Evaluating Global Decision Faithfulness of LLMs with Structured Tabular Decision Simulations

Anonymous authors
Paper under double-blind review

Abstract

Large language models (LLMs) often achieve impressive predictive accuracy, yet correctness alone does not imply that their decisions are grounded in relevant, domain-appropriate factors. In structured decision settings, such as medical triage, financial risk assessment, or policy analysis, reliable performance requires more than producing correct labels: a model should make consistent decisions across multiple instances and rely on relevant, domain-grounded decision factors. We introduce **Structured Tabular Decision Simulations (STaDS)**, an evaluation framework that casts expert-like decision problems into tabular form and evaluates LLMs along three behavioral dimensions: (i) question and instruction comprehension, (ii) knowledge-based prediction, and (iii) reliance on relevant decision factors. The third dimension extends faithfulness evaluation from local reasoning traces to **global decision faithfulness**: whether a model's stated decision factors align with the factors that behaviorally affect its predictions across many instances. By analyzing 9 frontier LLMs across 15 diverse decision settings, we find that predictive competence and global decision faithfulness are empirically separable: models frequently achieve high accuracy while exhibiting low or negative alignment between stated and behaviorally measured feature reliance. This accuracy-faithfulness gap is consistent across model families and domains, and remains visible in a targeted domain-specialized medical-model case study. Our results highlight that accuracy metrics alone are insufficient and motivate the adoption of global faithfulness evaluation as a complementary protocol.

1 Introduction

Large language models (LLMs) are increasingly used as **decision-support systems** in professional domains due to their strong predictive performance, acting as physicians for medical triage, analysts for financial risk assessment, or policy advisors for legislative decisions Abd-Alrazaq et al. (2023); Brown et al. (2020); Dong et al. (2022); Zhao et al. (2023). In such applications, users expect models not only to produce correct outputs, but to base those outputs on the decision factors that should govern the task. Yet current evaluations still overwhelmingly focus on surface-level metrics such as accuracy or task completion. What is largely missing is an assessment of whether a model's predictions are grounded in the decision factors that should govern the task, going beyond whether any single prediction is correct. We do not claim to directly observe a model's internal cognitive processes. Instead, we adopt a behavioral perspective: a model that genuinely relies on domain-relevant decision factors should exhibit stable, feature-sensitive predictions across many structurally similar cases, and should report decision factors consistent with that measured reliance. This view is grounded in cognitive science accounts in which understanding supports the application of concepts and principles across instances (Mayer, 1989; Bereiter, 2005), but we operationalize it narrowly and behaviorally through observable decision outputs rather than internal states. The question we ask is not "does this model understand medicine?" but rather "does this model's behavior, across many decisions, reflect consistent reliance on medically relevant factors?" Human experts exemplify this property: a physician is expected to base diagnoses on established medical knowledge rather than incidental correlations, and their reliability is assessed not by a single correct answer but by coherent, domain-grounded behavior across cases.

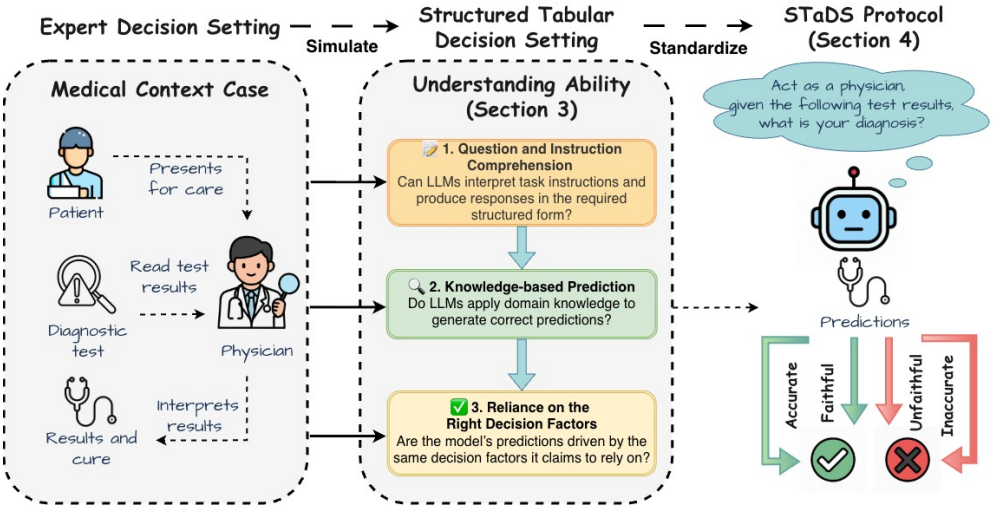


Figure 1: Overview of the STaDS protocol. STaDS simulates expert-like decision-making in structured tabular settings and evaluates LLM behavior along three dimensions: (1) question and instruction comprehension, assessing task interpretation and output adherence; (2) knowledge-based prediction, evaluating whether the model can produce accurate decisions; and (3) reliance on relevant decision factors, testing whether the model's stated factors align with the factors that behaviorally affect its predictions. Together, these dimensions provide an operational framework for evaluating structured decision competence and global decision faithfulness in LLMs.

Recent work has begun to ask whether LLMs "reason faithfully", especially through chain-of-thought (CoT) rationales. Such analyses mostly operate at the level of a single problem instance: does the explanation accompanying an answer reflect the steps that actually produced it? (Wei et al., 2022; Lewkowycz et al., 2022; Barez et al., 2025; Yu et al., 2024) These studies expose important failures of ***local*** reasoning faithfulness, but they do not tell us whether a model behaves like a reliable expert across many decisions in a domain (Jacovi & Goldberg, 2020; Arcuschin et al., 2025). In this work, we extend the concept of faithfulness from local reasoning traces to ***global*** decision faithfulness, exploring whether the model's predictions across multiple cases consistently rely on meaningful decision factors rather than superficial correlations or unstable shortcuts (see Fig. 1).

This motivates a shift in evaluation perspective. We propose the **Structured Tabular Decision Simulations (STaDS)** protocol: a systematic evaluation framework that casts expert-like decision problems into tabular form, enabling controlled assessment of both predictive performance and global decision faithfulness. Unlike reasoning benchmarks that emphasize step-by-step justifications, where models may make or conceal errors in intermediate reasoning steps (Turpin et al., 2023), STaDS focuses on an end-to-end behavioral question: whether LLMs consistently apply the decision factors that govern outcomes within a domain, and whether their stated factor rankings match their measured behavioral reliance. Tabular decision settings are particularly useful for this purpose because features and labels are explicit, structured, and semantically interpretable (Kim, 2024; Zarlenga et al., 2023; Li et al., 2023) (More discussion in Sec. 3.1.1).

Research Questions and Contributions. We target two central research questions:

> ***RQ1:*** *To what extent do LLMs exhibit structured decision competence across diverse tabular decision settings, beyond isolated prediction accuracy?*
>
> ***RQ2:*** *To what extent do LLMs exhibit global decision faithfulness, aligning their stated decision factors with the factors that behaviorally affect their predictions within a domain?*

To answer these questions, we present a unified evaluation framework and empirical study of structured decision competence and global decision faithfulness in LLMs. Our primary contributions are as follows:

- **STaDS Protocol:** We propose the **Structured Tabular Decision Simulation (STaDS)** protocol as a behavioral evaluation framework for structured decision competence and global decision faithfulness in LLMs. STaDS operationalizes this target through three behavioral proxies: (i) *question and instruction comprehension*, the ability to interpret task instructions and follow structured output specifications; (ii) *knowledge-based prediction*, the capacity to apply intrinsic and in-context knowledge to produce accurate predictions; and (iii) *reliance on relevant decision factors*, the degree to which a model's predictions are driven by the same decision factors it claims to rely on. The framework provides a structured, reproducible, and extensible setting for connecting interpretability-oriented behavioral interventions with explainability-oriented self-reports in LLM evaluation (Jacovi & Goldberg, 2020; Agarwal et al., 2022).

- **STaDS Metrics and Benchmarks.** Building on these three axes, we define targeted metric suites for each: (1) **Comprehension Fidelity**, captured by Len-F1, UnkLbl%, and the format-related component of Penalized Accuracy to quantify instruction adherence; (2) **Predictive Competence**, measured through zero/few-shot Accuracy, Macro-F1, and overall Penalized Accuracy to assess knowledge-grounded prediction; and (3) **Decision Faithfulness**, evaluated via LAO-based feature reliance, Self-Decision Faithfulness, and SelfAtt@k to determine whether stated and behaviorally measured decision factors align at the global level. To support systematic evaluation, we curate a suite of *15* real-world tabular datasets spanning healthcare, finance, and public policy, summing to approximately *160k* decision instances. This is accompanied by *standardized instruction templates* designed to reduce prompt-specific artifacts and promote reproducibility.

- **Empirical Insights and Robustness Analyses.** We conduct a large-scale study of *9* state-of-the-art LLMs, including advanced closed-source models (GPT, Gemini (Achiam et al., 2023; Team et al., 2023)) and leading open-source models (LLaMA, Mistral, DeepSeek, Qwen, Gemma), across all benchmarks (Dubey et al., 2024; DeepSeek, 2024; Yang et al., 2025; Team et al., 2025). Our analysis reveals that predictive competence and global decision faithfulness are empirically separable: across model–dataset pairs, high penalized accuracy frequently co-occurs with low or negative self–LAO alignment, and this pattern holds across model families, domains, and prompting regimes. A smaller subset of model–dataset pairs achieves both strong predictive performance and positive decision faithfulness, confirming that the two properties are compatible but not automatically coupled. We further test the robustness of these findings through alternative perturbation operators, sensitivity analysis, post-processing audits, correlated group ablations, and a domain-specialized medical model case study.

2 Background & Related Work

We position STaDS relative to work on LLM explainability, reasoning faithfulness, tabular reasoning, in-context learning, and benchmark design. Our focus is not to define LLM understanding in its broadest sense, but to operationalize one behaviorally testable dimension of it: whether LLMs making repeated structured decisions rely on stable, semantically meaningful decision factors, and whether their stated factors align with their measured behavioral reliance.

Explainability & Interpretability for LLMs. Explainability involves providing *human-understandable justifications* for model decisions, typically through post-hoc methods linking inputs to outputs (Adadi & Berrada, 2018; Li et al., 2022). Interpretability, conversely, emphasizes *transparent internal mechanisms* such as weights and attention interactions, making the model's behavior intrinsically comprehensible (Das & Rad, 2020; Ali et al., 2023). In the LLM setting, explainability has been pursued through prompt-based rationales (Liu et al., 2023), contrastive saliency for token-level influence (Min et al., 2023), and gradient-free feature attribution methods for structured tasks (Sui et al., 2024). While these approaches can generate plausible explanations, they do not necessarily guarantee *faithfulness*, i.e., alignment between an explanation and the factors that actually affect model predictions (Jacovi & Goldberg, 2020; Agarwal et al., 2022). Evaluation practices also remain fragmented, often relying on human judgments or narrow task-specific benchmarks.

STaDS draws on this XAI perspective, but uses it for behavioral evaluation rather than for explaining a fixed trained model. Specifically, we compare a model's self-reported feature ranking with perturbation-based behavioral reliance measured by missing-information interventions. This allows us to test whether the factors the model claims to use are aligned with the factors to which its predictions are actually sensitive.

Reasoning & Unfaithfulness. Chain-of-thought prompting is widely used to evaluate logical reasoning in free-form text (Wang et al., 2024), where advanced models generate step-by-step intermediate reasoning. However, such reasoning can be unreliable: models may produce errors in intermediate steps or exploit latent shortcuts that yield correct answers for the wrong reasons. Recent studies further highlight evidence of *unfaithfulness* in both thinking and non-thinking frontier models, showing that they sometimes provide correct or incorrect answers accompanied by fabricated rationales or illogical justifications (Barez et al., 2025; Lanham et al., 2023; Arcuschin et al., 2025). We distinguish this reasoning-level unfaithfulness from the *behavioral inconsistency* targeted in STaDS, which captures differences between what a model claims and how it actually decides. Rather than evaluating intermediate reasoning steps, STaDS evaluates whether a model's *global feature reliance* across many related decisions aligns with its stated attributions, shifting the focus from local rationale faithfulness to domain-level decision faithfulness.

Step-level Reasoning & Global Attribution Reasoning. Arcuschin et al. (2025) evaluates *Unfaithful Illogical Shortcuts* through three steps: answer correctness, step criticality, and step unfaithfulness. Their analysis centers on intermediate reasoning steps that accumulate toward a **single** decision, making models vulnerable to biases or fabricated justifications at the **step level**. In contrast, STaDS deliberately avoids relying on step-level reasoning traces. Instead, it evaluates whether a model's **overall attribution ranking**, elicited through self-attribution, faithfully reflects the features governing a **set** of related decisions. We acknowledge that self-stated attribution rankings may not perfectly reflect an LLM's internal decision process; this is precisely why STaDS compares them against perturbation-based behavioral reliance.

Tabular Reasoning and Prediction. A growing body of work develops table-centric models tailored for structured data, including TAPAS (Herzig et al., 2020), TURL (Deng et al., 2022), TableLlama (Zhang et al., 2023), and TabPFN (Hollmann et al., 2025). These models support tasks such as entity linking, column annotation, and fact extraction (Deng et al., 2022; Zhang et al., 2023), and more broadly span table interpretation, augmentation, question answering, fact verification, and dialogue generation. While these methods advance training efficiency and correctness across diverse applications, they often rely on specialized architectures restricted to particular table formats or tasks (Sui et al., 2024). Specifically, they mainly target table semantics, tabular prediction, or question answering.

By contrast, STaDS treats tabular data as a structured decision-simulation setting. Explicit columns enable controlled missing-information interventions, systematic perturbations, and unambiguous mapping between self-reported feature rankings and input attributes. STaDS is therefore not another table-specific architecture or tabular QA benchmark; it is a complementary evaluation framework for testing whether LLMs behave as faithful repeated decision-makers within a structured domain. [1]

In-Context Learning (ICL). ICL enables LLMs to adapt to new tasks from demonstrations provided in the prompt without parameter updates (Dong et al., 2022; Brown et al., 2020). Prior work has shown that ICL is sensitive to demonstration selection, order, label mapping, and formatting (Wang et al., 2023; Wei et al., 2023; Akyürek et al., 2022; Min et al., 2022). These studies highlight that in-context competence can be fragile. STaDS is complementary: rather than only measuring whether demonstrations improve answer accuracy, it asks whether the resulting predictions are behaviorally aligned with the decision factors the model claims to use. Moreover, STaDS evaluates a sequence of structured decisions within a domain, enabling global analysis of decision-factor reliance rather than only instance-level correctness.

Benchmark Landscape. Existing benchmark suites typically assess competence, reasoning, or explanation plausibility in isolation, such as GLUE and MMLU for broad competence (Wang et al., 2018; Hendrycks

[1] Recent table-specific models such as TableLlama, built on LLaMA-2 (7B), can handle contexts of up to 8K tokens, yet remain unable to process the longer tabular inputs considered in this work.

Table 1: Comparison between STaDS and related evaluation paradigms. Existing benchmarks typically evaluate whether a model can answer an individual query, retrieve information from context, reason over table entries, or fit a tabular prediction problem. STaDS instead evaluates LLMs as repeated decision-makers within a structured domain, with explicit comparison between self-reported decision factors and perturbation-based behavioral reliance.

Evaluation paradigm	**Typical input**	**Unit of evaluation**	**Primary evaluation signal**	**Repeated domain-level decisions?**	**Tests stated-vs-behavioral decision factors?**
Multi-hop question answering	Natural-language documents, facts, or passages	Individual question	Answer correctness and supporting evidence retrieval	No	No; explanations may be evaluated, but not global feature reliance
Long-context question answering	Long documents or multi-document contexts	Individual query over a long context	Retrieval accuracy, answer correctness, or citation quality	No	No; typically tests context use rather than stable decision-factor reliance
Tabular question answering	Table plus natural-language question	Individual table query	Correct answer extraction, aggregation, or reasoning over table cells	Limited	No; focuses on answering questions about a table, not decision faithfulness across a domain
LLM-based tabular prediction	Serialized tabular instances or examples	Individual prediction or dataset-level predictive performance	Accuracy, F1, or calibration	Sometimes	Rarely; usually evaluates predictive competence without comparing self-attribution to behavioral reliance
Chain-of-thought faithfulness	Problem statement plus generated rationale	Individual reasoning trace	Whether the rationale supports or causally affects the answer	No	Local only; evaluates rationale–answer faithfulness for a single instance
Post-hoc feature attribution / XAI	Trained predictive model and input features	Model prediction or dataset-level attribution	Feature importance, explanation stability, or explanation agreement	Yes, for trained models	Partially; usually explains a fixed ML model rather than evaluating an LM as a prompted decision-maker
STaDS (ours)	Structured tabular decision simulations with natural-language instructions	Domain-level repeated decision process	Comprehension fidelity, predictive competence, and global decision faithfulness	**Yes**	**Yes; compares self-reported decision factors with perturbation-based behavioral reliance**

et al., 2020), GSM8K and DROP for reasoning (Cobbe et al., 2021; Dua et al., 2019), and ERASER or e-SNLI for explanation plausibility (DeYoung et al., 2019; Camburu et al., 2018). However, these benchmarks usually evaluate competence, reasoning, or explanation quality separately. They rarely test whether an LLM behaves as a repeated decision-maker whose stated decision factors match behaviorally measured reliance across a structured domain. XAI benchmarks such as OpenXAI (Agarwal et al., 2022) focus on explanation faithfulness for fixed predictive models, whereas STaDS evaluates LLMs directly through prompted tabular decision simulations. In this sense, STaDS integrates predictive competence, instruction fidelity, and global decision faithfulness within a controlled evaluation setting.

The Gap. Existing LLM evaluation paradigms typically assess whether a model can answer individual questions from textual evidence, retrieve or aggregate information from tables, or perform tabular prediction. In parallel, traditional XAI primarily explains the behavior of trained predictive models. STaDS bridges these lines of work by using structured tabular decision settings to evaluate LLMs as repeated decision-makers within a domain. The key question is not only whether the model is accurate, but whether its stated decision factors align with behaviorally measured feature reliance. Table 1 summarizes this distinction.

3 STaDS Protocol: Evaluating Global Decision Faithfulness

We introduce STaDS as a systematic evaluation framework for testing global decision faithfulness in LLMs under structured tabular decision simulations.

3.1 Operationalizing Understanding as Global Decision Faithfulness

Understanding is a broad and abstract notion, and we do not claim to measure it exhaustively. Instead, we focus on one operational dimension relevant to structured decision-making: whether a model can follow the task, make accurate predictions, and align its stated decision factors with behaviorally measured reliance. This view is motivated by cognitive-science accounts in which understanding supports the application of concepts and principles to guide intelligent behavior (Mayer, 1989; Bereiter, 2005). In STaDS, we therefore operationalize understanding as observable decision behavior rather than as direct access to a model's internal cognitive state.

We use the following operational definition:

> **A model's observable ability to follow a structured decision task, make accurate predictions, and align its stated decision factors with measured behavioral reliance.**

We decompose this operational target into three dimensions:

1. **Question and instruction comprehension**: This dimension is the ability to correctly interpret a task: to recognize what is being requested, identify the goal state, and determine the appropriate form of response. In cognitive terms, this is often described as constructing a situation model or problem representation: "what is going on in this task, and what is being asked of me?" (Chi et al., 2014). Successful comprehension requires mapping linguistic instructions to an internal representation of required actions, not merely decoding words. In human learners, failure at this stage, such as misreading the question or misunderstanding constraints, is considered a failure of understanding even before problem solving begins (Chi et al., 1981). In STaDS, instruction comprehension is therefore evaluated behaviorally through output validity, length fidelity, and adherence to the required prediction format.

2. **Knowledge-based prediction**: This dimension is the capacity to apply relevant prior knowledge, in-context examples, or induced task structure to produce correct predictions in a new decision context. This corresponds to transfer in cognitive science: the ability to apply learned knowledge or principles from one setting to another (Bransford et al., 2000). Transfer is widely treated as a marker of deeper understanding, because it suggests that performance is not tied only to rote pattern matching or memorized responses. In STaDS, knowledge-based prediction is assessed through conventional predictive metrics such as accuracy, macro-F1, and penalized accuracy over masked tabular instances.

3. **Global decision faithfulness**: This dimension captures whether the model's decisions are sensitive to the same task-relevant factors it identifies as important. Expert performance is not only accurate; it is also structured around relevant features of the decision problem. In STaDS, we therefore compare (i) what an LLM claims matters, via self-attribution, with (ii) what affects its predictions under feature perturbation. Alignment between these two signals provides evidence that the model's stated decision factors reflect its measured decision behavior; misalignment suggests post-hoc or unreliable self-attribution.

Taken together, these three dimensions define the operational target of STaDS. They do not exhaust all meanings of understanding, but they capture a practical form of structured decision competence: following the task, predicting accurately, and reporting decision factors that match measured behavioral reliance.

3.1.1 What are Decision Factors?

In structured decision settings, decision factors are the explicit, semantically grounded variables that may influence outcomes within a domain. Each factor corresponds to an interpretable input attribute, such as age, income, tumor size, or credit history, that a decision-maker may consider when predicting the target label. STaDS does not assume that these features are necessarily causal. Rather, it treats them as candidate decision-relevant variables whose importance can be compared across three signals: self-reported importance, statistical feature-label association, and behavioral sensitivity under perturbation.

3.2 Why Tabular Decision Simulations?

Tabular decision simulations provide a principled setting for evaluating global decision faithfulness. Their design offers several advantages over other data formats:

1. **Instance-level structure.** Each row corresponds to a complete, self-contained decision instance: the set of feature values in that row provides the information available for determining an outcome. This framing mirrors many case-by-case judgments in real-world domains. It makes the available evidence explicit for each case, although it does not eliminate the possibility that models exploit spurious correlations or dataset-level regularities. Unlike tasks in vision or multimodal reasoning, which often require additional perceptual processing or contextualization, tabular data provides a decision-ready input format with explicit and interpretable features.

