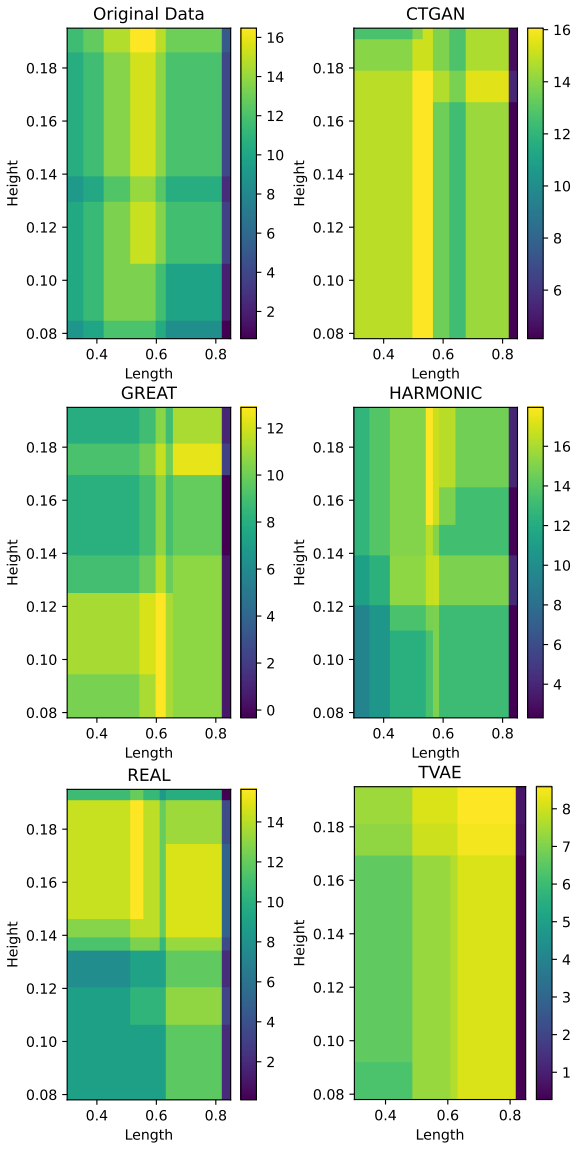
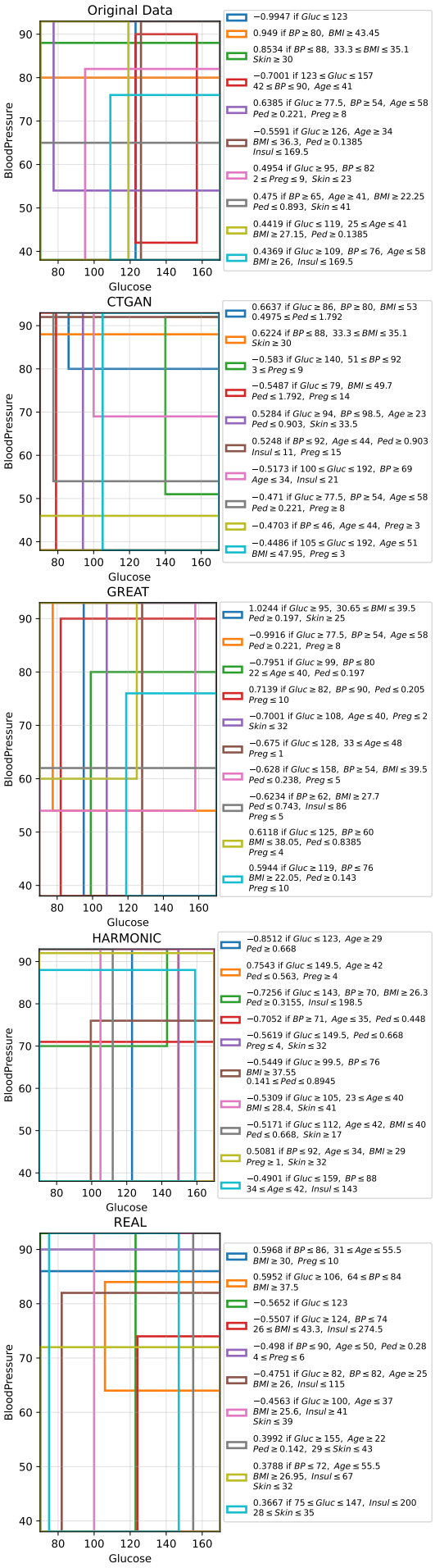


Figure 9: The outputs of the additive rule ensembles trained on original and synthetic data of Abalone.