2. **Global-level faithfulness.** Because attributes are explicitly named, defined, and consistently shared across rows, tabular data naturally supports analysis of *global feature importance*, an established goal in XAI (Samek et al., 2017; Ali et al., 2023). This distinguishes tabular simulations from conventional reasoning tasks, where faithfulness is typically examined at the level of individual decisions and intermediate reasoning steps. In contrast, tabular simulations enable evaluation of whether a model's decision-making aligns with coherent, domain-wide patterns of feature reliance, providing a bridge between local prediction accuracy and global decision consistency.

3. **Clear decision setting.** While many evaluation tasks adopt binary questions, such as yes/no in open-ended text or image-based object detection, such questions are often constructed for the benchmark (Li et al., 2023; Arcuschin et al., 2025). In contrast, tabular data naturally encode decision outcomes as binary or multi-class classification labels, which can be directly transformed into prediction tasks without additional question design. This framing also avoids some biases inherent in multiple-choice formats, where prior work shows that models can change answers under superficial perturbations (Arcuschin et al., 2025). While our present focus is classification, extending the protocol to regression tasks is a natural direction for future work.

4. **Explainable end-to-end evaluation.** Because both features and labels carry explicit, domain-grounded meanings, such as age, income, or medical indicators, they are directly interpretable to humans without requiring additional segmentation or concept mapping. Tabular simulations capture the full evaluation pipeline from structured inputs to predicted outputs, without requiring access to model internals or relying on intermediate reasoning traces. This design allows comprehension fidelity, predictive competence, and decision faithfulness to be assessed within a unified decision setting.

5. **Generality with systematic probing.** Tabular data appear across many real-world domains, including healthcare, finance, science, policy, and education. Their structured format also enables systematic perturbations, such as varying the number of rows, masking attributes, replacing feature values, or ablating features. These interventions provide behavioral tests of whether predictions depend on the decision factors the model claims to use.

In this way, tabular decision simulations provide a distinct, structured, and reproducible environment where global decision faithfulness can be quantitatively assessed against explicit, domain-relevant input features. At the same time, STaDS differs from standard tabular ML evaluation: the model receives the task through

natural-language instructions, must infer and apply the relevant decision structure in context, and must report the factors it believes are important. This makes the setting suitable for quantitatively assessing whether LLMs make structured decisions in a way that is both accurate and globally faithful.

3.3 STaDS Protocol Tasks

In this section, we operationalize global decision faithfulness through observable behavioral indicators. These indicators correspond to three aspects of structured decision behavior: task compliance, predictive competence, and alignment between stated and behaviorally measured decision factors.

Behavioral Indicators for Structured Decision Behavior. Violations of the output specification provide diagnostic signals for different aspects of model behavior. **Question and instruction comprehension violations**: producing the wrong number of predictions, misaligned outputs, or irrelevant text indicates that the model has misinterpreted task instructions or failed to identify the required outputs. These violations assess whether the model correctly comprehends the task and follows the required response format. **Knowledge-based prediction violations**: generating labels in the correct format but with low accuracy, or producing invalid labels outside $\mathcal{Y}$, suggests that the model has failed to ground its prior knowledge or in-context adaptation in the current decision task. These violations assess whether the model can apply relevant knowledge to make correct predictions. **Global decision faithfulness violations**: providing self-claimed feature rankings inconsistent with measured feature reliance indicates that the model's stated rationale does not align with its observed decision behavior. Such misalignment suggests that a model may make accurate predictions while providing unreliable or post-hoc self-attributions.

STaDS Evaluation Tasks. STaDS evaluates structured decision behavior through three tasks:

1. **Comprehension Fidelity**: This task evaluates whether the model can correctly interpret task instructions and adhere to output specifications. It checks whether the model produces the correct number of predictions, follows the specified output format, and uses only valid labels.

2. **Predictive Competence**: This task measures whether the model can generate accurate predictions for masked rows under zero-shot and few-shot settings. It tests whether the model can apply relevant knowledge or infer useful decision patterns from in-context demonstrations.

3. **Global Decision Faithfulness**: After generating predictions, the model is asked to provide a feature-importance ranking through self-attribution. This ranking is compared with behavioral attributions obtained through systematic perturbations such as Leave-Any-Out (LAO) analysis. This task probes whether the model's stated decision factors align with the features that measurably affect its predictions.

4 STaDS Metrics: Quantifying Comprehension, Competence, and Faithfulness

To evaluate structured decision behavior, STaDS introduces metrics for comprehension fidelity, predictive competence, and global decision faithfulness. We first formalize the input and output.

Input formulation. A tabular decision task is represented as $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where each $x_i \in \mathbb{R}^d$ is a feature vector and $y_i \in \mathcal{Y}$ is the corresponding label. To evaluate the model, $\mathcal{D}$ is rendered into a structured prompt $\mathcal{C} = (I, T, S_k)$ at inference time, consisting of:

- I: a natural-language instruction specifying the professional role, task, target-label encoding, and an attribute glossary mapping features to domain concepts;
- T: a textual rendering of the structured table, where target labels are masked as `class=?`;
- $S_k = \{(x^{(j)}, y^{(j)})\}_{j=1}^{k}$: an optional set of k in-prompt demonstrations.

Table 2: Core notation used in STaDS.

Symbol	Meaning
$\mathcal{D}$	Tabular decision dataset or domain
x_i	Feature vector for instance i
y_i	Ground-truth label for instance i
$\mathcal{Y}_{\text{valid}}$	Valid label set specified by the task
m	Number of input features
n_g	Number of expected ground-truth predictions
n_p	Number of predictions produced by the model
n_a	Number of aligned prediction–label pairs
Acc	Accuracy over aligned prediction–label pairs
PenAcc	Accuracy penalized for length and label-validity violations
Δ_j	Performance change after perturbing feature j
π_{self}	Feature ranking stated by the model
π_{LAO}	Feature ranking induced by LAO perturbation

Output specification. The LLM f_θ is required to output predictions for the masked rows:

$$\hat{\mathbf{y}} = \big(\hat{y}_1, \ldots, \hat{y}_{n_p}\big) = f_\theta(\mathcal{C}),$$

where n_p is the number of predictions generated. Outputs are expected to follow strict formatting rules: the number of predictions must match the number of queried rows, every prediction must belong to the valid label set, and no additional text should be returned. These constraints ensure that performance reflects not only correctness but also task compliance.

Prediction alignment. Let n_g denote the number of ground-truth labels expected for evaluation and n_p the number of predictions produced by the model. Since LLMs may return too few or too many predictions despite explicit instructions, we define the aligned evaluation length as:

$$n_a = \min(n_p, n_g),$$

so that only the first n_a prediction–label pairs $(\hat{y}_i, y_i)$ are used for accuracy-based scoring. Length mismatch is not ignored; it is separately penalized through the comprehension-fidelity metrics below.

We define the following sets:

- **Valid label set ($\mathcal{Y}_{\text{valid}}$)**: the complete set of permissible label values specified by the task. Any $\hat{y}_i \notin \mathcal{Y}_{\text{valid}}$ is treated as an invalid prediction.
- **Ground-truth label set ($\mathcal{Y}_{\text{gt}}$)**: the set of unique ground-truth labels among the aligned pairs,

$$\mathcal{Y}_{\text{gt}} = \{y_i : 1 \le i \le n_a\}.$$

- **Predicted label set ($\hat{\mathcal{Y}}$)**: the set of valid predicted labels among the aligned pairs,

$$\hat{\mathcal{Y}} = \{\hat{y}_i : \hat{y}_i \in \mathcal{Y}_{\text{valid}}, 1 \le i \le n_a\}.$$

Zero/Few Shot Settings. STaDS explicitly considers performance in both zero-shot and few-shot settings, since the gap between them reflects the extent to which models rely on intrinsic knowledge versus in-context adaptation.

- **Zero-shot ($k = 0$):** The model receives prompt $\mathcal{C}$ with no demonstrations and must predict all rows whose labels are masked as `class=?`. This setting tests whether the model can ground its predictions directly in the instructions and table structure, reflecting its intrinsic knowledge grounding Chen et al. (2023). In other words, does the model already know enough about the decision domain to perform competently without examples?

- **Few-shot ($k > 0$):** The model receives $\mathcal{C}$ with k labelled demonstrations injected into the prompt. This probes whether the model can align row-level features with labels when given exemplars, i.e., whether it can perform in-context learning in a manner analogous to how humans adapt to case-based examples Petroni et al. (2019).

General Metrics. We adopt conventional classification metrics Accuracy (Acc), Macro-F1, and Label-Set Jaccard (Set-Jacc) for reference. These capture baseline task performance and provide a point of comparison to existing tabular classification benchmarks.

For each $c \in \mathcal{Y}_{\text{gt}}$, compute precision, recall, and $F1_c$ on the aligned set.

$$\text{Acc} \;=\; \frac{1}{n_a}\sum_{i=1}^{n_a} 1[\hat{y}_i = y_i], \quad \text{Macro-F1} \;=\; \frac{1}{|\mathcal{Y}_{\text{gt}}|}\sum_{c\in\mathcal{Y}_{\text{gt}}} F1_c.$$

The label-set Jaccard is given by:

$$\text{Set-Jacc} \;=\; \frac{|\hat{\mathcal{Y}} \cap \mathcal{Y}_{\text{gt}}|}{|\hat{\mathcal{Y}} \cup \mathcal{Y}_{\text{gt}}|}.$$

Comprehension Fidelity Metrics. We explicitly penalize *over/under-production* as well as *unknown labels*, as these indicate a failure to understand and follow task instructions.

- **Length F1 (Len-F1)**: Len-F1 measures output-length fidelity, where the model produces incorrect number of expected predictions. Let $P_L = n_a/n_p$ (set 0 if $n_p = 0$) denote precision with respect to output length, and $R_L = n_a/n_g$ (set 0 if $n_g = 0$) denote recall with respect to the number of ground-truth labels. Then,

$$\text{Len-F1} \;=\; \begin{cases} \dfrac{2P_L R_L}{P_L + R_L}, & \text{if } P_L + R_L > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

- **Unknown-Label Rate (UnkLbl%)** UnkLbl% refers to the fraction of all produced predictions outside the valid label set:

$$\text{UnkLbl\%} \;=\; \frac{1}{n_p}\sum_{i=1}^{n_p} \mathbf{1}[\hat{y}_i \notin \mathcal{Y}_{\text{valid}}] \quad (0 \text{ if } n_p = 0). \tag{2}$$

Predictive Competence Metrics. We integrate correctness, output-length fidelity, and label validity into a single measure of predictive quality, **Penalized Accuracy (PenAcc)**:

$$\text{PenAcc} = \max\left\{0,\; \text{Acc} - \big(\alpha(1 - \text{Len-F1}) + \beta\,\text{UnkLbl\%}\big)\right\}. \tag{3}$$

where $\alpha, \beta \geq 0$ are penalty weights. Unless otherwise stated, we use the neutral default $\alpha = \beta = 0.5$. Because both $(1 - \text{Len-F1})$ and UnkLbl% lie in $[0, 1]$, choosing $\alpha + \beta \leq 1$ bounds the maximum penalty by 1. We clip the resulting score at zero, so that PenAcc $\in [0, 1]$ while still penalizing instruction-following failures.

Remark. This metric suite assesses (i) whether the output length matches the required prediction count (Len-F1), (ii) whether all predicted labels are valid (UnkLbl%), and (iii) whether the predictions themselves are correct (Acc). PenAcc consolidates these requirements into a single indicator of predictive competence under instruction-following constraints. An ideal model achieves Acc = PenAcc; any non-zero penalty $\Delta_{\text{acc}} = \text{Acc} - \text{PenAcc} > 0$ indicates an output-format or label-validity violation.

Decision Faithfulness Metrics. Decision faithfulness is evaluated by comparing a model's *stated decision factors* with its *measured behavioral reliance*. We use two complementary attribution signals:

- **Self-claimed Attribution.** Given the same context, the model is prompted to produce a ranking π_{self} over the m features, indicating which attributes it believes are most important for predicting the target variable. This ranking captures the model's stated account of its decision factors.
- **Leave-Any-Out (LAO) Attribution.** For each feature $j \in \{1, \ldots, m\}$, we re-evaluate the model under identical prompting while removing that feature from all rows:

$$\Delta_j = \text{Perf}(\mathcal{D}) - \text{Perf}\big(\mathcal{D} \setminus x_{[:,j]}\big),$$

where Perf is the same predictive metric used for predictive competence, such as Accuracy, Macro-F1, or PenAcc. This yields an attribution vector $\boldsymbol{\Delta} = (\Delta_1, \ldots, \Delta_m)$ and an induced ranking π_{LAO}, where larger Δ_j indicates stronger behavioral sensitivity to missing feature j[2]. LAO is used here as a behavioral missing-information intervention. It does not reveal mechanistic causal importance inside the model; rather, it measures how much predictive behavior changes when a named input field is unavailable.

Comparing π_{self} and π_{LAO} assesses the extent to which the model's stated decision factors faithfully reflect its measured behavioral reliance. We then formalize the following metrics for global decision faithfulness.

- **Self-Attribution Recall (SelfAtt@ k)**: This metric measures how well the model's self-reported important features cover the valid feature set. Specifically, given the ground-truth feature set $\mathcal{S}_m$ and $\text{Top}_k(\pi_{\text{self}})$ as the first k distinct features in π_{self}, then SelfAtt@ k is defined as:

$$\textsc{SelfAtt@}k \;=\; \frac{|\mathcal{S}_m \cap \text{Top}_k(\pi_{\text{self}})|}{|\mathcal{S}_m|}, \quad k = |\mathcal{S}_m| \text{ by default.}$$

- **Self-Faith: Global decision faithfulness (ρ)**: This metric measures the agreement between the model's self-claimed attribution ranking and its measured behavioral reliance on features. Specifically, we calculate Spearman's rank correlation coefficient (ρ) between the behavioral ranking derived from LAO scores (π_{LAO}) and the self-claimed ranking (π_{self}). A higher agreement between these rankings indicates how closely the model's stated feature ranking matches its measured behavioral reliance.

$$\text{Spearman's } \rho \;=\; 1 - \frac{6}{m(m^2-1)} \sum_{i=1}^{m} (r_i - s_i)^2,$$

  where r_i and s_i denote the respective ranks of feature i in π_{self} and π_{LAO}. p-values for ρ (and τ) can be obtained through permutation tests. Kendall's τ (rank correlation) serves as a complementary rank-agreement measure.

- **LAO Magnitude (σ_{LAO})**: This metric captures the dispersion of the model's behavioral reliance across features, reflecting the concentration and interpretability of its decision rationale. It is computed as the standard deviation of the LAO performance changes across all features:

$$\sigma_{\text{LAO}} \;=\; \sqrt{\frac{1}{m-1} \sum_{j=1}^{m} (\Delta_j - \bar{\Delta})^2}, \quad \bar{\Delta} = \frac{1}{m} \sum_{j=1}^{m} \Delta_j,$$

  where $\boldsymbol{\Delta} = (\Delta_1, \ldots, \Delta_m)$. A small σ_{LAO} indicates that feature effects are evenly distributed, suggesting reliance on many weak signals. A large σ_{LAO} indicates concentrated behavioral sensitivity to a smaller subset of features, which may be easier to inspect, but does not by itself imply that the relied-upon features are semantically appropriate.

4.1 Integrated View

Taken together, **Comprehension Fidelity**, **Predictive Competence**, and **Global Decision Faithfulness** allow us to map model behavior into distinct regimes of structured decision performance:

- **Accurate & Faithful:** The model produces accurate predictions and its stated decision factors align with behaviorally measured reliance.
- **Accurate & Unfaithful:** The model predicts well, but its stated rationale is misaligned with measured reliance, suggesting unreliable self-attribution.
- **Inaccurate & Faithful:** The model's stated factors align with measured reliance, but the resulting decision rule does not yield accurate predictions.
- **Inaccurate & Unfaithful:** The model neither predicts accurately nor provides self-attributions aligned with its measured behavior.

[2]Ablation can target individual features or pre-defined feature groups to capture higher-order interactions.

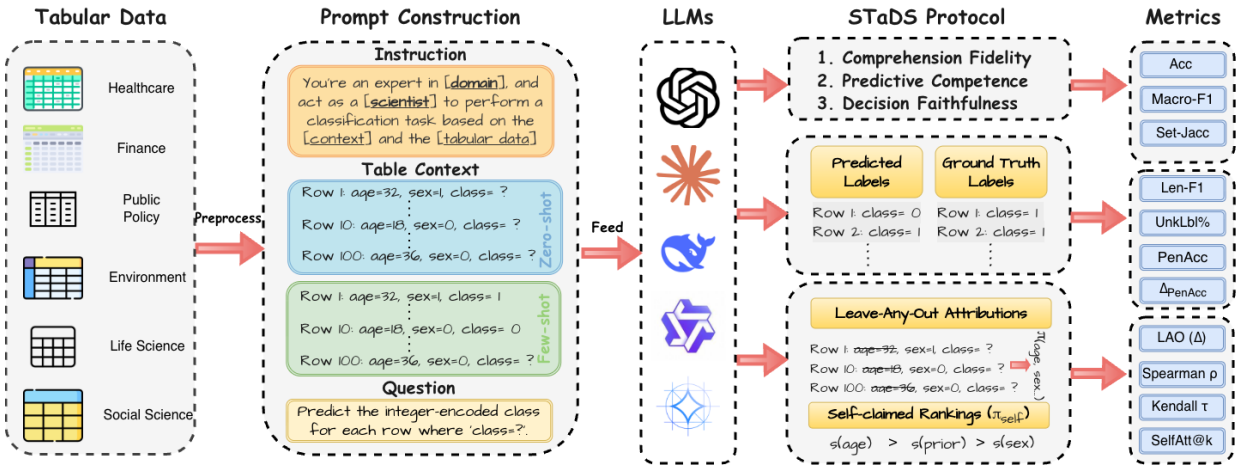


Figure 2: Overview of STaDS. Tabular data from different domains are serialized into structured prompts for LLMs. The model predicts labels for masked rows and is evaluated along three dimensions of structured decision behavior: *Comprehension Fidelity*, measured by Len-F1, UnkLbl%, and Δ_{acc}; *Predictive Competence*, measured by Accuracy, Macro-F1, and PenAcc; and *Global Decision Faithfulness*, measured by SelfAtt@k, Self-Faith, and LAO Magnitude.

Dataset	Sample (N)	Feature (m)	Task	Acting Role
Adult Income (Yeh & Lien, 2009)	32561	14	Binary	labour–economist
Breast Cancer (Xin et al., 2022)	277	9	Binary	clinical oncologist
Car Evaluation (Asuncion et al., 2007)	1728	6	Multi	automotive specialist
COMPAS (Jordan & Freiburger, 2015)	6172	25	Binary	criminal risk-assessment analyst
Congressional Voting (con, 1987)	232	16	Binary	legislative political scientist
Gaussian Synthetic (Agarwal et al., 2022)	5000	20	Binary	applied statistician
German Credit (Asuncion et al., 2007)	1000	20	Binary	bank credit risk analyst
Give Me Some Credit (Freshcorn, 2022)	102209	10	Binary	consumer credit risk analyst
Framingham Heart (World Health Organization, 2021)	3658	15	Binary	cardiovascular epidemiology analyst
HELOC (FICO) (Holter et al., 2018)	9871	23	Binary	home-equity lending risk analyst
Iris (Unwin & Kleinman, 2021)	150	4	Multi	botanical data analyst
Monk 1 / 2 / 3 (Thrun, 1991)	432	6	Binary	data analyst
Pima Diabetes (Smith et al., 1988)	768	8	Binary	diabetes researcher

Table 3: Summary of the tabular decision datasets used in STaDS. We report sample size (N), feature count (m), task type, and the professional role used to contextualize the prompt. Evaluation is based on output validity, predictive performance, and global decision faithfulness.

5 STaDS Benchmark & Experimental Setup

We instantiate STaDS on a suite of tabular decision domains and evaluate each model under a unified prompting, prediction-extraction, and attribution protocol. The setup is designed to support three goals: (i) measuring predictive competence under repeated structured decisions, (ii) comparing self-reported decision factors with behaviorally measured reliance, and (iii) testing whether the resulting conclusions are robust to formatting, post-processing, and perturbation choices. Fig. 2 summarizes the overall pipeline.

Benchmark datasets. We evaluate STaDS on 15 tabular classification datasets spanning healthcare, finance, public policy, synthetic rule-based tasks, and general structured prediction domains. The datasets include both binary and multi-class classification tasks, with sample sizes ranging from 150 to over 100,000 instances and feature counts ranging from 4 to 25. Table 3 summarizes the datasets, task types, feature counts, and role prompts used to contextualize each decision setting. For few-shot evaluation, each dataset is stratified into an 80% training split and a 20% test split. Training rows are used only as in-prompt demonstrations, while test rows provide masked instances for evaluation. In both zero-shot and few-shot settings, only rows whose labels are explicitly masked as `class=?` are queried and scored. The remaining rows serve as context or demonstrations depending on the prompt setting.

LLMs & Hardware. We evaluate nine general-purpose LLMs spanning open-weight and commercial systems. The open-weight models include Llama3-8B-Instruct, Llama3-3B, Mistral-7B-Instruct-v0.3, DeepSeek-Llama-8B, Qwen3-8B, and Gemma-1B/4B-it (Dubey et al., 2024; Jiang et al., 2023; DeepSeek, 2024; Yang et al., 2025; Team et al., 2025). These models are run on local GPU infrastructure, including 8×NVIDIA RTX 3090, GH200, and 8×A100 nodes. The commercial baselines, Gemini-2.5-Pro and GPT-4.1-mini, are accessed through their public APIs (Achiam et al., 2023; Team et al., 2023). Unless otherwise stated, all models are decoded with temperature 0.2, top-$p = 1.0$, and a maximum generation length of 8192 tokens.

To assess whether domain specialization changes the accuracy–faithfulness relationship, we additionally include a medical expert-model case study using `google/medgemma-4b` on healthcare datasets. This case study is reported separately because its purpose is not to expand the main leaderboard, but to test whether domain-specific supervision improves global decision faithfulness rather than only predictive plausibility.

5.1 Prompt Construction & Serialization

We use a single deterministic prompt template for all datasets and models. The template follows an **instruction–input–question–response structure**, inspired by structured prompting for tabular LLMs (Zhang et al., 2023). We use a key=value **serialization** because STaDS requires feature identities to be explicit and stable: individual columns must be removable for behavioral interventions, and model self-attributions must be mapped unambiguously back to input attributes. This representation is therefore not intended as an optimized tabular encoding, but as a controlled interface for decision-factor evaluation. Detailed prompt construction and be found in Appendix A.2.

```
Below is an instruction that describes a task, paired with an input table that provides
further context.
Write a response that appropriately completes the request.
⟨Instruction⟩
⟨Input⟩
⟨Question⟩
⟨Response⟩
```

5.2 Self-attribution & Behavioral Reliance Protocol.

To elicit the self-attribution ranking (π_{self}), we provide the full table and ask the model to order all input features by their importance for predicting the target variable.

Each prompt explicitly enforces a strict output format: a single comma-separated line listing all valid feature names in descending order of importance, with the target label `class` excluded. This reduces ambiguity and prevents additional text such as numbering, bullets, or explanations. LLMs might be prompted as follows to rank all the features:

> *Rank all the features in order of their importance for predicting the target variable, from most important to least...*

We expect the model to return a list of feature names. For instance,

```
[attribute_3, attribute_1, attribute_2, ..., attribute_7]
```

6 Results & Discussion

We evaluate LLM behavior along the three dimensions defined by STaDS: comprehension fidelity, predictive competence, and global decision faithfulness. We treat all metrics as behavioral proxies, characterizing observable model behavior under the STaDS protocol rather than directly revealing internal cognition.

6.1 Instruction Fidelity: Models Differ in Their Ability to Follow Structured Output Constraints

We first evaluate whether models follow the basic output constraints required by STaDS. Fig. 3 reports $\Delta_{\text{acc}} = \text{Acc} - \text{PenAcc}$ across datasets in two prompting settings. Larger values indicate that raw accuracy is reduced after penalizing length mismatch or invalid labels, and therefore reflect failures of instruction fidelity rather than prediction errors alone.

Dataset	Tokens (100 rows)	Zero-shot Acc	Zero-shot Macro-F1	Best Z Model	Few-shot Acc	Few-shot Macro-F1	Best F Model
Adult Income	10K	0.700	0.688	Gemini-2.5-Pro	0.737	0.708	Gemini-2.5-Pro
Breast Cancer	17K	0.729	0.524	GPT-4.1-mini	0.732	0.525	GPT-4.1-mini
Car Evaluation	16K	0.419	0.243	Gemini-2.5-Pro	0.600	0.616	Gemini-2.5-Pro
COMPAS	39K	0.816	0.810	Gemini-2.5-Pro	0.716	0.714	Gemini-2.5-Pro
Congression Vote	36K	0.534	0.348	Llama3-3B	0.638	0.636	Gemini-2.5-Pro
Gaussian Synthetic	48K	0.550	0.448	Gemini-2.5-Pro	0.880	0.873	GPT-4.1-mini
German Credit	13K	0.660	0.616	GPT-4.1-mini	0.889	0.862	Gemini-2.5-Pro
Give Me Some Credit	14K	0.830	0.832	DeepSeek-Llama-8B	0.917	0.916	GPT-4.1-mini
Heart Disease	12K	0.640	0.614	GPT-4.1-mini	0.700	0.697	GPT-4.1-mini
HELOC	25K	0.670	0.670	Gemini-2.5-Pro	0.885	0.883	Gemma-4B
Iris	7K	0.787	0.787	Gemini-2.5-Pro	1.000	1.000	Gemini-2.5-Pro
Monk 1	18K	0.620	0.613	Gemini-2.5-Pro	0.759	0.758	Qwen3-8B
Monk 2	18K	0.674	0.409	DeepSeek-Llama-8B	0.713	0.601	Qwen3-8B
Monk 3	18K	0.579	0.596	Gemini-2.5-Pro	0.644	0.642	Gemini-2.5-Pro
Pima Diabetes	31K	0.758	0.784	Gemini-2.5-Pro	0.820	0.814	Gemini-2.5-Pro

Table 4: Prediction results on tabular benchmarks. We report the best accuracy and macro-F1 across models for zero-shot and few-shot settings, along with the model achieving the best score and the approximate prompt token count when querying 100 test rows.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.17	NaN	0.00
Gemini-2.5-Pro	0.07	0.25 (0.38)	1.00
DeepSeek-Llama-8B	0.24	0.24 (0.41)	1.00
Llama3-8B	0.00	−0.05 (0.89)	0.73
Qwen3-8B	0.11	−0.17 (0.67)	0.44
GPT-4.1-mini	0.01	−0.02 (0.96)	1.00
Llama3-3B	0.11	−0.34 (0.24)	1.00
Mistral-7B	0.01	−0.54 (0.08)	0.73

Table 5: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Adult Income Dataset. NaN indicates π_{self} empty.

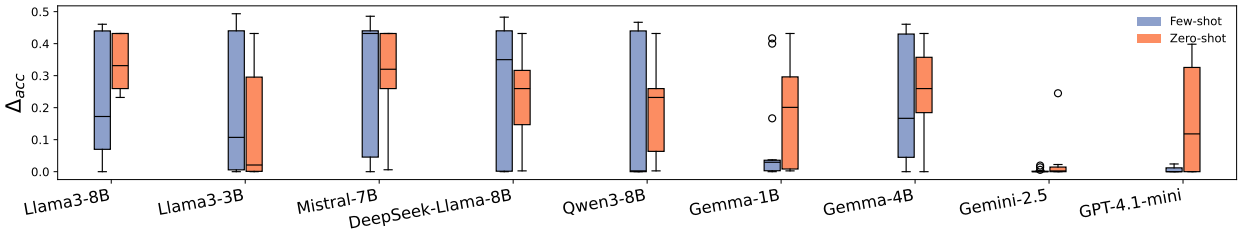
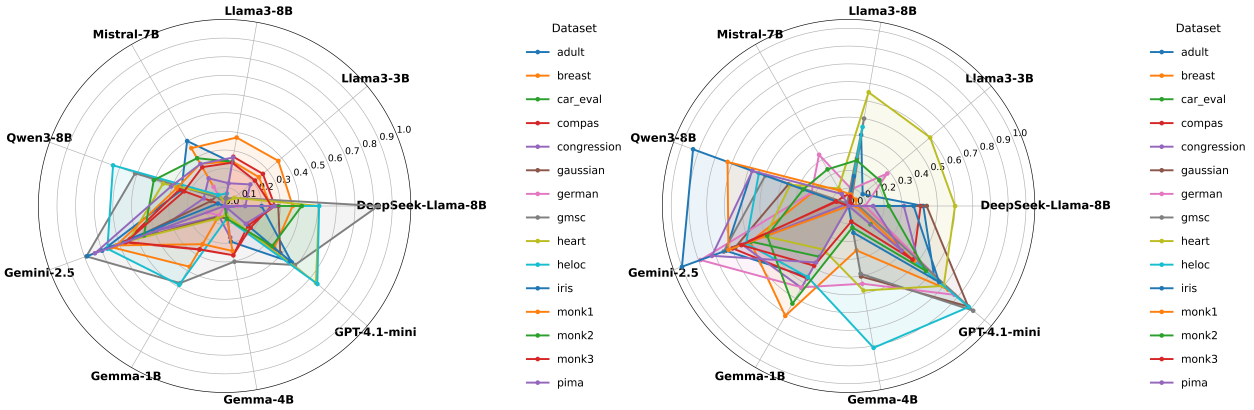


Figure 3: Box plots illustrate the distribution of $\Delta_{\text{acc}} = \text{Acc} - \text{PenAcc}$ for each model across all benchmark datasets. Blue and orange correspond to few-shot and zero-shot settings, respectively. Frontier models cluster near zero Δ_{acc}, while several open-source checkpoints incur format penalties, especially in few-shot setting, indicating heightened prompt sensitivity.

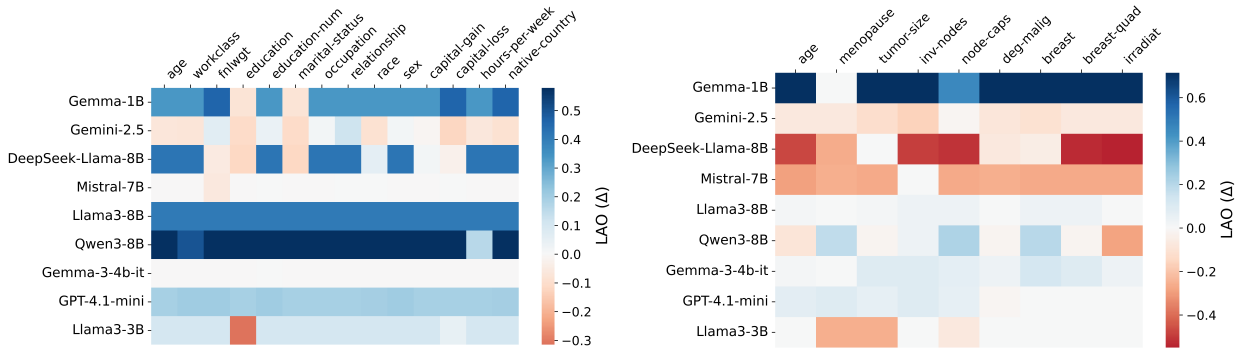
Instruction-following failures remain common in structured decision prompts. As shown in Fig. 3, stronger instruction-tuned models, particularly `Gemini-2.5-Pro`, tend to have near-zero Δ_{acc} across most datasets, indicating that their outputs usually satisfy the required format and label constraints. By contrast, several smaller open-weight models such as `Llama3-8B` incur substantial penalties, often due to over-production, under-production, or invalid labels. These failures are important because STaDS requires repeated ordered predictions: a model that skips rows, changes the output order, or produces labels outside the valid set cannot be evaluated as a reliable repeated decision-maker.



(a) Zero-shot across benchmark datasets

(b) Few-shot across benchmark datasets

Figure 4: Spider plots of Penalized Accuracy ($\alpha = 0.5, \beta = 0.5$) across models and datasets in (a) zero-shot and (b) few-shot settings. Each axis is a model; each colored trace is a dataset. Higher values indicate stronger accuracy and instruction-following. Few-shot generally inflates the polygons (with higher PenAcc) across datasets, with Gemini-2.5-Pro showing the most uniform gains.



(a) Adult Income

(b) Breast Cancer

Figure 5: Heatmap of LAO performance (Δ_{LAO}) for each feature (columns) and LLM (rows). Darker blue indicates a larger performance loss when the feature is removed (higher importance); red indicates a slight performance gain or negligible reliance. A few features dominate reliance for certain models (deep blue), while others spread reliance diffusely, consistent with their σ_{LAO}.

Format adherence is distinct from feature-ranking validity. One might expect that long or verbose outputs are the primary cause of instruction violations, self-attribution introduces a second output-control challenge: the model must return valid feature names rather than free-form explanations Tam et al. (2024). Table 5 illustrates this issue on Adult Income. Some models produce rankings that cover the expected feature set, whereas others omit or hallucinate feature names. We therefore report SelfAtt@k as a validity check for the self-attribution step, not as a direct measure of predictive competence or decision faithfulness.

6.2 Predictive Competence: Accuracy Varies Across Domains and Prompting Regimes

We next evaluate whether models produce correct labels for masked rows. Table 4 reports the best-performing model for each dataset in zero-shot and few-shot settings, and Fig. 4 summarizes PenAcc across model–dataset pairs. This reveals whether the model has failed to ground prior knowledge in the domain or apply it effectively in a tabular decision setting.

Context length alone does not explain predictive performance. While existing research commonly suggests that shorter contexts facilitate better model performance Liu et al. (2024), our results indicate complexity

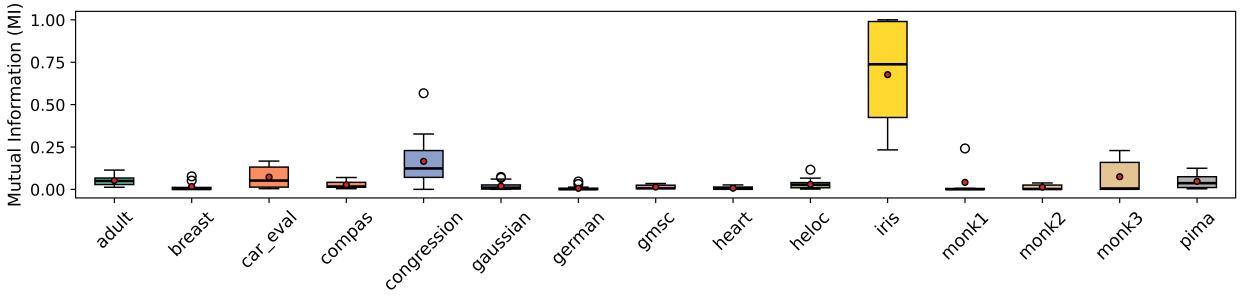


Figure 6: Summary of the distribution of normalized mutual information (NMI) by dataset; Overall, NMI magnitudes are low, indicating that raw dataset co-occurrence cannot account for decision faithfulness.

beyond mere input length. For instance, `Gemini-2.5-Pro` achieves high accuracy (0.81) on the COMPAS dataset (39K tokens) yet shows significantly lower performance (∼0.4) on the shorter Car Evaluation dataset. This discrepancy, presented in Table 4 and Fig. 4, may arise from the multi-class nature of the Car Evaluation task, highlighting that *context length alone does not fully explain model performance variations*. However, the multi-class nature from Iris does not account for this across all models.

More broadly, performance analyses in (Fig. 4, panels (a) and (b)) demonstrate that models tend to achieve better results on intermediate-sized contexts, (e.g., Breast Cancer: 17K tokens; Iris: 7K tokens; Heart Disease: 12K tokens) and less consistently on very long contexts (COMPAS, Pima Diabetes). These observations suggest a complex interaction between context length, task complexity, and model capability, cautioning against interpreting token count alone as a reliable predictor of competence.

Acting as a general professional across domains. PenAcc jointly captures a model's ability to *follow formatting rules* and *produce the correct label*, so it naturally reflects whether an LLM can "act like a well-trained professional" across diverse structured tabular settings. Fig. 4 overlays PenAcc in radar form for every model–dataset pair, while Table 4 reports the best zero/few-shot scores. `Gemini-2.5-Pro` dominates, topping 10/15 datasets and exhibiting almost identical polygons in the zero/few-shot plots, evidence of strong domain generalisation. Open-source checkpoints tell a more fragmented story: compact models such as `Llama3-3B` or `Gemma-1B` shine on small medical tables (e.g. Heart Disease) yet collapse on wide-schema sets like COMPAS; even larger `Llama3-8B` and `Qwen3-8B` sometimes fail simply because they mis-parse column headers or drift from the required output format rather than because they "do not know" the task.

Few-shot demonstrations improve performance but do not eliminate domain variability. We also compare zero-shot and few-shot performance to ask how much "being shown how to behave" helps a model act like that professional. Zero-shot in STaDS is effectively pure retrieval and schema inference: *the model must infer label semantics, column roles, and decision logic from the instruction and table alone*. Few-shot augments this with a handful of in-context demonstrations, i.e., explicit exemplars of how a domain expert would label similar rows. Across all 15 datasets, providing these demonstrations consistently improves the *best achievable* PenAcc (see Table 4), which is an upper-bound view (see detailed results in Appendix Tables 18 - 33). Averaged over all datasets, the best few-shot PenAcc exceeds the best zero-shot PenAcc by an absolute +0.15, corresponding to a 27% relative improvement. This pattern holds even for datasets where zero-shot performance is already strong, such as Iris improved from 0.79 to 1.00, and Give Me Some Credit improved from 0.83 to 0.92. In other words, demonstrations do not just rescue weaker models, and they sharpen already competent ones.

Answer to RQ1. Frontier LLMs exhibit *partial* professional-level competence out of the box: the strongest models can often produce valid, well-formatted predictions across diverse decision settings, but this behavior is not yet consistent across domains or model families. Demonstrations remain crucial, indicating that current models still require task-specific guidance rather than universally understanding tabular decision rules in a zero-shot setting.

6.3 Global Decision Faithfulness: Accuracy and Stated Reliance Often Diverge

We now examine whether models' stated decision factors align with behaviorally measured reliance. We use LAO Magnitude (σ_{LAO}), Self-Faith (ρ), and SelfAtt@k as behavioral proxies. Results are shown in Fig. 5, Fig. 7, Fig. 6, Table 6, and Appendix Figs. 15–16.

Figure 7: Penalized Accuracy vs. Self-Decision Faithfulness Across All Datasets. Each subplot shows penalized accuracy (x-axis) against self-faith (y-axis), where self-faith defined by (Spearman's ρ) between model-declared and behavioral (LAO) feature rankings. Zero-shot and few-shot results are shown with circles and squares, respectively. Models are colored constantly. Accuracy and faithfulness diverge for many model and dataset pairs (high-PenAcc with low ρ), i.e., accurate yet globally unfaithful.

Models do not always rely on the factors they report. Across datasets, we observe clear discrepancies between self-reported feature importance and behavioral reliance measured by LAO. On Adult Income, Table 5 shows that some models have positive but modest self–LAO alignment, while others have negative SELF-FAITH. For example, `Gemini-2.5-Pro` shows positive alignment, whereas `Mistral-7B` and `Llama3-3B` exhibit negative correlations. This indicates that the ranking a model reports can diverge from the ranking induced by perturbing features in the input. Fig. 5 further illustrates these differences. Some models show concentrated sensitivity to a small number of interpretable features, while others distribute sensitivity broadly or inconsistently across many columns. We interpret these heatmaps as behavioral reliance profiles, not as mechanistic explanations of internal model computation.

Fig. 7 plots PenAcc against SELF-FAITH. The resulting regimes show that predictive competence and global decision faithfulness can come apart. Some model–dataset pairs are both accurate and relatively faithful, such as `Gemini-2.5-Pro` on Iris. Others are accurate but weakly faithful, such as `GPT-4.1-mini` on Breast Cancer, where predictive performance is strong but self–LAO alignment remains low. Conversely, some models exhibit moderate self–LAO alignment despite limited predictive performance, indicating that consistency between stated and measured reliance is not sufficient for correctness. These regimes support the central claim of STaDS: accuracy alone does not determine whether a model's stated decision factors match its measured decision behavior.

Statistical dependence does not account for behavioral reliance. To separate behavioral reliance from dataset-level association, we compare model attributions with feature–label dependencies measured by normalized mutual information (NMI), as shown in Fig. 6. Most datasets exhibit low average feature–label dependence, suggesting that predictive performance cannot be explained by a single strongly correlated feature alone. High-dependence tasks such as Iris naturally support stronger in-context generalization, but this pattern does not hold uniformly across all datasets. Importantly, statistical association is not equivalent to behavioral reliance. A feature may be strongly associated with the label in the dataset while having little effect on a model's prediction when perturbed; conversely, a model may be behaviorally sensitive to a feature with weak marginal association. We therefore use NMI as a dataset-level reference signal, not as ground-truth causal structure.

Self-attributions align more with statistical association than with behavioral reliance. We triangulate three rankings: the model's self-attribution ranking π_{self}, the behavioral LAO ranking π_{LAO}, and the statistical feature–label dependence ranking π_{NMI}. Across datasets, Table 6 and Appendix Table 35 show that

$$\rho(\pi_{\text{self}},\, \pi_{\text{NMI}}) \;>\; \rho(\pi_{\text{LAO}},\, \pi_{\text{NMI}})$$

holds in the majority of model–dataset pairs where at least one correlation exceeds 0.2 in absolute value (12 of the 18 reported cases). Two alternative readings deserve mention. First, π_{NMI} may itself be a noisy importance proxy. Second, LAO can underestimate reliance when models compensate through correlated features; group ablations in Sec. 6.4 discusses this partially. We therefore treat this triangulation as diagnostic rather than definitive. This suggests that models tend to report rankings that reflect what *should* matter statistically rather than what measurably affects their predictions under controlled input removal, indicating a gap that gives an overly optimistic impression of decision faithfulness when self-report alone is used as the evaluation signal.

Answer to RQ2. Current LLMs frequently exhibit a gap between stated and behaviorally measured decision factors. Even when PenAcc is high, self-attributions can align more closely with dataset-level statistical associations than with LAO-based behavioral reliance. Under STaDS, current models therefore show partial global decision faithfulness: they can be accurate and produce plausible feature rankings, but their stated decision factors do not reliably match the factors that affect their predictions under intervention.

6.4 Robustness and Validation Analyses

We conduct additional analyses to test whether the main findings depend on specific methodological choices. These analyses are summarized here, with full results reported in Appendix B.

Deletion vs. perturbation operators. In classical model-centric ML, perturbation operators are often treated as alternative ways to disrupt the association between a feature and a trained model's prediction. We therefore compare deletion-LAO with four alternative perturbation operators: constant replacement, mean replacement, empirical marginal sampling, and column-wise permutation (full details in Appendix B.1). Table 7 summarizes this operator dependence. Agreement between alternative operators and deletion-LAO varies substantially across datasets and models. For example, Iris–GPT-4.1-mini shows high agreement under constant replacement ($\rho = 0.95$), but negative agreement under permutation ($\rho = -0.40$) and marginal sampling ($\rho = -0.80$). Similarly, Breast–Qwen3-8B changes from positive agreement under constant replacement ($\rho = 0.53$) to negative agreement under marginal sampling

Table 6: Triangulated faithfulness across selected models and datasets. We compare rank agreement between model self-attribution (π_{self}), LAO-based behavioral reliance (π_{LAO}), and statistical feature–label dependence (π_{NMI}). Brackets show p-values; * denotes $p < .05$ and † denotes $p < .10$.