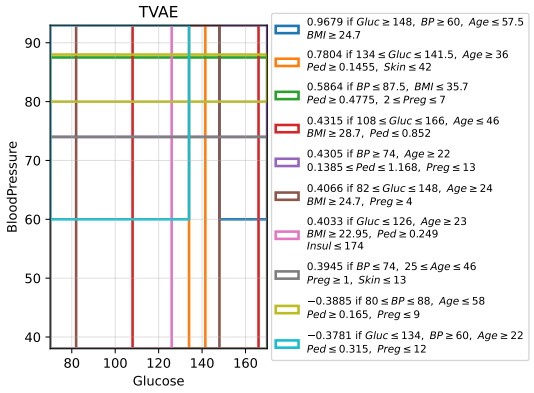


Figure 10: Visualization of the additive rule ensembles trained on original and synthetic data of Diabetes.

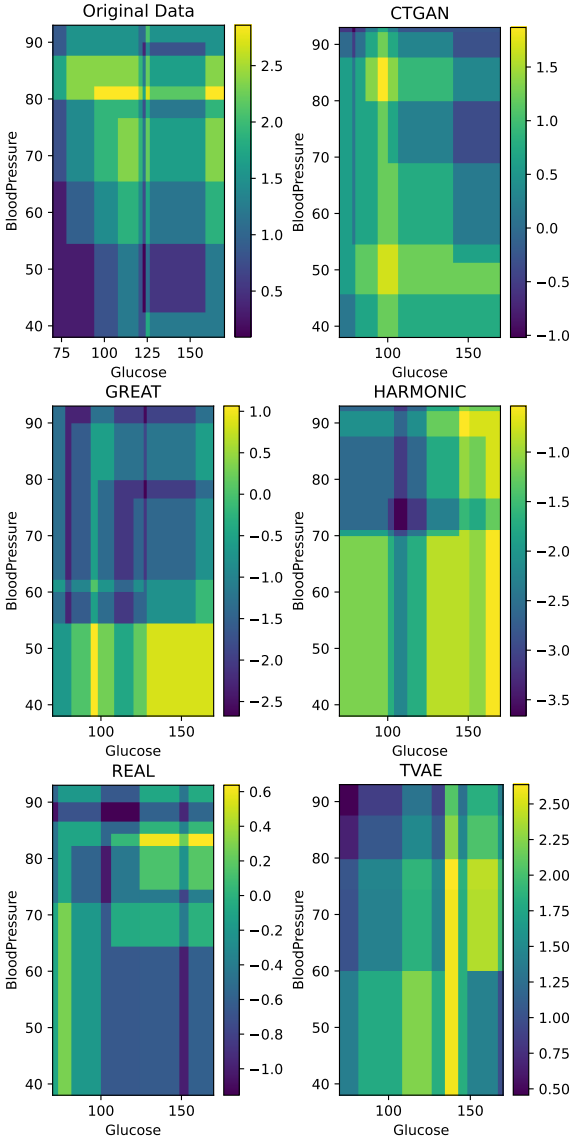


Figure 11: The outputs of the additive rule ensembles trained on original and synthetic data of Diabetes.