Dataset	Model	$\rho(\pi_{\text{self}}, \pi_{\text{LAO}})$	$\rho(\pi_{\text{self}}, \pi_{\text{NMI}})$	$\rho(\pi_{\text{LAO}}, \pi_{\text{NMI}})$
Adult Income	Gemma-4B	$0.552^\dagger$ [0.098]	0.394 [0.260]	0.547^* [0.043]
	Gemini-2.5-Pro	0.253 [0.383]	$0.477^\dagger$ [0.085]	0.187 [0.523]
	DeepSeek-Llama-8B	0.240 [0.409]	0.301 [0.296]	-0.187 [0.523]
	GPT-4.1-mini	-0.015 [0.958]	0.240 [0.409]	-0.618^* [0.019]
Breast Cancer	DeepSeek-Llama-8B	0.810^* [0.015]	0.586 [0.127]	0.009 [0.982]
	GPT-4.1-mini	0.217 [0.576]	0.775^* [0.014]	0.000 [1.000]
Car Evaluation	Gemini-2.5-Pro	0.657 [0.156]	0.886^* [0.019]	0.314 [0.544]
	GPT-4.1-mini	-0.600 [0.208]	0.657 [0.156]	-0.943^* [0.005]
	Qwen3-8B	0.543 [0.266]	0.257 [0.623]	0.429 [0.397]
COMPAS	Mistral-7B	0.881^* [0.004]	0.095 [0.823]	0.285 [0.425]
	Llama3-8B	0.033 [0.932]	-0.417 [0.265]	0.406 [0.244]
Iris	Gemini-2.5-Pro	0.800 [0.200]	1.000^* [0.000]	0.800 [0.200]
	GPT-4.1-mini	0.800 [0.200]	1.000^* [0.000]	0.800 [0.200]
Monk 1	DeepSeek-Llama-8B	0.200 [0.704]	0.676 [0.140]	0.845^* [0.034]
	Mistral-7B	1.000^* [0.000]	0.500 [0.667]	-0.068 [0.899]
Pima Diabetes	DeepSeek-Llama-8B	0.952^* [0.000]	-0.286 [0.493]	-0.286 [0.493]
	GPT-4.1-mini	-0.095 [0.823]	0.905^* [0.002]	-0.333 [0.420]
	Qwen3-8B	-0.429 [0.289]	0.857^* [0.007]	-0.286 [0.493]

Table 7: Perturbation-operator sensitivity across representative datasets and models. Rank agreement reports Spearman correlation ρ between the feature ranking induced by each alternative operator and the deletion-LAO ranking. Positive-effect rate reports the fraction of features for which perturbation improves predictive performance ($\Delta > 0$).

		Rank agreement with deletion-LAO				Positive-effect rate ($\Delta > 0$)				
Dataset	Model	Const.	Mean	Perm.	Marginal	Drop	Const.	Mean	Perm.	Marginal
Breast	Qwen3-8B	+0.53	-0.29	+0.22	-0.69	55.6%	66.7%	88.9%	66.7%	55.6%
Breast	GPT-4.1-mini	+0.05	+0.17	+0.41	+0.42	44.4%	11.1%	33.3%	44.4%	55.6%
Iris	Qwen3-8B	-0.40	+0.20	+0.00	+0.60	25.0%	75.0%	75.0%	50.0%	75.0%
Iris	GPT-4.1-mini	+0.95	+0.45	-0.40	-0.80	50.0%	0.0%	100.0%	25.0%	50.0%

($\rho = -0.69$). This distinction is particularly important for prompted LLMs. We interpret this as the perturbation changes the serialized input distribution, the visible schema, and the local decision context in prompted LLMs. Consequently, different operators need not recover the same feature-reliance ranking. These operators can be seen as instantiating different counterfactuals. Deletion completely removes a named field from the prompt; constant and mean replacement keep the feature slot visible while suppressing or substituting its value; marginal sampling preserves univariate plausibility while resampling values; and permutation preserves the empirical marginal distribution.

Because repeated LLM generations can vary even under the same prompt, single-run perturbation effects may conflate intervention impact with ordinary sampling variation. We therefore conduct a targeted distribution-based perturbation check using GPT-4.1-mini, avoiding additional parsing issues from smaller models with weaker format adherence. For each selected feature, we sample 30 generations from the unperturbed prompt and 30 generations from the feature-deleted prompt at temperature 0.1, and compare prediction-vector distributions using disagreement distance. As shown in Table 8, feature-deleted conditions have within-condition disagreement comparable to their matched baselines, while cross-condition distances exceed within-condition variability. This shows statistically reliable distributional effects for all tested deletion conditions ($p < 0.001$), indicating that deletion shifts the model's output distribution beyond ordinary sampling variation without simply making generations noisier. By contrast, column-wise permutation can produce larger distributional effects but also higher within-condition variability for informative petal features, reflecting a noisier counterfactual induced by corrupted row-wise alignment. These results support using deletion-LAO as the cleaner and more controlled primary intervention for the STaDS missing-information estimand.

Table 8: Distributional comparison between deletion and column-wise permutation for GPT-4.1-mini on Iris. Each perturbation condition uses 30 generations at temperature 0.1. Within-condition disagreement measures prediction-vector variability among repeated generations under the same condition. E_j measures cross-condition distributional separation beyond within-condition stochasticity. All reported distributional effects are significant under permutation testing ($p < 0.001$).

Feature	Within-base	Within-del.	E_j^{del}	Within-perm.	E_j^{perm}	Stability gap
`petal_length`	0.274	0.303	0.074	0.491	0.128	+0.188
`petal_width`	0.321	0.265	0.046	0.328	0.109	+0.063
`sepal_width`	0.265	0.262	0.067	0.247	0.026	-0.015

Penalty-weight sensitivity of PenAcc. We vary the PenAcc penalty weights over the simplex $\alpha + \beta = 1$ to test whether conclusions depend on the default $\alpha = \beta = 0.5$. The resulting performance-degradation curves vary smoothly, without qualitative reversals in the main findings. This supports the use of equal weighting as a neutral default rather than a tuned hyperparameter. Results are provided in Appendix B.2.

Post-processing audit. Because some raw outputs contain extra text or malformed lists, we audit the post-processing step used for prediction extraction. The audit compares raw extractable predictions with cleaned outputs and measures whether prediction accuracy changes after post-processing. For single-column ablations, all settings remain at 1.00 except Breast–Qwen3-8B, which drops slightly to 0.94. All evaluated multi-column settings remain at 1.00, as shown in Appendix B.3 Table 38. We further examined the small accuracy drop for Breast–Qwen3-8B under single-column ablation. In these cases, the main issue appears to be inconsistency between the generated reasoning trace and the final predicted label, rather than systematic correction by the post-processing model.

Correlated group ablations reveal model-dependent higher-order reliance. Because single-feature LAO can miss joint dependence among correlated predictors, we additionally ablate correlated feature groups for Iris and Breast Cancer. The results show clear but model-dependent higher-order effects. On Iris, Qwen3-8B exhibits super-additive reliance on correlated petal features: the joint removal of `petal_length` and `petal_width` causes a substantially larger drop than their individual removals would predict. GPT-4.1-mini, however, shows negative interactions on the same groups, suggesting redundancy rather than synergy. On Breast Cancer, GPT-4.1-mini shows positive non-additive effects for both pair and triple removals, whereas Qwen3-8B shows weaker and more heterogeneous interactions. Thus, group ablations confirm that single-feature LAO can underestimate higher-order reliance, but also demonstrate that such reliance is model- and dataset-specific. We use group ablations as a complementary stress test (see Appendix B.4 for more details).

6.5 Case study: Domain-specialized Medical Model

To test whether domain specialization reduces the accuracy–faithfulness gap, we evaluate `google/medgemma-4b` on two healthcare datasets, *Breast Cancer* and *Pima Diabetes*, and compare it with the general-domain Gemma3-4B baseline from the same family. For each model, we measure behavioral reliance using deletion-LAO and compare it with the model's self-reported feature ranking. Fig. 8 and Table 40 in Appendix B.5 show that domain specialization does not automatically yield global decision faithfulness in this case study. On *Pima Diabetes*, Gemma3-4B has an almost flat LAO profile despite producing a non-flat self-ranking, while MedGemma shows stronger behavioral sensitivity but still weak self–LAO alignment. On *Breast Cancer*, MedGemma identifies variables that are medically interpretable in the dataset context such as `inv-nodes`, `tumor-size`, and `deg-malig`, yet its self-reported ranking only weakly matches the measured LAO ranking. These results suggest that domain specialization can improve the apparent domain relevance of self-attributions without ensuring that stated decision factors match behavioral reliance under intervention. This reinforces the need to evaluate expert-domain LLMs not only by predictive performance or plausible explanations, but also by global decision faithfulness.

6.6 Conclusion & Limitation

This work introduced the STaDS protocol, a principled framework for evaluating structured decision competence and global decision faithfulness in LLMs. STaDS goes beyond predictive accuracy by testing whether models follow task specifications, make repeated decisions across structured domains, and align their self-reported decision factors with perturbation-based behavioral reliance. Across 9 LLMs and 15 tabular decision domains, we find that accuracy and

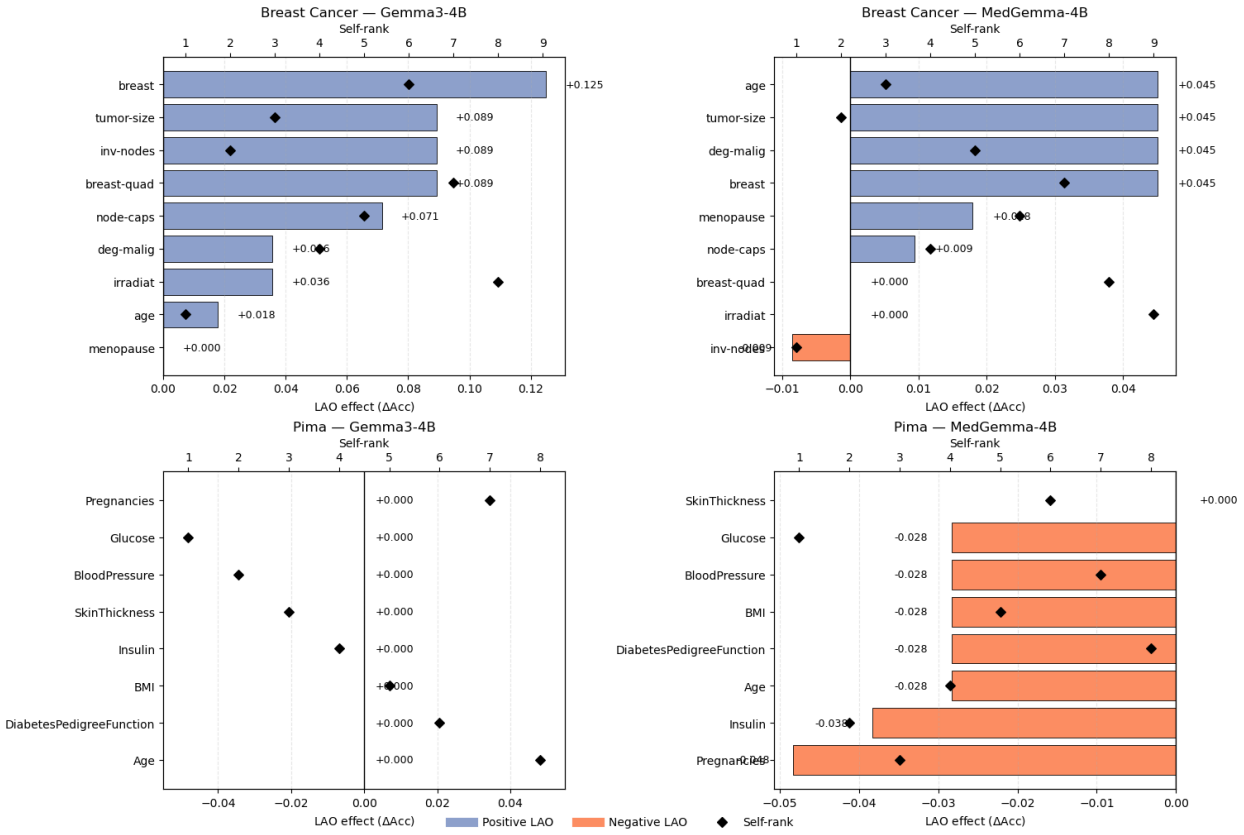


Figure 8: Comparison between the general-domain Gemma3-4B and domain-specialized MedGemma-4B on *Breast Cancer* and *Pima Diabetes*. Bars show the LAO effect of removing each feature, while black diamonds show the model's self-reported importance rank.

faithfulness are separable. Stronger models often achieve reasonable predictive performance, especially with few-shot demonstrations, but their stated feature rankings frequently diverge from LAO-based reliance. These findings suggest that global decision faithfulness, not accuracy alone, should serve as a necessary evaluation criterion when deploying LLMs in structured settings where the reliability of stated rationales carries real consequences.

Limitations & future work. Several limitations remain. First, STaDS does not provide ground-truth access to a model's internal decision process. LAO and related perturbation methods measure behavioral sensitivity under specified interventions, not mechanistic causal reliance. Accordingly, self–LAO disagreement should be interpreted as evidence that stated feature importance does not match measured behavior under the chosen intervention, rather than as a complete account of the model's internal computation. Future work could combine STaDS with mechanistic interpretability, activation-level analysis, or controlled synthetic tasks where the true decision rule is known.

Second, feature interactions remain only partially captured. Single-feature LAO provides a simple and interpretable first-order diagnostic, but correlated predictors and higher-order dependencies can make reliance distributed across feature groups. Our group-ablation analyses show that such interactions can be super-additive, redundant, or model-dependent. Scaling this analysis to larger feature sets is computationally challenging, since the number of possible feature groups grows combinatorially. Future work should develop more efficient interaction-search procedures, structured group-ablation strategies, and principled ways to separate redundancy from genuine joint reliance.

Finally, some benchmark datasets, such as Iris and Adult Income, are public and may have appeared in pre-training corpora. STaDS is therefore not intended to isolate memorization-free predictive competence on these datasets. However, the central analysis concerns the relationship between predictive behavior, self-attribution, and perturbation-based reliance. Even when dataset exposure may improve accuracy, it does not by itself explain why self-reported feature rankings diverge from measured behavioral reliance. Future work should extend STaDS to private, newly

collected, or synthetic rule-controlled datasets to more directly separate memorization from structured decision competence.

References

Congressional Voting Records. UCI Machine Learning Repository, 1987. DOI: https://doi.org/10.24432/C5C01P.

Alaa Abd-Alrazaq, Rawan AlSaad, Dari Alhuwail, Arfan Ahmed, Padraig Mark Healy, Syed Latifi, Sarah Aziz, Rafat Damseh, Sadam Alabed Alrazak, and Javaid Sheikh. Large language models in medical education: opportunities, challenges, and future directions. *JMIR medical education*, 9(1):e48291, 2023.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018.

Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. Openxai: Towards a transparent evaluation of model explanations. *Advances in neural information processing systems*, 35:15784–15799, 2022.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

Sajid Ali, Tamer Abuhmed, Shaker El-Sappagh, Khan Muhammad, Jose M Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz-Rodríguez, and Francisco Herrera. Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence. *Information fusion*, 99:101805, 2023.

Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-thought reasoning in the wild is not always faithful. *arXiv preprint arXiv:2503.08679*, 2025.

Arthur Asuncion, David Newman, et al. Uci machine learning repository, 2007.

Fazl Barez, Tung-Yu Wu, Iván Arcuschin, Michael Lan, Vincent Wang, Noah Siegel, Nicolas Collignon, Clement Neo, Isabelle Lee, Alasdair Paren, et al. Chain-of-thought is not explainability. *Preprint, alphaXiv*, pp. v1, 2025.

Carl Bereiter. *Education and mind in the knowledge age*. Routledge, 2005.

John D Bransford, Ann L Brown, Rodney R Cocking, et al. *How people learn*, volume 11. Washington, DC: National academy press, 2000.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31, 2018.

Wei-Lin Chen, Cheng-Kuang Wu, Yun-Nung Chen, and Hsin-Hsi Chen. Self-icl: Zero-shot in-context learning with self-generated demonstrations. *arXiv preprint arXiv:2305.15035*, 2023.

Michelene TH Chi, Paul J Feltovich, and Robert Glaser. Categorization and representation of physics problems by experts and novices. *Cognitive science*, 5(2):121–152, 1981.

Michelene TH Chi, Robert Glaser, and Marshall J Farr. *The nature of expertise*. Psychology Press, 2014.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Arun Das and Paul Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.

DeepSeek. Deepseek llms: Efficient distillation of llama models, 2024. URL https://github.com/deepseek-ai. Technical report.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. Turl: Table understanding through representation learning. *ACM SIGMOD Record*, 51(1):33–40, 2022.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*, 2019.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*, 2019.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pp. arXiv–2407, 2024.

Bryce Freshcorn. Give me some credit :: 2011 competition data. https://www.kaggle.com/datasets/brycecf/give-me-some-credit-dataset, 2022. Accessed: 2025-08-01.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jonathan Herzig, Paweł Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Martin Eisenschlos. Tapas: Weakly supervised table parsing via pre-training. *arXiv preprint arXiv:2004.02349*, 2020.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 01 2025. doi: 10.1038/s41586-024-08328-6. URL https://www.nature.com/articles/s41586-024-08328-6.

Steffen Holter, Oscar Gomez, and Enrico Bertini. Fico explainable machine learning challenge. *FICO COmmunity*, 2018.

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*, 2020.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie–Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. 2023. URL https://arxiv.org/abs/2310.06825.

Kareem L Jordan and Tina L Freiburger. The effect of race/ethnicity on sentencing: Examining sentence type, jail length, and prison length. *Journal of Ethnicity in Criminal Justice*, 13(3):179–196, 2015.

Hazel H Kim. How ambiguous are the rationales for natural language reasoning? a simple approach to handling rationale uncertainty. *arXiv preprint arXiv:2402.14337*, 2024.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

Xuhong Li, Haoyi Xiong, Xingjian Li, Xuanyu Wu, Xiao Zhang, Ji Liu, Jiang Bian, and Dejing Dou. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. *Knowledge and Information Systems*, 64(12):3197–3234, 2022.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

Fuxiao Liu, Paiheng Xu, Zongxia Li, Yue Feng, and Hyemi Song. Towards understanding in-context learning with contrastive demonstrations and saliency maps. *arXiv preprint arXiv:2307.05052*, 2023.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Richard E Mayer. Models for understanding. *Review of educational research*, 59(1):43–64, 1989.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*, 2023.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. In *ITG Symposium on Image Processing*, 2017.

Jack W Smith, James E Everhart, William C Dickson, William C Knowler, and Robert Scott Johannes. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care*, pp. 261, 1988.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 645–654, 2024.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. Let me speak freely? a study on the impact of format restrictions on large language model performance. In Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1218–1236, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-industry.91. URL https://aclanthology.org/2024.emnlp-industry.91/.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Sebastian Thrun. The monk's problems: A performance comparison of different learning algorithems. *Technical Report of Carnegie Mellon University*, 1991.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36: 74952–74965, 2023.

Antony Unwin and Kim Kleinman. The iris data set: In search of the source of virginica. *Significance*, 18, 2021. URL https://api.semanticscholar.org/CorpusID:244763032.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Label words are anchors: An information flow perspective for understanding in-context learning. *arXiv preprint arXiv:2305.14160*, 2023.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, et al. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *arXiv preprint arXiv:2401.04398*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*, 2023.

World Health Organization. Cardiovascular diseases (cvds): fact sheet. `https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)`, 2021. Accessed: 2025-08-01.

Rui Xin, Chudi Zhong, Zhi Chen, Takuya Takagi, Margo Seltzer, and Cynthia Rudin. Exploring the whole rashomon set of sparse decision trees. *Advances in neural information processing systems*, 35:14071–14084, 2022.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

I-Cheng Yeh and Che-hui Lien. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert systems with applications*, 36(2):2473–2480, 2009.

Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. Natural language reasoning, a survey. *ACM Computing Surveys*, 56(12):1–39, 2024.

Mateo Espinosa Zarlenga, Zohreh Shams, Michael Edward Nelson, Been Kim, and Mateja Jamnik. Tabcbm: Concept-based interpretable neural networks for tabular data. 2023.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. Tablellama: Towards open large generalist models for tables. *arXiv preprint arXiv:2311.09206*, 2023.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

Appendix

The appendix provides extended details of our experimental setup and additional results supporting the main text.

A Implementation Details.

We detail the full STaDS prompt template, decoding configuration, and evaluation setup to ensure reproducibility. Each prompt follows a deterministic structure composed of four blocks, as shown in Fig. 9 - 13.

All open-source models (Llama-3, Mistral-7B, Gemma, Qwen3, DeepSeek-R1) were evaluated on an 8×NVIDIA RTX 3090 cluster with temperature = 0.2 and top-p = 1.0. Commercial baselines (Gemini-2.5-Pro and GPT-4.1-mini) were accessed through their official APIs. Outputs were automatically cleaned using GPT-4.1-mini for consistent formatting. Official code will be published soon.

System Prompt:
Below is an instruction that describes a task, paired with an input table that provides further context. Write a response that appropriately completes the request.

Instruction:
Dataset: **{Dataset name}**
Role: You are a **{Role player}**
Task: Your task is to perform **{binary/multi} classification** , predicting {task description}.
Target Encoding: {0: {class_1}, 1: {class_2}}
Attribute Glossary: **attribute_1**: attribute information,..., *** attribute _m*: attribute information.
Class Priors: **0**: 0.51, **1**: 0.5.

Your Job:
- For every row where 'class=?', predict its integer-encoded target relying solely on your pre-trained knowledge.
- Return one integer per row, in the exact same order as the rows appear.
- The number of predictions must equal the number of rows with 'class=?'.

Important:
- Do NOT include any code, code blocks, or explanations of code in your answer.
- Do NOT use or mention sklearn, pandas, or any code-based methods.
- DO NOT output text as target labels, only output the integer-encoded target.

Input:
Row 1: age=2, menopause=2, tumor-size=0, inv-nodes=0, breast=1, breast-quad=2, class=?
⋮
Row 5: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=?
Row 7: age=8, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=?
⋮
Row 9: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=?

Question:
Predict the integer-encoded class for each row where 'class=?'. There are exactly {num prediction} rows with 'class=?' in the input table. Do not output more or fewer predictions than {num prediction}.

Figure 9: **Zero-shot prompt composition.** This setup is designed to evaluate: (i) the model's ability of instruction/table comprehension, and (ii) its intrinsic, pre-trained knowledge in specific role, serving as zero-shot baselines.

A.1 Runtime and computational cost

STaDS requires repeated model inference under unperturbed and perturbed prompts. The total cost therefore scales with the number of datasets, models, prompt settings, features, and perturbation operators. For a dataset with m features, a single deletion-LAO evaluation requires one unperturbed run plus m feature-deleted runs per model and

System Prompt:
Below is an instruction that describes a task, paired with an input table that provides further context. Write a response that appropriately completes the request.