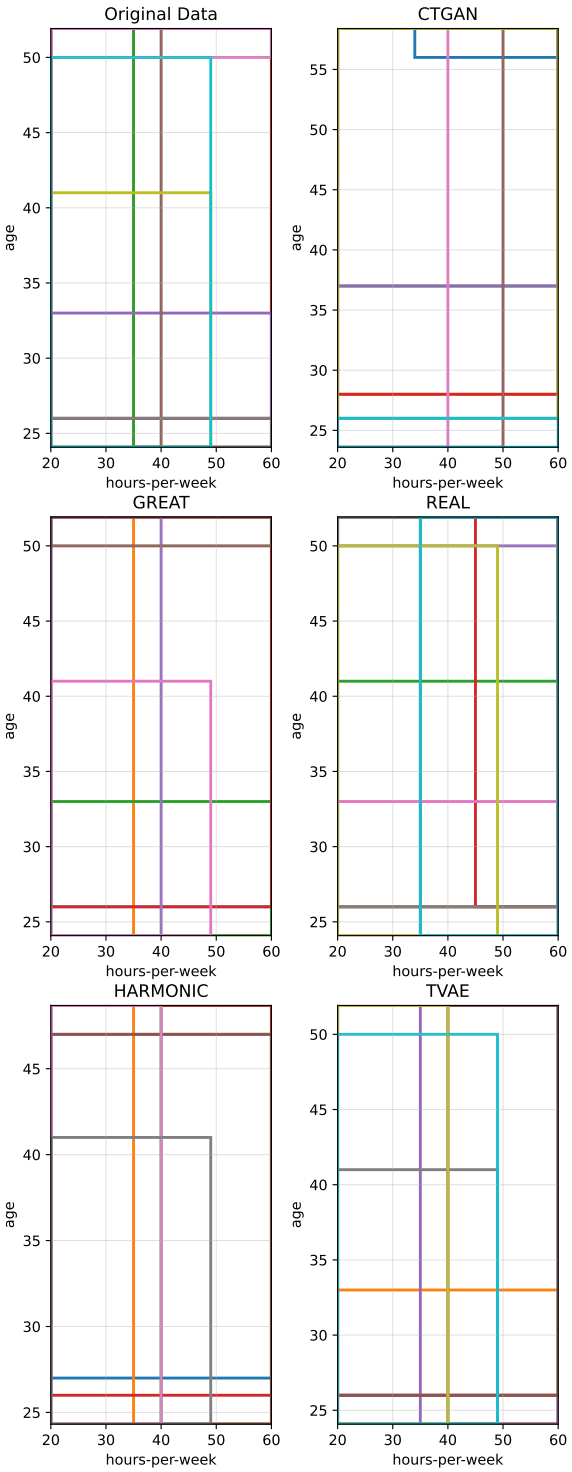


Figure 12: Visualization of the additive rule ensembles trained on original and synthetic data of Adult.

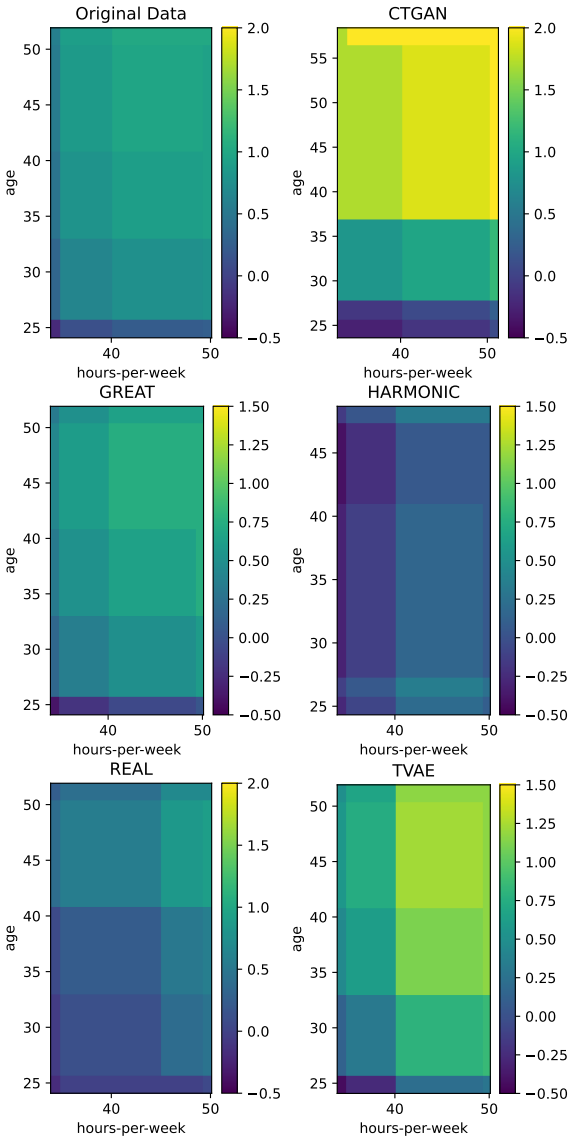


Figure 13: The outputs of the additive rule ensembles trained on original and synthetic data of Adult.