Instruction:
Dataset: **{Dataset name}**
Role: You are a **{Role player}**
Task: Your task is to perform **{binary/multi} classification** , predicting {task description}.
Target Encoding: {0: {class_1}, 1: {class_2}}
Attribute Glossary: **attribute_1**: attribute information,..., *** attribute _m*: attribute information.
Class Priors: **0**: 0.51, **1**: 0.5.

Your Job:
- For every row where 'class=?', predict its integer-encoded target relying solely on your pre-trained knowledge.
- Return one integer per row, in the exact same order as the rows appear.
- The number of predictions must equal the number of rows with 'class=?'.

Important:
- Do NOT include any code, code blocks, or explanations of code in your answer.
- Do NOT use or mention sklearn, pandas, or any code-based methods.
- DO NOT output text as target labels, only output the integer-encoded target.

Input:
Row 1: age=2, menopause=2, tumor-size=0, inv-nodes=0, breast=1, breast-quad=2, class=0
⋮
Row 5: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=1
Row 7: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=?
⋮
Row 9: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=?

Question:
Predict the integer-encoded class for each row where 'class=?'. There are exactly {num prediction} rows with 'class=?' in the input table. Do not output more or fewer predictions than {num prediction}.

Response:
DeepSeek-R1-Llama-8B:
<think>Alright, so I'm trying to figure out how to predict whether a <think>
output:[1, 4, 5, 6, 7, 8, 9, 10]
Gemma-1B:
output:[0, 0, 0, 0, 0, 0, 0, 0,..., 0]
Gemma-4B:
output:[1, 1, 1, 1, 1, 1, 1, 1,..., 1]
Qwen-8B:
<think>Alright, so I'm trying to figure out how to predict whether a <think>
output:
Row 1: age=2, ... inv-nodes=0, breast=1, breast-quad=2, class=0
Row 2: age=8, ...,inv-nodes=0, breast=1, breast-quad=2, class=0
⋮
Row 8: age=8, ...,inv-nodes=0, breast=1, breast-quad=2, class=0
Llama3.2-8B:
output:[0, 0, 0, 0, 0, 0, 0, 0,..., 0]
GPT-4.1-mini:
output:[1, 1, 0, 1, 0, 0, 1, 0]
Gemini-2.5-Pro:
output:[0, 1, 0, 1, 0, 0, 0, 1]

Figure 10: **Few-shot prompt composition.** The setup is designed to evaluate (i) the model's capacity for in-context generalization beyond zero-shot baselines, and (ii) its ability to jointly parse instructions and structured tabular inputs. The bottom panel presents model outputs for the same prediction prompt on the example dataset (Breast Cancer) under the few-shot configuration.

System Prompt:
Below is an instruction that describes a task, paired with an input table that provides further context. Write a response that appropriately completes the request.

Instruction:
Dataset: **{Dataset name}**
Role: You are a **{Role player}**
Task: Your task is to perform **{binary/multi} classification** , predicting {task description}.
Target Encoding: {0: {class_1}, 1: {class_2}}
Attribute Glossary: **attribute_1**: attribute information,..., *** attribute _m*: attribute information.
Class Priors: **0**: 0.51, **1**: 0.5.

Your Job:
- Return a **single comma-separated list** of the most important feature names (attributes) that influence the target variable according to your general understanding of the dataset.
- Only include **original feature names** as listed in the Attribute Glossary.
- Do **not** include reasoning, explanations, or duplicate features.
- The number of features should be appropriate for your global understanding of the task.

Important:
- Do NOT include any code, code blocks, or explanations of code in your answer.
- Do NOT use or mention sklearn, pandas, or any code-based methods.
- DO NOT output text as target labels, only output the integer-encoded target.

Input:
Row 1: age=2, menopause=2, tumor-size=0, inv-nodes=0, breast=1, breast-quad=2, class=0
⋮
Row 5: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=1
Row 7: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=1
⋮
Row 9: age=4, menopause=0, tumor-size=4, inv-nodes=0, breast=1, breast-quad=2, class=0

Question:
Based on your overall understanding of the dataset and the prediction task, rank all the features (**attributes**) in order of their importance for predicting the target variable (**class**), from most important to least important. Your response should be a single, comma-separated list of feature names, ordered by their relevance to the prediction task across the entire dataset.

Response:
DeepSeek-R1-Llama-8B:
<think>Alright, so I'm trying to figure out how to predict whether a <think>
output: tumor-size, inv-nodes, deg-malig, age, menopause, node-caps, irradiat, breast-quad
Gemma-1B:
output: menopause, breast-quad, breast, breast-quad, breast, breast-quad, breast, breast-quad, breast, breast-quad, breast, breast-quad
Gemma-4B:
output: age, inv-nodes, tumor-size, deg-malig, node-caps, breast, breast-quad, irradiat
Qwen-8B:
<think>Okay, let's tackle this problem. So, the task is to determine the most important features for predicting whether a ... <think>
output: tumor-size,inv-nodes,deg-malig,node caps, irradiat, menopause, age, breast-quad
Llama3.2-3B:
['tumor-size', 'inv-nodes', 'node-caps', 'deg-malig', 'breast-quad', 'breast', 'irradiat']
GPT-4.1-mini:
['tumor-size', 'breast-quad', 'deg-malig', inv-nodes', 'breast', 'node-caps', 'irradiat']
Gemini-2.5-Pro:
['tumor-size', 'inv-nodes', ' 'irradiat', 'deg-malig', 'breast-quad', 'breast', 'node-caps ']

Figure 11: **Self-attribution prompt composition.** This setup is designed to produce a ranking of m features, indicating which attributes it believes were most influential for its decision. The bottom panel presents model outputs for the same attribution prompt on the example dataset (Breast Cancer).

System Prompt:
Below is an instruction that describes a task, paired with an input table that provides further context. Write a response that appropriately completes the request.

Instruction:
Dataset: **{Dataset name}**
Role: You are a **{Role player}**
Task: Your task is to perform **{binary/multi} classification** , predicting {task description}.
Target Encoding: {0: {class_1}, 1: {class_2}}
Attribute Glossary: **attribute_1**: attribute information,..., *** attribute _m*: attribute information.
Class Priors: **0**: 0.51, **1**: 0.5.

Your Job:
- For every row where 'class=?', predict its integer-encoded target relying solely on your pre-trained knowledge.
- Return one integer per row, in the exact same order as the rows appear.
- The number of predictions must equal the number of rows with 'class=?'.

Important:
- Do NOT include any code, code blocks, or explanations of code in your answer.
- Do NOT use or mention sklearn, pandas, or any code-based methods.
- DO NOT output text as target labels, only output the integer-encoded target.

Input:
Row 1: age=2, menopause=2, ~~tumor-size=0~~, inv-nodes=0, breast=1, breast-quad=2, class=0
⋮
Row 5: age=4, menopause=0, ~~tumor-size=4~~, inv-nodes=0, breast=1, breast-quad=2, class=1
Row 7: age=4, menopause=0, ~~tumor-size=4~~, inv-nodes=0, breast=1, breast-quad=2, class=?
⋮
Row 9: age=4, menopause=0, ~~tumor-size=4~~, inv-nodes=0, breast=1, breast-quad=2, class=?

Question:
Predict the integer-encoded class for each row where 'class=?'. There are exactly {num prediction} rows with 'class=?' in the input table. Do not output more or fewer predictions than {num prediction}.

Response:
DeepSeek-R1-Llama-8B:
<think>Alright, so I'm trying to figure out how to predict whether a <think>
output:[1, 4, 5, 6, 7, 8, 9, 10]
Gemma-1B:
output:[0, 0, 0, 0, 0, 0, 0, 0,..., 0]
Gemma-4B:
output:[1, 1, 1, 1, 1, 1, 1, 1,..., 1]
Qwen-8B:
<think>Alright, so I'm trying to figure out how to predict whether a <think>
output:
Row 1: age=2, ... inv-nodes=0, breast=1, breast-quad=2, class=0
Row 2: age=8, ...,inv-nodes=0, breast=1, breast-quad=2, class=0
⋮
Row 8: age=8, ...,inv-nodes=0, breast=1, breast-quad=2, class=0
Llama3.2-8B:
output:[0, 0, 0, 0, 0, 0, 0, 0,..., 0]
GPT-4.1-mini:
output:[1, 1, 0, 1, 0, 0, 1, 0]
Gemini-2.5-Pro:
output:[0, 1, 0, 1, 0, 0, 0, 1]

Figure 12: **LAO prompt composition.** This setup is designed to re-evaluate the model under identical prompting while ablating that feature from every row. The bottom panel presents model outputs for the same attribution prompt on the example dataset (Breast Cancer).

System Prompt:
Below is the output from a language model that has made predictions for a tabular dataset.
The output is expected to be a list of features, but it may not be formatted correctly or may contain additional text. Your task is to extract the feature rankings from this output. The output may contain extra explanations or formatting, which you should ignore. You must return a complete Python list of feature names (e.g., ['sex', 'age', ...])

User: Here is the text: {output text from LLMs}.

System Prompt:
Below is the output from a language model that has made predictions for a tabular dataset.
The output is expected to be a list of predictions, but it may not be formatted correctly or may contain additional text. Your task is to extract the integer predictions from this output. The output may contain extra explanations or formatting, which you should ignore. You must return a complete Python list of integers (e.g., [0, 1, 0, ...]), where each integer corresponds to a prediction for a row in the dataset.

User: Here is the text: {output text from LLMs}.

Figure 13: **Post-processing assistant prompts** for extracting structured outputs from LLM generations. *Top panel*: Prompt for extracting a list of integer predictions from noisy or verbose LLM responses. *Bottom panel*: Prompt for extracting ranked feature names from attribution outputs, isolating the comma-separated list from surrounding text.

Table 9: Runtime for single-feature LAO evaluation. Reported times correspond to one ablation pass per feature, aggregated across all features within the dataset. Mean and median values are computed across feature-deletion runs and provide an estimate of typical per-feature wall-clock cost.

Dataset	Model	# Runs	Mean (s)	Median (s)
Breast Cancer	GPT-4.1-mini	10	9.97	9.87
Iris	GPT-4.1-mini	5	3.78	4.27
Breast Cancer	Qwen3-8B	10	257.23	263.57
Iris	Qwen3-8B	5	102.45	75.22
Breast Cancer	MedGemma-4B	10	461.28	458.60
Pima Diabetes	MedGemma-4B	9	884.15	883.32

prompt setting. Replacement, marginal-sampling, permutation, group-ablation, and distribution-based analyses add additional intervention runs.

Table 9 reports wall-clock runtime for single-feature LAO evaluation. Runtime is dominated by inference latency. GPT-4.1-mini is the fastest among the tested models, requiring 3.78s per feature on Iris and 9.97s on *Breast Cancer*. Qwen3-8B is slower, requiring 102.45s per feature on Iris and 257.23s on *Breast Cancer*. The domain-specialized MedGemma-4B is slower still, requiring 461.28s per feature on *Breast Cancer* and 884.15s on *Pima Diabetes*.

Table 10 reports runtime for correlated group ablations. Because each feature group is removed jointly in a single evaluation pass, group ablations are broadly comparable to one single-feature run rather than additive in the number of removed features. Overall, the cost profile is practical for API models and moderate-scale local models, but substantially higher for domain-specialized local models such as MedGemma-4B. Figure 14 summarizes the per-feature runtime differences.

Table 10: Runtime for correlated group ablations. Each group ablation requires a single evaluation pass for the jointly removed feature set. Values are illustrative because each group configuration was executed once.

Dataset	Model	Group size	Total / run (s)
breast	GPT-4.1-mini	3 cols	6.62
breast	GPT-4.1-mini	2 cols	6.12
breast	Qwen3-8B	3 cols	184.98
breast	Qwen3-8B	2 cols	242.33
iris	GPT-4.1-mini	3 cols	4.34
iris	GPT-4.1-mini	2 cols	12.43
iris	Qwen3-8B	3 cols	136.94
iris	Qwen3-8B	2 cols	121.78

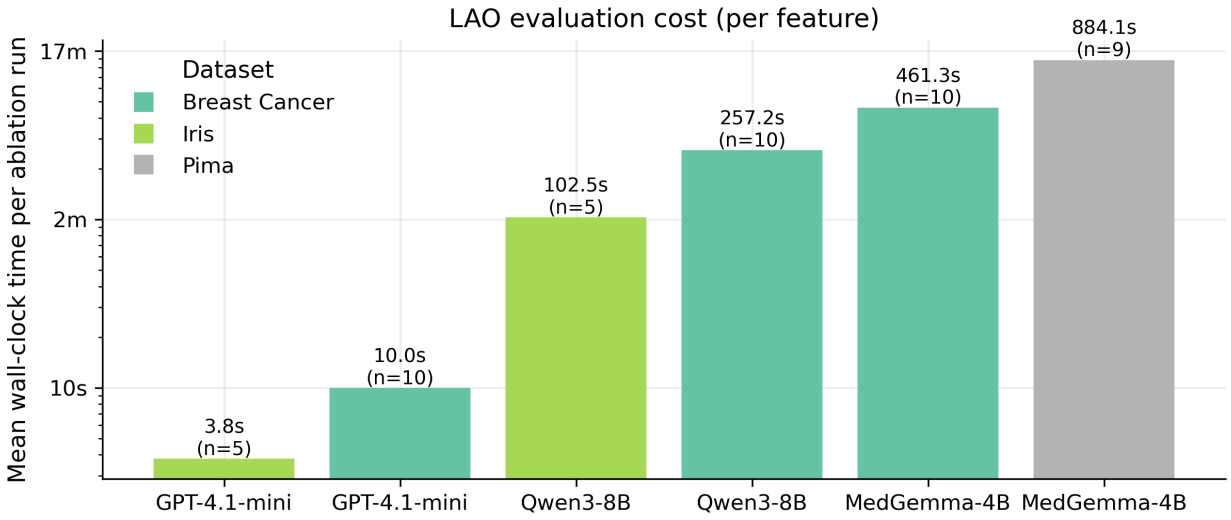


Figure 14: Per-feature wall-clock cost of LAO evaluation across datasets and models. GPT-4.1-mini is the most efficient among the tested models, Qwen3-8B is substantially slower, and MedGemma-4B incurs the largest cost. The y-axis is shown on a logarithmic scale to make differences across model families visible.

A.2 Prompt Construction & Serialization

A.2.1 Instruction Template.

Each prompt begins with a concise, self-contained instruction specifying *who* the model should act as and *what* task it is to perform. Specifically, we include the following fields: `⟨DATASET⟩`, `⟨ROLE⟩`, `⟨TASK TYPE⟩`, `⟨TARGET ENCODING⟩`, and an `⟨ATTRIBUTE GLOSSARY⟩`. Below are descriptions:

`⟨DATASET⟩` The name of the benchmark dataset. *Example:* `Breast Cancer`.

`⟨ROLE⟩` The professional identity that the model is instructed to assume, chosen to reflect domain expertise. *Example:* `clinical oncologist`.

`⟨TASK TYPE⟩` The prediction setting (e.g., binary or multi-class classification) together with its domain-specific description. *Example:* `binary classification - predicting whether a breast cancer patient will experience recurrence or not`.

`⟨TARGET ENCODING⟩` The mapping between integer-coded labels and their semantic meanings. This ensures the model outputs strictly integer predictions while preserving human interpretability. *Example:* `{0: no-recurrence-events, 1: recurrence-events}`.

`⟨ATTRIBUTE GLOSSARY⟩` A glossary listing each input feature, its semantic description, and categorical encodings (if applicable). This grounds the tabular features in explicit domain knowledge. *Example (Breast Cancer dataset, partial):*

- `age`: Age group of the patient {0: 10–19, 1: 20–29, . . . , 8: 90–99}
- `menopause`: Menopausal status {0: lt40, 1: ge40, 2: premeno}
- `tumor-size`: Tumor size intervals in mm {0: 0–4, 1: 5–9, . . . , 11: 55–59}
- `node-caps`: Capsular invasion {0: no, 1: yes}
- `irradiat`: Radiation therapy received {0: no, 1: yes}

The instruction typically opens with:

```
Act as a professional ⟨ROLE⟩,
Your task is to perform ⟨TASK TYPE⟩, predicting whether ...  or not.
⟨TARGET ENCODING⟩ ⟨ATTRIBUTE GLOSSARY⟩
For every row where "class=?", predict its integer target, relying solely on your
pre-trained knowledge.
Return one integer per row, in the exact same order as the rows appear.
The number of predictions must equal the number of rows with "class=?".
```

A.2.2 Tabular Input.

The dataset is rendered as plain text, with one row per line:

```
Row Num:  attribute_1 = 2, attribute_2 = 0, ..., class = ?
```

Categorical variables are integer-encoded, and the target-label encoding is provided explicitly in the instruction. Rows requiring prediction are marked with `class=?`. For deletion-based LAO experiments, the specified feature or feature group is removed from every row while all other attributes remain unchanged. For robustness analyses, we additionally compare deletion with constant replacement, mean replacement, empirical marginal sampling, and column-wise permutation.

A.2.3 Question.

The task is restated in a single sentence, explicitly specifying *the exact number N* of unknown rows for clarity:

> *Predict the integer-encoded class for the N rows where* `class=?`*. Output* exactly *N predictions in the same order.*

The length requirement links directly to the Len-F1 metric.

A.2.4 Output Format.

The model is instructed to return a list of integer labels. For instance,

```
[0, 2, 1, ..., 3]
```

i.e., a Python-style list of N comma-separated integers and *no additional text*. Any deviation triggers the label penalties in Sec. 4.

Models are instructed to return a Python-style list of integer labels with no additional text. Because some models nevertheless produce explanations, malformed lists, or extra tokens, we apply a standardized extraction step to convert raw outputs into clean prediction lists. This step uses GPT-4.1-mini only as a formatting normalizer: it is instructed to extract the integer predictions from the raw response without inferring missing labels or correcting the model's substantive answers.

B Extended Results

Comprehension Fidelity & Predictive Competence Table 18 – 32 list detailed Acc, Macro-F1, PenAcc, Len-F1, UnkLbl%, and Set-Jacc for zero-shot and few-shot settings across all 15 datasets and 11 models. Table 33 summarizes penalized accuracy of best performing models.

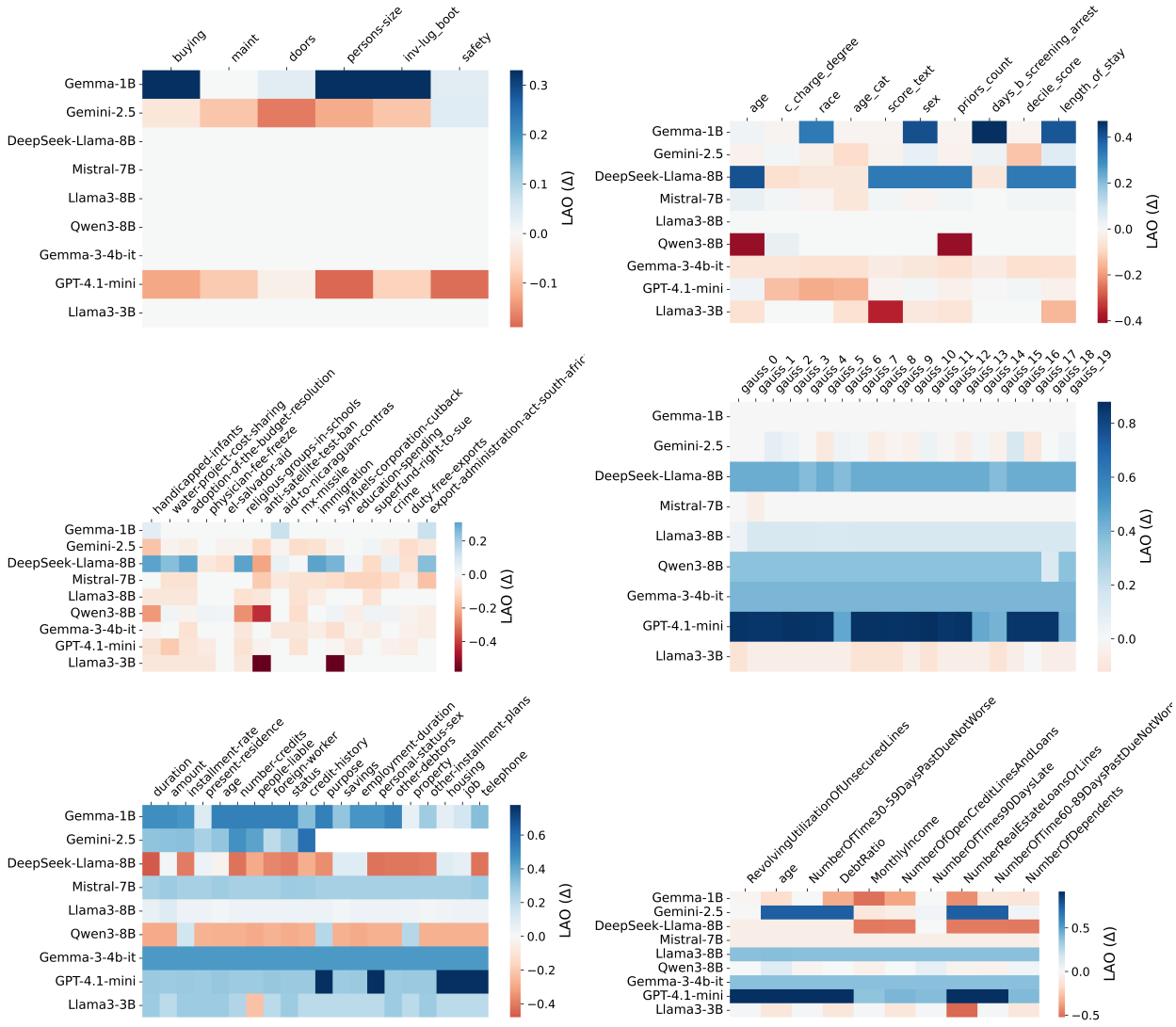


Figure 15: Heatmap of LAO performance (Δ_{LAO}) for each feature (columns) and LLM (rows). Darker blue indicates a larger performance loss when the feature is removed (higher importance); red indicates a slight performance gain or negligible reliance.

Decision Faithfulness Table 11 - 17 list Δ_{LAO}, Self-Faith, and SelfAtt@k for every model–dataset pair. Fig. 15, 16, and 17 visualize heatmaps of LAO performance (Δ_{LAO}) across all datasets.

Average of feature–target statistical dependency metrics across all benchmark datasets is provided in Table 34. "Triangulated" faithfulness across all models and datasets are provided in Tabel 35.

Table 35: "Triangulated" Faithfulness Across All Models and Datasets.

Dataset	Model	$\rho(\pi_{\text{self}}, \pi_{\text{LAO}})$	$\rho(\pi_{\text{self}}, \pi_{\text{NMI}})$	$\rho(\pi_{\text{LAO}}, \pi_{\text{NMI}})$
Adult Income				
	Gemma-4B	0.552† [0.098]	0.394 [0.260]	0.547* [0.043]
	Gemma-1B	— [—]	— [—]	-0.165 [0.573]
	Gemini-2.5-Pro	0.253 [0.383]	0.477† [0.085]	0.187 [0.523]
	DeepSeek-Llama-8B	0.240 [0.409]	0.301 [0.296]	-0.187 [0.523]
	Mistral-7B	-0.545† [0.083]	-0.509 [0.110]	0.235 [0.418]

Continued on next page

Dataset	Model	$\rho(\pi_{\text{self}}, \pi_{\text{LAO}})$	$\rho(\pi_{\text{self}}, \pi_{\text{NMI}})$	$\rho(\pi_{\text{LAO}}, \pi_{\text{NMI}})$
	Llama3-8B	-0.045 [0.894]	0.436 [0.180]	-0.182 [0.533]
	Qwen3-8B	-0.167 [0.668]	0.350 [0.356]	0.481† [0.081]
	GPT-4.1-mini	-0.015 [0.958]	0.240 [0.409]	-0.618* [0.019]
	Llama3-3B	-0.336 [0.240]	0.121 [0.681]	-0.415 [0.140]
Breast Cancer				
	DeepSeek-Llama-8B	0.810* [0.015]	0.586 [0.127]	0.009 [0.982]
	Llama3-3B	-0.929* [0.003]	0.259 [0.574]	0.183 [0.638]
	GPT-4.1-mini	0.217 [0.576]	0.775* [0.014]	0.000 [1.000]
	Mistral-7B	0.150 [0.700]	0.366 [0.333]	0.583† [0.099]
	Gemini-2.5-Pro	-0.607 [0.148]	0.595 [0.159]	-0.409 [0.274]
	Llama3-8B	-0.183 [0.637]	0.522 [0.149]	-0.148 [0.704]
	Gemma-1B	-1.000* [0.000]	— [—]	0.574 [0.106]
	Qwen3-8B	0.333 [0.420]	0.634† [0.091]	-0.392 [0.297]
	Gemma-4B	-0.048 [0.911]	0.366 [0.373]	0.078 [0.841]
Car Evaluation				
	Llama3-3B	-1.000 [—]	1.000 [—]	0.257 [0.623]
	Gemini-2.5-Pro	0.657 [0.156]	0.886* [0.019]	0.314 [0.544]
	GPT-4.1-mini	-0.600 [0.208]	0.657 [0.156]	-0.943* [0.005]
	DeepSeek-Llama-8B	0.029 [0.957]	0.143 [0.787]	-0.143 [0.787]
	Llama3-8B	0.314 [0.544]	0.657 [0.156]	0.429 [0.397]
	Mistral-7B	-0.086 [0.872]	0.086 [0.872]	-0.543 [0.266]
	Gemma-4B	-0.943* [0.005]	-0.086 [0.872]	-0.143 [0.787]
	Gemma-1B	-0.600 [0.208]	-0.314 [0.544]	-0.086 [0.872]
	Qwen3-8B	0.543 [0.266]	0.257 [0.623]	0.429 [0.397]
COMPAS				
	Gemini-2.5-Pro	-0.576 [0.082]	0.030 [0.934]	-0.212 [0.556]
	Gemma-4B	0.500 [0.207]	-0.167 [0.693]	-0.200 [0.580]
	Gemma-1B	-0.452 [0.260]	-0.119 [0.779]	-0.515 [0.128]
	Llama3-8B	0.033 [0.932]	-0.417 [0.265]	0.406 [0.244]
	Mistral-7B	0.881* [0.004]	0.095 [0.823]	0.285 [0.425]
	Qwen3-8B	-0.455 [0.187]	0.103 [0.777]	-0.248 [0.489]
	GPT-4.1-mini	0.212 [0.556]	0.394 [0.260]	0.248 [0.489]
	Llama3-3B	-0.086 [0.872]	-0.543 [0.266]	0.103 [0.777]
	DeepSeek-Llama-8B	0.030 [0.934]	0.636* [0.048]	0.188 [0.603]
Congressional Voting				
	DeepSeek-Llama-8B	0.248 [0.392]	-0.095 [0.748]	-0.453 [0.078]
	Llama3-8B	-0.243 [0.383]	-0.013 [0.965]	0.477† [0.062]
	Mistral-7B	0.445 [0.128]	-0.058 [0.851]	0.319 [0.228]
	GPT-4.1-mini	0.226 [0.399]	0.350 [0.184]	-0.037 [0.892]
	Gemma-4B	0.068 [0.810]	0.579* [0.024]	0.306 [0.249]
	Gemma-1B	0.643 [0.119]	-0.487 [0.268]	-0.311 [0.242]
	Qwen3-8B	0.014 [0.960]	0.465 [0.081]	0.255 [0.341]
	Llama3-3B	-0.379 [0.147]	0.041 [0.880]	0.284 [0.286]
	Gemini-2.5-Pro	0.435 [0.092]	0.748* [0.001]	0.383 [0.144]
Gaussian Synthetic				
	Gemma-1B	— [—]	— [—]	-0.061 [0.798]
	Gemma-4B	-0.060 [0.801]	0.645* [0.002]	-0.301 [0.197]
	DeepSeek-Llama-8B	0.800 [0.200]	0.800 [0.200]	0.271 [0.248]
	Llama3-3B	-0.232 [0.326]	0.645* [0.002]	-0.057 [0.810]
	GPT-4.1-mini	-0.477* [0.034]	0.192 [0.418]	-0.101 [0.671]
	Mistral-7B	-0.356 [0.123]	0.575* [0.008]	-0.525* [0.017]
	Qwen3-8B	0.171 [0.470]	-0.193 [0.414]	-0.031 [0.897]
	Gemini-2.5-Pro	-0.059 [0.806]	0.380 [0.098]	0.225 [0.340]
	Llama3-8B	-0.018 [0.940]	0.645* [0.002]	-0.306 [0.190]
German Credit				
	Llama3-3B	0.344 [0.137]	0.021 [0.929]	0.268 [0.253]
	DeepSeek-Llama-8B	0.260 [0.283]	-0.108 [0.659]	-0.430 [0.059]
	Llama3-8B	0.586* [0.008]	-0.033 [0.893]	0.092 [0.699]
	Mistral-7B	-0.164 [0.631]	0.569 [0.068]	0.021 [0.929]
	Qwen3-8B	-0.051 [0.836]	-0.066 [0.787]	-0.066 [0.783]
	GPT-4.1-mini	-0.179 [0.450]	-0.053 [0.825]	-0.095 [0.689]
	Gemma-1B	0.250 [0.369]	0.113 [0.689]	-0.148 [0.533]
	Gemma-4B	0.162 [0.521]	0.125 [0.622]	0.097 [0.684]
	Gemini-2.5-Pro	0.139 [0.701]	-0.020 [0.955]	0.389 [0.266]
Give Me Some Credit				
	Mistral-7B	-0.429 [0.289]	0.452 [0.260]	-0.394 [0.260]
	GPT-4.1-mini	0.236 [0.511]	0.709* [0.022]	0.261 [0.467]
	Gemma-4B	-0.083 [0.831]	0.400 [0.286]	-0.139 [0.701]
	Gemini-2.5-Pro	-0.152 [0.676]	0.758* [0.011]	0.127 [0.726]
	Llama3-8B	0.224 [0.533]	0.224 [0.533]	-0.564 [0.090]
	Gemma-1B	— [—]	— [—]	0.685* [0.029]
	DeepSeek-Llama-8B	0.595 [0.120]	0.333 [0.420]	0.673* [0.033]

Continued on next page

Dataset	Model	$\rho(\pi_{\text{self}}, \pi_{\text{LAO}})$	$\rho(\pi_{\text{self}}, \pi_{\text{NMI}})$	$\rho(\pi_{\text{LAO}}, \pi_{\text{NMI}})$
	Qwen3-8B	0.214 [0.610]	0.857* [0.007]	-0.030 [0.934]
	Llama3-3B	0.107 [0.819]	0.893* [0.007]	0.406 [0.244]
Heart Disease				
	Mistral-7B	-0.200 [0.475]	0.170 [0.545]	-0.435 [0.105]
	Gemma-1B	0.039 [0.889]	-0.129 [0.647]	0.275 [0.322]
	Gemma-4B	-0.154 [0.633]	0.303 [0.339]	0.144 [0.609]
	Gemini-2.5-Pro	0.657* [0.008]	0.572* [0.026]	0.061 [0.829]
	Llama3-8B	0.032 [0.909]	0.380 [0.162]	-0.046 [0.870]
	GPT-4.1-mini	0.496 [0.060]	0.321 [0.244]	-0.100 [0.724]
	Llama3-3B	0.393 [0.147]	-0.129 [0.647]	-0.184 [0.511]
	DeepSeek-Llama-8B	0.115 [0.707]	0.328 [0.274]	-0.266 [0.339]
	Qwen3-8B	0.054 [0.850]	0.245 [0.378]	0.172 [0.541]
HELOC				
	Gemma-4B	0.139 [0.536]	0.194 [0.388]	0.024 [0.914]
	Gemma-1B	1.000 [—]	1.000 [—]	0.196 [0.371]
	GPT-4.1-mini	-0.229 [0.293]	0.173 [0.430]	0.060 [0.785]
	DeepSeek-Llama-8B	-0.005 [0.984]	0.265 [0.287]	0.123 [0.578]
	Gemini-2.5-Pro	0.251 [0.248]	0.696* [0.000]	0.275 [0.205]
	Llama3-3B	-0.218 [0.317]	0.133 [0.544]	-0.009 [0.968]
	Llama3-8B	0.156 [0.564]	-0.097 [0.721]	0.199 [0.364]
	Qwen3-8B	-0.170 [0.438]	0.042 [0.847]	0.410 [0.052]
	Mistral-7B	0.462* [0.030]	0.263 [0.238]	0.190 [0.386]
Iris				
	Llama3-8B	0.200 [0.800]	0.400 [0.600]	0.800 [0.200]
	DeepSeek-Llama-8B	-0.800 [0.200]	-0.600 [0.400]	0.000 [1.000]
	Mistral-7B	0.800 [0.200]	-0.600 [0.400]	-0.800 [0.200]
	Gemini-2.5-Pro	0.800 [0.200]	1.000* [0.000]	0.800 [0.200]
	Gemma-1B	0.600 [0.400]	-0.600 [0.400]	-1.000* [0.000]
	Gemma-4B	0.400 [0.600]	-0.600 [0.400]	0.400 [0.600]
	Qwen3-8B	0.400 [0.600]	1.000* [0.000]	0.400 [0.600]
	GPT-4.1-mini	0.800 [0.200]	1.000* [0.000]	0.800 [0.200]
	Llama3-3B	-0.600 [0.400]	-0.600 [0.400]	1.000* [0.000]
Monk 1				
	DeepSeek-Llama-8B	0.200 [0.704]	0.676 [0.140]	0.845* [0.034]
	Gemini-2.5-Pro	0.943* [0.005]	0.372 [0.468]	0.541 [0.268]
	Llama3-3B	-0.943* [0.005]	-0.101 [0.848]	0.338 [0.512]
	Mistral-7B	1.000* [0.000]	0.500 [0.667]	-0.068 [0.899]
	Llama3-8B	-0.657 [0.156]	0.372 [0.468]	0.034 [0.949]
	GPT-4.1-mini	0.314 [0.544]	0.372 [0.468]	-0.135 [0.798]
	Gemma-4B	-0.900* [0.037]	-0.112 [0.858]	-0.372 [0.468]
	Gemma-1B	-0.200 [0.704]	-0.101 [0.848]	-0.778 [0.069]
	Qwen3-8B	0.500 [0.667]	-1.000* [0.000]	-0.169 [0.749]
Monk 2				
	Gemini-2.5-Pro	-0.371 [0.468]	0.030 [0.954]	0.152 [0.774]
	Llama3-8B	0.886* [0.019]	0.030 [0.954]	-0.030 [0.954]
	Gemma-1B	-0.771 [0.072]	0.030 [0.954]	-0.638 [0.173]
	Gemma-4B	0.771 [0.072]	0.030 [0.954]	-0.152 [0.774]
	DeepSeek-Llama-8B	0.000 [1.000]	-0.316 [0.684]	-0.395 [0.439]
	Mistral-7B	-0.657 [0.156]	0.030 [0.954]	0.516 [0.295]
	Qwen3-8B	0.314 [0.544]	-0.091 [0.864]	-0.030 [0.954]
	GPT-4.1-mini	0.771 [0.072]	0.030 [0.954]	-0.213 [0.686]
	Llama3-3B	0.086 [0.872]	0.030 [0.954]	0.030 [0.954]
Monk 3				
	Mistral-7B	0.500 [0.667]	0.500 [0.667]	-0.395 [0.439]
	Gemma-1B	-0.200 [0.704]	-0.395 [0.439]	0.334 [0.518]
	Gemma-4B	0.500 [0.391]	-0.447 [0.450]	0.516 [0.295]
	Llama3-8B	0.143 [0.787]	0.395 [0.439]	-0.152 [0.774]
	GPT-4.1-mini	-0.600 [0.208]	0.395 [0.439]	-0.395 [0.439]
	Qwen3-8B	1.000* [0.000]	0.500 [0.667]	0.516 [0.295]
	Gemini-2.5-Pro	-0.200 [0.704]	0.577 [0.231]	0.334 [0.518]
	DeepSeek-Llama-8B	0.714 [0.111]	0.395 [0.439]	0.395 [0.439]
	Llama3-3B	-0.143 [0.787]	-0.395 [0.439]	-0.516 [0.295]
Pima Diabetes				
	Gemini-2.5-Pro	0.429 [0.289]	0.976* [0.000]	0.571 [0.139]
	Gemma-1B	-0.429 [0.289]	0.429 [0.289]	0.238 [0.570]
	Gemma-4B	0.333 [0.420]	0.095 [0.823]	-0.571 [0.139]
	Qwen3-8B	-0.429 [0.289]	0.857* [0.007]	-0.286 [0.493]
	GPT-4.1-mini	-0.095 [0.823]	0.905* [0.002]	-0.333 [0.420]
	DeepSeek-Llama-8B	0.952* [0.000]	-0.286 [0.493]	-0.286 [0.493]
	Llama3-3B	-0.393 [0.383]	-0.036 [0.939]	-0.286 [0.493]
	Llama3-8B	-0.381 [0.352]	-0.262 [0.531]	-0.286 [0.493]
	Mistral-7B	-0.400 [0.505]	-0.300 [0.624]	0.048 [0.911]

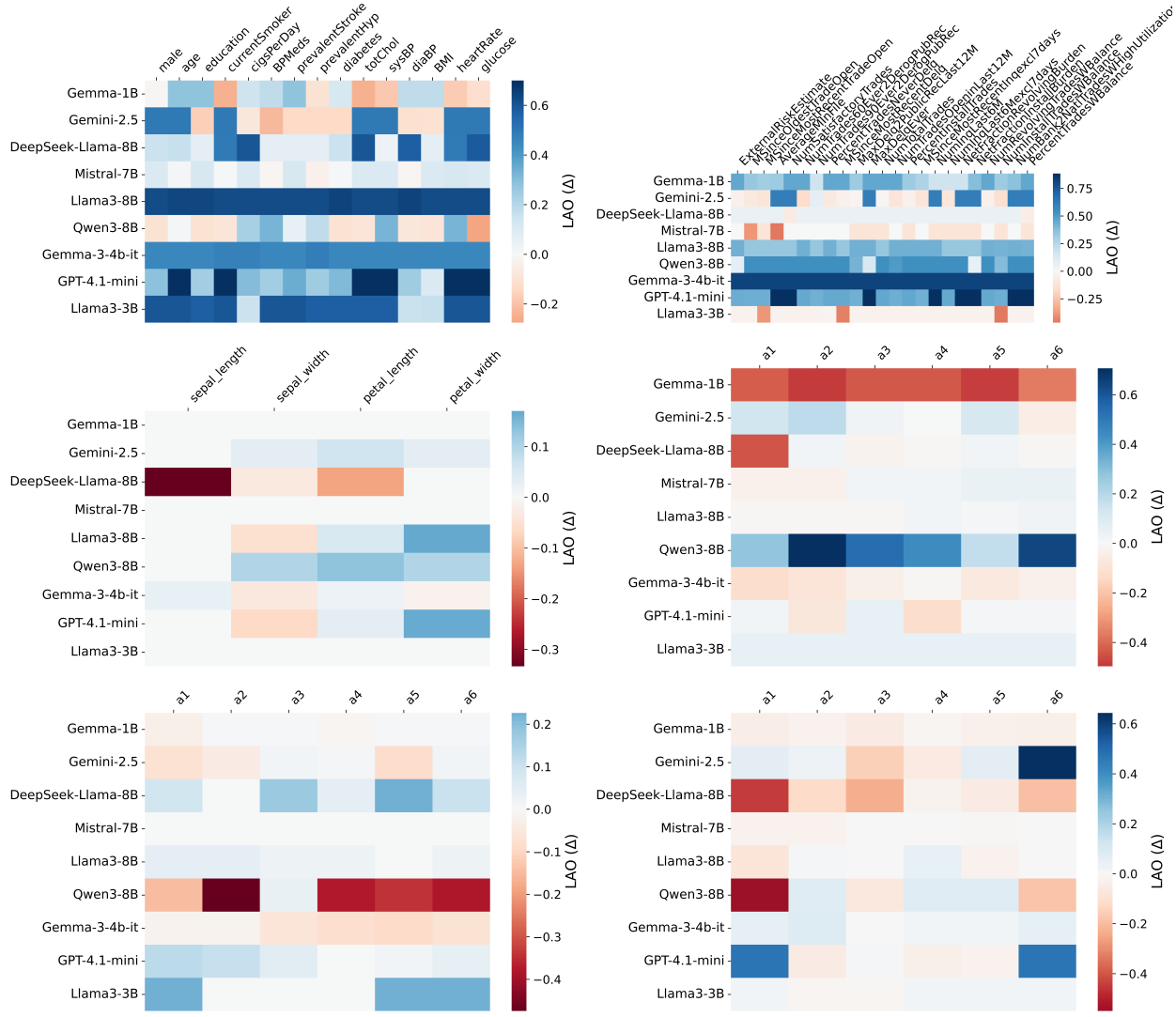


Figure 16: Heatmap of LAO performance (Δ_{LAO}) for each feature (columns) and LLM (rows). Darker blue indicates a larger performance loss when the feature is removed (higher importance); red indicates a slight performance gain or negligible reliance.

Notes. Dashes (—) denote undefined due to NaNs or degenerate ranks. Stars: * $p<.05$, † $p<.10$. Brackets show p-values.

B.1 Perturbation-operator sensitivity

Deletion-based LAO is the primary perturbation used in STaDS because it directly instantiates a missing-information counterfactual: a named decision factor is removed from the prompt, and we measure how the model's predictive behavior changes when that field is unavailable. However, deletion is not the only possible perturbation for tabular inputs. To evaluate how sensitive the inferred feature-reliance profile is to perturbation semantics, we compare deletion-LAO with four alternative operators on two representative datasets, *Breast Cancer* and Iris, using Qwen3-8B and GPT-4.1-mini.

The perturbation operators preserve and disrupt different aspects of the structured prompt. Constant replacement keeps the feature slot visible but replaces all values with a fixed placeholder. Mean replacement preserves the feature scale while suppressing instance-specific variation. Empirical marginal sampling resamples values from the feature's marginal distribution, preserving univariate plausibility while breaking the original row-level assignment. Column-wise permutation preserves the exact empirical set of observed values but reassigns them to different rows, thereby

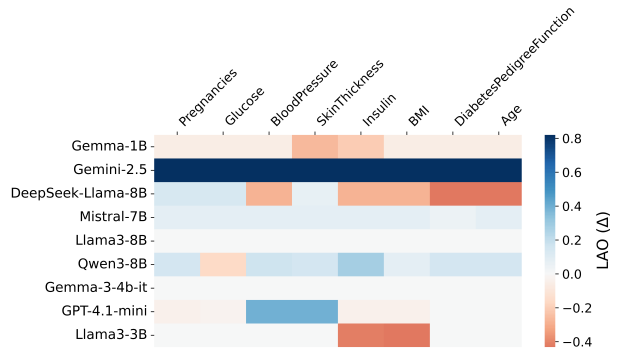


Figure 17: Heatmap of LAO performance (Δ_{LAO}) for each feature (columns) of `Car Evaluation` and LLM (rows). Darker blue indicates a larger performance loss when the feature is removed (higher importance); red indicates a slight performance gain or negligible reliance.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.17	NaN	0.00
Gemini-2.5-Pro	0.07	0.25 (0.38)	1.00
DeepSeek-Llama-8B	0.24	0.24 (0.41)	1.00
Llama3-8B	0.00	−0.05 (0.89)	0.73
Qwen3-8B	0.11	−0.17 (0.67)	0.44
GPT-4.1-mini	0.01	−0.02 (0.96)	1.00
Llama3-3B	0.11	−0.34 (0.24)	1.00
Mistral-7B	0.01	−0.54 (0.08)	0.73

Table 11: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Adult Income Dataset. NaN indicates π_{self} empty.

breaking row-wise alignment with the remaining features. These distinctions are important for prompted LLMs because perturbations alter not only feature information, but also the serialized input distribution, visible schema, and local decision context.

For each operator, we perturb one feature at a time, re-run the same prompting and evaluation pipeline, and induce a feature ranking from the resulting performance-change profile. We then compare each operator-specific ranking with the deletion-LAO ranking using Spearman correlation. We also report the positive-effect rate, defined as the fraction of features for which perturbation improves predictive performance:

$$\Delta = \text{Acc}_{\text{perturbed}} - \text{Acc}_{\text{base}} > 0.$$

Positive effects are informative because they show that perturbations do not always behave like simple removal of useful information; in LLM prompts, perturbations can also remove misleading fields or introduce prompt artifacts that improve performance.

Table 36 shows that feature-reliance rankings are strongly operator-dependent. Agreement with deletion-LAO varies widely across datasets and models, including negative correlations. For example, Iris–GPT-4.1-mini has high agreement under constant replacement ($\rho = 0.95$), but negative agreement under permutation ($\rho = -0.40$) and marginal sampling ($\rho = -0.80$). Breast–Qwen3-8B similarly shifts from positive agreement under constant replacement ($\rho = 0.53$) to negative agreement under marginal sampling ($\rho = -0.69$). Thus, different perturbation mechanisms can induce materially different importance orderings.

Table 37 further shows that perturbations often improve performance. Positive effects occur for 66.7%–88.9% of Breast–Qwen3-8B features under several replacement or permutation operators, for 75.0% of Iris–Qwen3-8B features under multiple operators, and for all Iris–GPT-4.1-mini features under mean replacement. Figure 18 visualizes these effects at the feature level: both the sign and magnitude of feature-level performance changes depend substantially on the perturbation operator, and the induced feature rankings can move markedly across operators.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.24	−1.00 (0.00)	0.33
Gemini-2.5-Pro	0.04	−0.61 (0.15)	0.78
DeepSeek-Llama-8B	0.23	0.81 (0.01)	0.89
Llama3-8B	0.02	−0.18 (0.64)	1.00
Qwen3-8B	0.16	0.33 (0.42)	0.89
GPT-4.1-mini	0.04	0.22 (0.58)	1.00
Llama3-3B	0.11	−0.93 (0.00)	0.78
Mistral-7B	0.09	0.15 (0.70)	1.00

Table 12: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Breast Cancer Dataset. NaN indicates π_{self} empty.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.17	−0.60 (0.21)	1.00
Gemini-2.5-Pro	0.07	0.66 (0.16)	1.00
DeepSeek-Llama-8B	0.00	0.03 (0.96)	1.00
Llama3-8B	0.00	0.31 (0.54)	1.00
Qwen3-8B	0.00	0.54 (0.27)	1.00
GPT-4.1-mini	0.07	−0.60 (0.21)	1.00
Llama3-3B	0.00	−1.00 (NaN)	0.33

Table 13: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Car Evaluation Dataset. NaN indicates π_{self} empty.

These results motivate our use of deletion-LAO as the primary STaDS intervention. The claim is not that deletion is universally superior to all alternatives. Rather, deletion is most directly aligned with the STaDS estimand: sensitivity to the absence of a named decision factor. Replacement, marginal-sampling, and permutation variants are therefore treated as complementary diagnostics for operator sensitivity, including sensitivity to substituted values, resampled marginal distributions, or corrupted row-wise feature alignment.

B.2 Sensitivity of Penalized Accuracy to penalty weights

Penalized Accuracy combines prediction correctness with penalties for output-length mismatch and invalid labels. Because this metric contains penalty weights, we evaluate whether the main trends depend on the default choice $\alpha = \beta = 0.5$. Recall that:

$$\text{PenAcc} = \text{Acc} - \big(\alpha(1 - \text{Len-F1}) + \beta\,\text{UnkLbl\%}\big), \tag{4}$$

where $\alpha, \beta \geq 0$ control the relative penalty assigned to length-fidelity and invalid-label violations.

We vary (α, β) along the simplex $\alpha + \beta = 1$ and recompute both the base PenAcc and the LAO-induced degradation under each setting. We report the sensitivity analysis on Iris and *Breast Cancer* as representative case studies.

Figure 20 shows that the mean degradation induced by feature removal changes smoothly as the weighting shifts between length-fidelity and unknown-label penalties. We do not observe abrupt reversals or qualitatively unstable behavior across the explored range. These results indicate that the main LAO-based conclusions are not an artifact of a single hand-picked penalty configuration. The precise PenAcc values naturally change with the intended emphasis on different format failures, but the degradation trends remain stable. We therefore use equal weighting as a neutral default in the main experiments rather than as a tuned hyperparameter.

B.3 Audit of output post-processing

Some LLM outputs contain extra text, malformed lists, or other formatting deviations that complicate metric extraction. STaDS therefore uses an output-cleaning step to extract prediction lists. To test whether this step changes substantive conclusions, we audit post-processing under three settings: no ablation, single-column ablation, and multi-column ablation.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemini-2.5-Pro	0.06	−0.58 (0.08)	1.00
Gemma-1B	0.21	−0.45 (0.26)	0.80
DeepSeek-Llama-8B	0.21	0.03 (0.94)	1.00
Llama3-8B	0.00	0.03 (0.93)	0.90
Qwen3-8B	0.18	−0.45 (0.19)	1.00
GPT-4.1-mini	0.08	0.21 (0.56)	1.00
Llama3-3B	0.11	−0.09 (0.87)	0.60

Table 14: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); COMPAS Dataset. NaN indicates π_{self} empty.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.04	0.64 (0.12)	0.44
Gemini-2.5-Pro	0.05	0.44 (0.09)	1.00
DeepSeek-Llama-8B	0.19	0.25 (0.39)	0.88
Llama3-8B	0.04	−0.24 (0.38)	0.94
Qwen3-8B	0.13	0.01 (0.96)	0.94
GPT-4.1-mini	0.04	0.23 (0.40)	1.00
Llama3-3B	0.19	−0.38 (0.15)	1.00

Table 15: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Congression Vote Record Dataset. NaN indicates π_{self} empty.

Table 38 shows that post-processing preserves accuracy in nearly all evaluated settings. All no-ablation runs remain at 1.00 post-audit accuracy. For single-column ablations, all settings remain at 1.00 except *Breast Cancer*–Qwen3-8B, which drops slightly to 0.94. All evaluated multi-column settings remain at 1.00. These results suggest that post-processing primarily normalizes formatting rather than correcting model predictions. We also examined the small accuracy drop for *Breast Cancer*–Qwen3-8B under single-column ablation and found that the main issue is inconsistency between generated reasoning text and the final predicted label, rather than systematic correction by the post-processing model.

B.4 Correlated group ablations and higher-order reliance

Single-feature LAO provides a first-order view of behavioral reliance, but correlated predictors can produce higher-order effects that are not captured by removing features individually. To examine such cases, we conduct group ablations on correlated feature subsets for Iris and *Breast Cancer*. For Iris, we consider the pair `petal_length`–`petal_width` and the three-feature set `petal_length`–`petal_width`–`sepal_length`. For *Breast Cancer*, we consider `inv-nodes`–`deg-malig`–`irradiat`, selected from the strongest statistical associations in Table 34.

We define the standardized performance drop after removing a feature set S as:

$$\Delta(S) = \text{Acc}_{\text{base}} - \text{Acc}_{\setminus S}, \tag{5}$$

where larger positive values indicate stronger reliance on the removed feature set. To isolate non-additive group effects, we compute:

$$I(S) = \Delta(S) - \sum_{j \in S} \Delta(\{j\}), \tag{6}$$

where $I(S) > 0$ indicates super-additive interaction, $I(S) \approx 0$ indicates near-additivity, and $I(S) < 0$ indicates redundancy or overlap.

Table 39 and Figures 21–22 show that higher-order reliance is present in some settings but is strongly model-dependent. On Iris, Qwen3-8B exhibits clear super-additive behavior: removing `petal_length` and `petal_width` individually yields moderate drops (0.133 and 0.100), but removing the pair yields a much larger drop (0.433), giving a positive pair interaction of 0.200. Removing the three-feature group yields an even larger drop (0.567), with a triple interaction of 0.333. This suggests joint reliance on correlated petal features that single-feature ablations do not fully capture.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.00	0.60 (0.40)	1.00
Gemini-2.5-Pro	0.03	0.80 (0.20)	1.00
DeepSeek-Llama-8B	0.15	−0.80 (0.20)	1.00
Llama3-8B	0.10	0.20 (0.80)	1.00
Qwen3-8B	0.06	0.40 (0.60)	1.00
GPT-4.1-mini	0.10	0.80 (0.20)	1.00
Llama3-3B	0.00	−0.60 (0.40)	1.00

Table 16: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Iris Dataset. NaN indicates π_{self} empty.

Model	σ_{LAO}	Self-Faith	SelfAtt@ k
Gemma-1B	0.05	−0.20 (0.70)	1.00
Gemini-2.5-Pro	0.09	0.94 (0.00)	1.00
DeepSeek-Llama-8B	0.18	0.20 (0.70)	1.00
Llama3-8B	0.02	−0.66 (0.16)	1.00
Qwen3-8B	0.21	0.50 (0.67)	0.50
GPT-4.1-mini	0.07	0.31 (0.54)	1.00
Llama3-3B	0.00	−0.94 (0.00)	1.00

Table 17: Decision faithfulness metrics between self-attribution rank (π_{self}) and LAO-attribution rank (π_{LAO}); Monk 1 Dataset. NaN indicates π_{self} empty.

By contrast, GPT-4.1-mini on Iris exhibits negative interactions for the same feature groups. Although individual removals produce drops of 0.033 and 0.167, pair removal yields only 0.037, corresponding to a negative interaction of −0.163. The three-feature group likewise yields a near-zero drop (0.003), suggesting redundancy rather than synergy. On *Breast Cancer*, the pattern is again model-dependent. GPT-4.1-mini exhibits positive non-additive effects for both pair and triple removals, whereas Qwen3-8B shows weaker and more heterogeneous group effects.

These findings show that single-feature LAO can miss higher-order reliance, but also that interaction effects are not universal. Group ablations are therefore best understood as a complementary stress test for joint dependence rather than as a replacement for single-feature LAO. Extending this analysis to larger feature sets is possible, but the computational cost grows quickly with interaction order.

B.5 Domain-specialized medical model: MedGemma

Domain specialization may improve medical plausibility, but it does not necessarily guarantee global decision faithfulness. To examine this distinction, we evaluate `google/medgemma-4b` on two healthcare datasets, *Breast Cancer* and *Pima Diabetes*, and compare it with the general-domain Gemma3-4B baseline. For each model, we measure behavioral reliance using deletion-LAO and compare the induced feature ranking with the model's self-reported feature-importance ranking.

Figure 8 shows that domain specialization does not automatically eliminate the gap between stated and behavioral reliance. On *Pima Diabetes*, Gemma3-4B exhibits an almost flat LAO profile despite producing a non-flat self-reported ranking. This indicates that the model can articulate a preference ordering without showing corresponding behavioral sensitivity under intervention. MedGemma produces more domain-relevant self-attributions and stronger behavioral sensitivity than Gemma3-4B, but its stated ranking still does not align strongly with the LAO ranking. On *Breast Cancer*, both models again show mismatches between self-attribution and behavioral reliance. MedGemma identifies medically interpretable variables in the dataset context, such as `inv-nodes`, `tumor-size`, and `deg-malig`, but the resulting self–LAO agreement remains weak.

Table 40 summarizes this pattern. The main conclusion is that domain specialization appears to improve the plausibility of self-reported rationales more readily than global decision faithfulness. Expert-domain models may produce more domain-appropriate explanations while still relying behaviorally on a different set of features under intervention.

Adult Income — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.010	0.020	0.000	0.198	81.8	0.182
Gemma-4B	0.490	0.428	0.194	0.408	0.0	1.000
Llama3-8B	0.530	0.530	0.236	0.412	0.0	1.000
Mistral-7B	0.410	0.377	0.402	0.985	0.0	1.000
GPT-4.1-mini	0.470	0.320	0.470	1.000	0.0	1.000
Llama3-3B	0.010	0.020	0.000	0.020	0.0	0.500
Gemini-2.5-Pro	0.700	0.688	0.700	1.000	0.0	1.000
Qwen3-8B	0.380	0.433	0.247	0.734	0.0	1.000
DeepSeek-Llama-8B	0.500	0.333	0.198	0.397	0.0	0.500
Adult Income — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-4B	0.053	0.083	0.000	0.100	0.0	0.500
Gemma-1B	0.474	0.321	0.461	0.974	0.0	1.000
Qwen3-8B	0.579	0.367	0.579	1.000	0.0	0.500
Llama3-3B	0.579	0.367	0.104	0.049	0.0	0.500
Gemini-2.5-Pro	0.737	0.708	0.737	1.000	0.0	1.000
Llama3-8B	0.579	0.367	0.407	0.655	0.0	0.500
GPT-4.1-mini	0.632	0.614	0.632	1.000	0.0	1.000
DeepSeek-Llama-8B	0.421	0.296	0.421	1.000	0.0	0.500
Mistral-7B	0.053	0.083	0.000	0.100	0.0	0.500

Table 18: Adult Income: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Concrete failure-case illustrations. The MedGemma case study provides two concrete examples of the accuracy–faithfulness gap in a clinically motivated setting. In the *Pima Diabetes* task, MedGemma reports a clinically plausible ranking in which `Glucose`, `Insulin`, `Pregnancies`, `Age`, and `BMI` are treated as salient decision factors. However, the deletion-LAO profile does not show corresponding behavioral dependence on several of these stated factors; instead, the strongest measured effect is associated with `SkinThickness`, while several self-attributed variables have weak or even negative LAO effects. This is a failure mode in which the model's explanation is medically plausible but behaviorally unsupported under intervention. In the *Breast Cancer* task, MedGemma similarly identifies medically interpretable variables such as `inv-nodes`, `tumor-size`, and `deg-malig`, yet the measured LAO ranking only partially overlaps with this self-ranking, with stronger behavioral effects also appearing for features such as `age` and `breast`. These examples illustrate why domain specialization alone is insufficient for global decision faithfulness: an expert-domain model may produce rationales that appear clinically appropriate while its predictions remain sensitive to a different set of input fields. Thus, the failure is not merely low predictive accuracy or malformed output, but a mismatch between stated medical decision factors and the factors that actually affect the model's decisions under controlled missing-information interventions.

Breast — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
DeepSeek-Llama-8B	0.697	0.433	0.365	0.337	0.0	1.000
Gemini-2.5-Pro	0.650	0.575	0.646	0.993	0.0	1.000
Gemma-1B	0.708	0.414	0.376	0.337	0.0	0.500
Gemma-4B	0.000	0.000	0.000	0.007	1.0	0.000
Mistral-7B	0.690	0.408	0.358	0.337	0.0	1.000
GPT-4.1-mini	0.729	0.524	0.348	0.238	0.0	1.000
Llama3-8B	0.704	0.413	0.373	0.337	0.0	1.000
Qwen3-8B	0.224	0.296	0.000	0.478	0.0	1.000
Llama3-3B	0.708	0.414	0.376	0.337	0.0	0.500
Breast — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Mistral-7B	0.036	0.082	0.000	0.133	0.0	1.000
GPT-4.1-mini	0.732	0.525	0.732	1.000	0.0	1.000
Qwen3-8B	0.679	0.491	0.218	0.079	0.0	1.000
Gemini-2.5-Pro	0.589	0.548	0.589	1.000	0.0	1.000
Llama3-8B	0.732	0.525	0.272	0.079	0.0	1.000
DeepSeek-Llama-8B	0.554	0.495	0.093	0.079	0.0	1.000
Gemma-4B	0.714	0.470	0.254	0.079	0.0	1.000
Gemma-1B	0.714	0.417	0.714	1.000	0.0	0.500
Llama3-3B	0.107	0.120	0.000	0.303	0.0	0.500

Table 19: Breast: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Car evaluation — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemini-2.5-Pro	0.419	0.243	0.406	0.973	0.0	0.500
Gemma-4B	0.326	0.123	0.045	0.439	0.0	0.250
Gemma-1B	0.005	0.008	0.000	0.036	0.0	1.000
GPT-4.1-mini	0.326	0.129	0.000	0.316	0.0	0.750
DeepSeek-Llama-8B	0.263	0.219	0.000	0.439	0.0	1.000
Mistral-7B	0.279	0.215	0.000	0.439	0.0	0.750
Llama3-3B	0.000	0.000	0.000	0.000	0.0	0.000
Qwen3-8B	0.021	0.030	0.000	0.410	0.0	0.500
Llama3-8B	0.326	0.123	0.045	0.439	0.0	0.250
Car evaluation — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Mistral-7B	0.333	0.125	0.000	0.104	0.0	0.250
Gemma-1B	0.333	0.126	0.330	0.993	0.0	0.250
Gemma-4B	0.293	0.113	0.000	0.104	0.0	0.500
DeepSeek-Llama-8B	0.333	0.125	0.000	0.104	0.0	0.250
Llama3-8B	0.013	0.019	0.000	0.026	0.0	0.250
Gemini-2.5-Pro	0.600	0.616	0.586	0.973	0.0	1.000
Qwen3-8B	0.307	0.117	0.000	0.104	0.0	0.250
Llama3-3B	0.000	0.000	0.000	0.000	0.0	0.000
GPT-4.1-mini	0.173	0.079	0.160	0.974	0.0	0.750

Table 20: Car evaluation: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

COMPAS — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.502	0.338	0.270	0.536	0.0	1.000
Gemma-4B	0.500	0.495	0.268	0.536	0.0	1.000
Qwen3-8B	0.490	0.329	0.258	0.536	0.0	1.000
Gemini-2.5-Pro	0.816	0.810	0.571	0.510	0.0	1.000
GPT-4.1-mini	0.468	0.319	0.164	0.392	0.0	1.000
Llama3-8B	0.500	0.333	0.268	0.536	0.0	0.500
Mistral-7B	0.006	0.012	0.000	0.016	0.0	1.000
Llama3-3B	0.500	0.333	0.268	0.536	0.0	0.500
DeepSeek-Llama-8B	0.502	0.338	0.270	0.536	0.0	1.000
COMPAS — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-8B	0.511	0.338	0.072	0.121	0.0	0.500
Mistral-7B	0.500	0.333	0.061	0.121	0.0	1.000
Gemma-4B	0.011	0.022	0.000	0.022	0.0	0.500
GPT-4.1-mini	0.500	0.355	0.476	0.952	0.0	1.000
Gemma-1B	0.500	0.333	0.471	0.941	0.0	1.000
Qwen3-8B	0.511	0.338	0.072	0.121	0.0	0.500
Llama3-3B	0.011	0.022	0.000	0.108	0.0	1.000
Gemini-2.5-Pro	0.716	0.714	0.697	0.962	0.0	1.000
DeepSeek-Llama-8B	0.409	0.321	0.406	0.994	0.0	1.000

Table 21: COMPAS: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Congression Vote — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Mistral-7B	0.526	0.524	0.171	0.291	0.0	1.000
Gemini-2.5-Pro	0.397	0.395	0.388	0.982	0.0	1.000
Llama3-3B	0.534	0.348	0.180	0.291	0.0	0.500
Llama3-8B	0.478	0.478	0.124	0.291	0.0	1.000
Qwen3-8B	0.474	0.360	0.119	0.291	0.0	1.000
GPT-4.1-mini	0.409	0.401	0.011	0.204	0.0	1.000
DeepSeek-Llama-8B	0.466	0.325	0.111	0.291	0.0	1.000
Gemma-1B	0.263	0.274	0.062	0.598	0.0	0.500
Gemma-4B	0.526	0.525	0.171	0.291	0.0	1.000
Congression Vote — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-3B	0.064	0.086	0.000	0.351	0.0	0.500
GPT-4.1-mini	0.532	0.451	0.521	0.978	0.0	1.000
Gemini-2.5-Pro	0.638	0.636	0.638	1.000	0.0	1.000
Qwen3-8B	0.489	0.478	0.023	0.067	0.0	1.000
DeepSeek-Llama-8B	0.489	0.390	0.311	0.644	0.0	1.000
Mistral-7B	0.468	0.467	0.001	0.067	0.0	1.000
Llama3-8B	0.404	0.402	0.000	0.067	0.0	1.000
Gemma-4B	0.447	0.447	0.000	0.067	0.0	1.000
Gemma-1B	0.532	0.347	0.532	1.000	0.0	0.500

Table 22: Congression Vote: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Gaussian Synthetic — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.500	0.500	0.068	0.137	0.0	1.000
Gemma-4B	0.500	0.487	0.068	0.137	0.0	1.000
GPT-4.1-mini	0.390	0.281	0.390	1.000	0.0	1.000
Qwen3-8B	0.510	0.355	0.078	0.137	0.0	1.000
Llama3-8B	0.500	0.500	0.068	0.137	0.0	1.000
Gemini-2.5-Pro	0.550	0.448	0.550	1.000	0.0	1.000
DeepSeek-Llama-8B	0.350	0.398	0.288	0.876	0.0	1.000
Mistral-7B	0.500	0.500	0.068	0.137	0.0	1.000
Llama3-3B	0.000	0.000	0.000	0.000	0.0	0.000
Gaussian Synthetic — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-8B	0.440	0.306	0.273	0.667	0.0	1.000
Gemma-4B	0.440	0.306	0.403	0.926	0.0	0.500
Gemma-1B	0.400	0.384	0.000	0.036	0.0	1.000
Mistral-7B	0.000	0.000	0.000	0.958	95.7	0.042
GPT-4.1-mini	0.880	0.873	0.880	1.000	0.0	1.000
Gemini-2.5-Pro	0.680	0.603	0.680	1.000	0.0	1.000
Qwen3-8B	0.360	0.359	0.360	1.000	0.0	1.000
Llama3-3B	0.000	0.000	0.000	0.000	0.0	0.000
DeepSeek-Llama-8B	0.440	0.306	0.440	1.000	0.0	0.500

Table 23: Gaussian Synthetic: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

German Credit Risk — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
GPT-4.1-mini	0.660	0.616	0.660	1.000	0.00	1.000
Mistral-7B	0.550	0.436	0.118	0.137	0.00	1.000
Llama3-8B	0.500	0.500	0.068	0.137	0.00	1.000
DeepSeek-Llama-8B	0.490	0.329	0.058	0.137	0.00	1.000
Gemma-1B	0.490	0.490	0.058	0.137	0.00	1.000
Gemini-2.5-Pro	0.460	0.457	0.460	1.000	0.00	1.000
Qwen3-8B	0.020	0.038	0.000	0.387	0.00	1.000
Llama3-3B	0.000	0.000	0.000	0.000	0.00	0.000
Gemma-4B	0.000	0.000	0.000	0.020	1.00	0.000
German Credit Risk — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Mistral-7B	0.333	0.325	0.333	1.000	0.0	1.000
DeepSeek-Llama-8B	0.444	0.308	0.094	0.300	0.0	1.000
Llama3-8B	0.222	0.222	0.096	0.947	20.0	0.333
Llama3-3B	0.778	0.438	0.284	0.013	0.0	0.500
GPT-4.1-mini	0.778	0.679	0.778	1.000	0.0	1.000
Gemini-2.5-Pro	0.889	0.862	0.889	1.000	0.0	1.000
Gemma-4B	0.444	0.444	0.444	1.000	0.0	1.000
Qwen3-8B	0.222	0.182	0.222	1.000	0.0	0.500
Gemma-1B	0.556	0.357	0.529	0.947	0.0	1.000

Table 24: German Credit Risk: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Give Me Some Credit — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
GPT-4.1-mini	0.550	0.533	0.492	0.885	0.0	1.000
Llama3-8B	0.470	0.320	0.038	0.137	0.0	1.000
DeepSeek-Llama-8B	0.830	0.832	0.825	0.990	0.0	1.000
Mistral-7B	0.480	0.324	0.048	0.137	0.0	1.000
Llama3-3B	0.500	0.333	0.068	0.137	0.0	0.500
Gemini-2.5-Pro	0.790	0.785	0.790	1.000	0.0	1.000
Qwen3-8B	0.570	0.646	0.511	0.883	0.0	1.000
Gemma-4B	0.440	0.436	0.303	0.726	0.0	1.000
Gemma-1B	0.480	0.327	0.477	0.995	0.0	1.000
Give Me Some Credit — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Qwen3-8B	0.500	0.500	0.500	1.000	0.0	1.000
Llama3-3B	0.500	0.333	0.017	0.035	0.0	0.500
DeepSeek-Llama-8B	0.542	0.420	0.059	0.035	0.0	1.000
Gemini-2.5-Pro	0.750	0.743	0.750	1.000	0.0	1.000
Llama3-8B	0.500	0.333	0.500	1.000	0.0	0.500
Mistral-7B	0.500	0.333	0.017	0.035	0.0	0.500
GPT-4.1-mini	0.917	0.916	0.917	1.000	0.0	1.000
Gemma-1B	0.417	0.294	0.000	0.035	0.0	1.000
Gemma-4B	0.500	0.438	0.387	0.774	0.0	1.000

Table 25: Give Me Some Credit: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Heart Disease — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Qwen3-8B	0.420	0.484	0.352	0.864	0.0	1.000
GPT-4.1-mini	0.640	0.614	0.640	1.000	0.0	1.000
Llama3-8B	0.450	0.449	0.018	0.137	0.0	1.000
Gemma-1B	0.510	0.510	0.078	0.137	0.0	1.000
Gemma-4B	0.480	0.448	0.048	0.137	0.0	1.000
Mistral-7B	0.500	0.487	0.068	0.137	0.0	1.000
DeepSeek-Llama-8B	0.510	0.372	0.507	0.995	0.0	1.000
Llama3-3B	0.500	0.333	0.068	0.137	0.0	0.500
Gemini-2.5-Pro	0.550	0.544	0.550	1.000	0.0	1.000
Heart Disease — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-4B	0.650	0.642	0.483	0.667	0.0	1.000
Gemma-1B	0.450	0.429	0.283	0.667	0.0	1.000
GPT-4.1-mini	0.700	0.697	0.700	1.000	0.0	1.000
DeepSeek-Llama-8B	0.600	0.375	0.600	1.000	0.0	0.500
Llama3-8B	0.650	0.642	0.650	1.000	0.0	1.000
Llama3-3B	0.600	0.375	0.600	1.000	0.0	0.500
Gemini-2.5-Pro	0.500	0.495	0.500	1.000	0.0	1.000
Qwen3-8B	0.350	0.307	0.350	1.000	0.0	1.000
Mistral-7B	0.600	0.375	0.114	0.029	0.0	0.500

Table 26: Heart Disease: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

HELOC — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-3B	0.000	0.000	0.000	0.000	0.0	0.000
Llama3-8B	0.500	0.333	0.068	0.137	0.0	0.500
GPT-4.1-mini	0.650	0.601	0.650	1.000	0.0	1.000
DeepSeek-Llama-8B	0.510	0.372	0.507	0.995	0.0	1.000
Mistral-7B	0.500	0.500	0.068	0.137	0.0	1.000
Qwen3-8B	0.640	0.596	0.637	0.995	0.0	1.000
Gemma-4B	0.500	0.500	0.068	0.137	0.0	1.000
Gemma-1B	0.490	0.331	0.487	0.995	0.0	1.000
Gemini-2.5-Pro	0.670	0.670	0.670	1.000	0.0	1.000
HELOC — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-3B	0.038	0.077	0.000	0.074	0.0	0.500
Gemma-1B	0.462	0.316	0.462	1.000	0.0	0.500
Gemma-4B	0.885	0.883	0.811	0.852	0.0	1.000
Qwen3-8B	0.538	0.513	0.538	1.000	0.0	1.000
Llama3-8B	0.462	0.324	0.452	0.980	0.0	0.500
Gemini-2.5-Pro	0.615	0.613	0.615	1.000	0.0	1.000
GPT-4.1-mini	0.885	0.880	0.885	1.000	0.0	1.000
Mistral-7B	0.038	0.077	0.000	0.143	0.0	1.000
DeepSeek-Llama-8B	0.538	0.350	0.057	0.037	0.0	0.500

Table 27: HELOC: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Iris — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-3B	0.007	0.013	0.000	0.039	0.0	1.000
GPT-4.1-mini	0.487	0.516	0.458	0.944	0.0	1.000
Llama3-8B	0.373	0.373	0.000	0.198	0.0	1.000
Qwen3-8B	0.440	0.402	0.039	0.198	0.0	1.000
Mistral-7B	0.320	0.266	0.000	0.198	0.0	1.000
Gemma-1B	0.007	0.013	0.000	0.997	98.0	0.020
Gemma-4B	0.360	0.360	0.000	0.198	0.0	1.000
Gemini-2.5-Pro	0.787	0.787	0.785	0.997	0.0	1.000
DeepSeek-Llama-8B	0.380	0.376	0.000	0.198	0.0	1.000
Iris — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-3B	0.333	0.167	0.000	0.043	0.0	0.333
Gemma-4B	0.333	0.334	0.152	0.638	0.0	1.000
Gemma-1B	0.033	0.061	0.000	0.984	90.3	0.097
Gemini-2.5-Pro	1.000	1.000	1.000	1.000	0.0	1.000
GPT-4.1-mini	0.667	0.662	0.667	1.000	0.0	1.000
Qwen3-8B	0.933	0.933	0.933	1.000	0.0	1.000
DeepSeek-Llama-8B	0.367	0.330	0.367	1.000	0.0	1.000
Llama3-8B	0.300	0.295	0.170	0.741	0.0	1.000
Mistral-7B	0.367	0.355	0.000	0.043	0.0	1.000

Table 28: Iris: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Monk 1 — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Qwen3-8B	0.535	0.427	0.275	0.481	0.0	1.000
Llama3-8B	0.502	0.502	0.243	0.481	0.0	1.000
Gemma-4B	0.505	0.505	0.245	0.481	0.0	1.000
Mistral-7B	0.521	0.507	0.261	0.481	0.0	1.000
Gemma-1B	0.498	0.334	0.088	0.481	30.0	0.333
Gemini-2.5-Pro	0.620	0.613	0.618	0.995	0.0	1.000
GPT-4.1-mini	0.118	0.193	0.000	0.348	0.0	1.000
DeepSeek-Llama-8B	0.507	0.352	0.247	0.481	0.0	1.000
Llama3-3B	0.500	0.333	0.240	0.481	0.0	0.500
Monk 1 — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.034	0.059	0.000	0.168	0.0	1.000
Gemma-4B	0.011	0.023	0.000	0.023	0.0	0.500
GPT-4.1-mini	0.506	0.495	0.482	0.952	0.0	1.000
Llama3-8B	0.506	0.449	0.066	0.120	0.0	1.000
Mistral-7B	0.540	0.512	0.100	0.120	0.0	1.000
Gemini-2.5-Pro	0.724	0.724	0.721	0.994	0.0	1.000
Llama3-3B	0.494	0.331	0.054	0.120	0.0	1.000
DeepSeek-Llama-8B	0.483	0.476	0.043	0.120	0.0	1.000
Qwen3-8B	0.759	0.758	0.726	0.935	0.0	1.000

Table 29: Monk 1: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Monk 2 — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.174	0.187	0.000	0.481	40.0	0.200
Gemma-4B	0.329	0.247	0.069	0.481	0.000	0.500
Llama3-8B	0.500	0.485	0.240	0.481	0.000	1.000
DeepSeek-Llama-8B	0.674	0.409	0.414	0.481	0.000	1.000
Qwen3-8B	0.664	0.476	0.405	0.481	0.000	1.000
Mistral-7B	0.556	0.498	0.296	0.481	0.000	1.000
Llama3-3B	0.021	0.030	0.000	0.045	0.000	0.500
GPT-4.1-mini	0.660	0.397	0.334	0.348	0.000	1.000
Gemini-2.5-Pro	0.484	0.475	0.461	0.955	0.000	1.000
Monk 2 — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemma-1B	0.667	0.400	0.634	0.935	0.00	0.500
Gemma-4B	0.563	0.422	0.123	0.120	0.00	1.000
Llama3-8B	0.701	0.525	0.261	0.120	0.00	1.000
DeepSeek-Llama-8B	0.667	0.552	0.227	0.120	0.00	1.000
Qwen3-8B	0.713	0.601	0.273	0.120	0.00	1.000
Mistral-7B	0.678	0.436	0.238	0.120	0.00	1.000
Llama3-3B	0.667	0.400	0.227	0.120	0.00	0.500
GPT-4.1-mini	0.575	0.450	0.569	0.988	0.00	1.000
Gemini-2.5-Pro	0.494	0.456	0.489	0.989	0.00	1.000

Table 30: Monk 2: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Monk 3 — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
DeepSeek-Llama-8B	0.512	0.511	0.252	0.481	0.00	1.000
Llama3-3B	0.472	0.321	0.213	0.481	0.00	0.500
Gemini-2.5-Pro	0.579	0.596	0.563	0.969	0.00	1.000
GPT-4.1-mini	0.507	0.423	0.181	0.348	0.00	1.000
Llama3-8B	0.495	0.495	0.236	0.481	0.00	1.000
Gemma-4B	0.528	0.345	0.268	0.481	0.00	0.500
Gemma-1B	0.528	0.345	0.268	0.481	0.00	0.500
Mistral-7B	0.500	0.474	0.240	0.481	0.00	1.000
Qwen3-8B	0.491	0.362	0.231	0.481	0.00	1.000
Monk 3 — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Llama3-8B	0.494	0.494	0.054	0.120	0.00	1.000
Gemini-2.5-Pro	0.644	0.642	0.644	1.000	0.00	1.000
Qwen3-8B	0.517	0.516	0.077	0.120	0.00	1.000
Llama3-3B	0.460	0.315	0.020	0.120	0.00	1.000
Mistral-7B	0.437	0.335	0.000	0.120	0.00	1.000
DeepSeek-Llama-8B	0.437	0.432	0.000	0.120	0.00	1.000
Gemma-1B	0.425	0.351	0.388	0.926	0.00	1.000
Gemma-4B	0.529	0.346	0.089	0.120	0.00	0.500
GPT-4.1-mini	0.494	0.492	0.470	0.952	0.00	1.000

Table 31: Monk 3: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Pima Diabetes — Zero-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemini-2.5-Pro	0.758	0.784	0.741	0.967	0.00	1.000
Qwen3-8B	0.538	0.531	0.306	0.536	0.00	1.000
Llama3-8B	0.496	0.332	0.264	0.536	0.00	1.000
Mistral-7B	0.492	0.330	0.260	0.536	0.00	1.000
DeepSeek-Llama-8B	0.500	0.333	0.268	0.536	0.00	0.500
GPT-4.1-mini	0.452	0.311	0.148	0.392	0.00	1.000
Gemma-1B	0.196	0.282	0.000	0.331	0.00	1.000
Gemma-4B	0.020	0.038	0.000	0.039	0.00	0.500
Llama3-3B	0.002	0.004	0.000	0.008	0.00	1.000
Pima Diabetes — Few-shot						
Model	Acc	Macro-F1	PenAcc	Len-F1	UnkLbl%	Set-Jacc
Gemini-2.5-Pro	0.820	0.814	0.820	1.000	0.00	1.000
Qwen3-8B	0.580	0.408	0.577	0.995	0.00	1.000
DeepSeek-Llama-8B	0.570	0.363	0.138	0.137	0.00	0.500
Mistral-7B	0.510	0.508	0.078	0.137	0.00	1.000
Gemma-4B	0.420	0.296	0.000	0.137	0.00	1.000
Llama3-8B	0.430	0.301	0.000	0.137	0.00	0.500
GPT-4.1-mini	0.390	0.281	0.390	1.000	0.00	1.000
Gemma-1B	0.370	0.272	0.367	0.995	0.00	1.000
Llama3-3B	0.000	0.000	0.000	0.000	0.00	0.000

Table 32: Pima Diabetes: STaDS metrics per model for zero-shot and few-shot. Acc = accuracy; PenAcc = penalised accuracy; Len-F1 = length F1; UnkLbl% = unknown label rate (%).

Dataset	Zero-shot (by PA) P-Acc	(Acc)	Best Z Model	Few-shot (by PA) P-Acc	(Acc)	Best F Model
Adult Income	0.700	0.700	Gemini-2.5-Pro	0.737	0.737	Gemini-2.5-Pro
Breast Cancer	0.646	0.650	Gemini-2.5-Pro	0.732	0.732	GPT-4.1-mini
Car Evaluation	0.406	0.419	Gemini-2.5-Pro	0.586	0.600	Gemini-2.5-Pro
COMPAS	0.571	0.816	Gemini-2.5-Pro	0.697	0.716	Gemini-2.5-Pro
Congression Vote	0.388	0.397	Gemini-2.5-Pro	0.638	0.638	Gemini-2.5-Pro
Gaussian Synthetic	0.550	0.550	Gemini-2.5-Pro	0.880	0.880	GPT-4.1-mini
German Credit	0.660	0.660	GPT-4.1-mini	0.889	0.889	Gemini-2.5-Pro
Give Me Some Credit	0.825	0.830	DeepSeek-Llama-8B	0.917	0.917	GPT-4.1-mini
Heart Disease	0.640	0.640	GPT-4.1-mini	0.700	0.700	GPT-4.1-mini
HELOC	0.670	0.670	Gemini-2.5-Pro	0.885	0.885	GPT-4.1-mini
Iris	0.785	0.787	Gemini-2.5-Pro	1.000	1.000	Gemini-2.5-Pro
Monk 1	0.618	0.620	Gemini-2.5-Pro	0.726	0.759	Qwen3-8B
Monk 2	0.461	0.484	Gemini-2.5-Pro	0.634	0.667	Gemma-1B
Monk 3	0.563	0.579	Gemini-2.5-Pro	0.644	0.644	Gemini-2.5-Pro
Pima Diabetes	0.741	0.758	Gemini-2.5-Pro	0.820	0.820	Gemini-2.5-Pro

Table 33: Penalised accuracy (P-Acc) summary. Higher is better; penalisation reduces scores for overlong outputs and invalid labels.(α=0.5, β=0.5).

Dataset	**Mean Cramér's V**	**Mean NMI**	**Mean Pearson r**	**Mean Spearman ρ**	**Top-3 by NMI**
Adult Income	0.308	0.053	0.143	0.164	`relationship`, `marital-status`, `capital-gain`
Breast Cancer	0.162	0.017	–	–	`inv-nodes`, `deg-malig`, `irradiat`
Car Evaluation	0.196	0.072	–	–	`persons`, `safety`, `buying`
COMPAS	0.203	0.027	0.071	0.086	`decile_score`, `score_text`, `priors_count`
Congressional Voting	0.503	0.165	–	–	`physician-fee-freeze`, `el-salvador-aid`, `education-spending`
Gaussian Synthetic	–	0.019	−0.019	−0.022	`gauss_1`, `gauss_2`, `gauss_6`
German Credit	0.025	0.006	−0.025	−0.023	`other-installment-plans`, `installment-rate`, `number-credits`
Give Me Some Credit	0.205	0.014	−0.031	0.017	`RevolvingUtilizationOfUnsecuredLines`, `NumberOfTimes90DaysLate`, `NumberOfTime30-59DaysPastDueNotWorse`
Heart Disease	0.083	0.008	0.122	0.104	`age`, `prevalentHyp`, `sysBP`
HELOC	0.189	0.030	0.035	0.036	`ExternalRiskEstimate`, `NetFractionRevolvingBurden`, `PercentTradesWBalance`
Iris	0.633	0.677	0.866	0.867	`petal_length`, `petal_width`, `sepal_length`
Monk 1	0.095	0.041	–	–	`a5`, `a2`, `a1`
Monk 2	0.021	0.012	–	–	`a4`, `a1`, `a6`
Monk 3	0.213	0.075	–	–	`a5`, `a2`, `a6`
Pima Diabetes	0.252	0.047	0.206	0.224	`Glucose`, `BMI`, `Age`

Table 34: Average of feature–target statistical dependency metrics across all benchmark datasets. Values are averaged over all features within each dataset. Dashes indicate non-applicable metrics (e.g., Pearson/Spearman for categorical targets). Top-ranked features by NMI highlight dominant statistical dependencies, which serve as proxies for co-occurrence rather than causal relationships.

Table 36: Rank agreement between alternative perturbation operators and deletion-based LAO. Each entry reports Spearman correlation ρ between the feature ranking induced by an alternative operator and the deletion-LAO ranking. High positive values indicate similar induced rankings, whereas low or negative values indicate that the inferred reliance ordering changes under the perturbation semantics. The broad variation across datasets and models shows that feature-reliance estimates are operator-dependent in prompted tabular LLM evaluation.

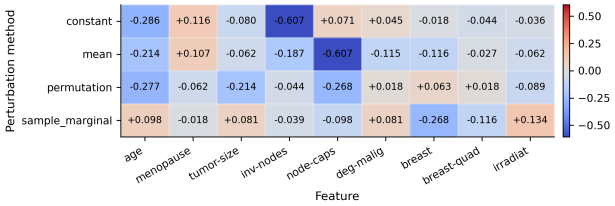
Dataset	Model	Constant	Mean	Permutation	Marginal
Breast	Qwen3-8B	+0.53	-0.29	+0.22	-0.69
Breast	GPT-4.1-mini	+0.05	+0.17	+0.41	+0.42
Iris	Qwen3-8B	-0.40	+0.20	+0.00	+0.60
Iris	GPT-4.1-mini	+0.95	+0.45	-0.40	-0.80

Table 37: Fraction of features with positive perturbation effect ($\Delta > 0$), where $\Delta = \mathrm{Acc}_{\mathrm{perturbed}} - \mathrm{Acc}_{\mathrm{base}}$. Positive values indicate cases where perturbing a feature improves predictive performance rather than degrading it. Frequent positive effects show that perturbations can alter the prompt in ways that do not correspond to simple performance degradation from removing useful information.

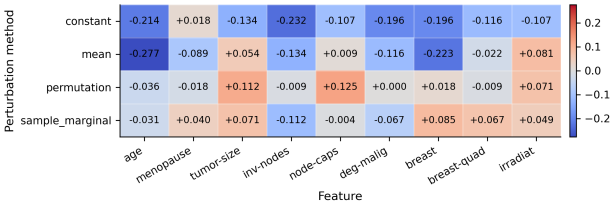
Dataset	Model	Drop (LAO)	Constant	Mean	Permutation	Marginal
Breast	Qwen3-8B	55.6%	66.7%	88.9%	66.7%	55.6%
Breast	GPT-4.1-mini	44.4%	11.1%	33.3%	44.4%	55.6%
Iris	Qwen3-8B	25.0%	75.0%	75.0%	50.0%	75.0%
Iris	GPT-4.1-mini	50.0%	0.0%	100.0%	25.0%	50.0%

Table 38: Post-auditing accuracy after output cleaning. We report average accuracy after applying the output-cleaning step under no-ablation, single-feature ablation, and multi-column ablation settings. Near-perfect preservation of accuracy indicates that post-processing acts primarily as formatting normalization rather than substantive prediction correction.

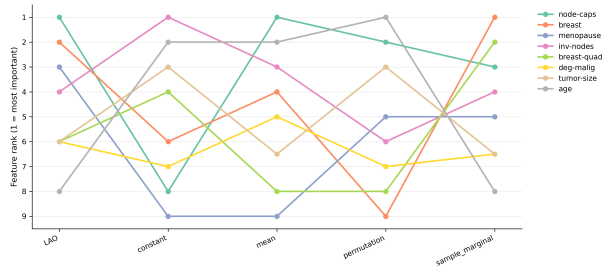
Dataset	Model	Ablation setting	Post-audit Acc (avg)
breast	GPT-4.1-mini	0 col	1.00
breast	Qwen3-8B	0 col	1.00
iris	GPT-4.1-mini	0 col	1.00
iris	Qwen3-8B	0 col	1.00
breast	GPT-4.1-mini	1 col	1.00
breast	Qwen3-8B	1 col	0.94
iris	GPT-4.1-mini	1 col	1.00
iris	Qwen3-8B	1 col	1.00
pima	MedGemma-4B	1 col	1.00
iris	GPT-4.1-mini	multi-cols	1.00
breast	Qwen3-8B	multi-cols	1.00
breast	GPT-4.1-mini	multi-cols	1.00
iris	Qwen3-8B	multi-cols	1.00



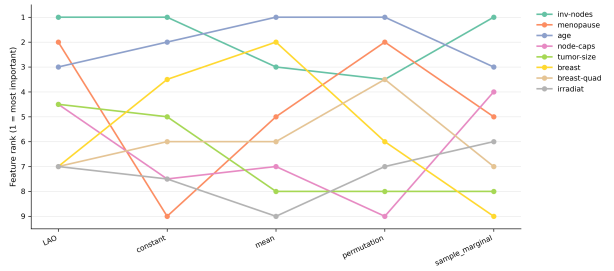
(a) *Breast Cancer*, Qwen3-8B: feature-level performance change under each perturbation operator.



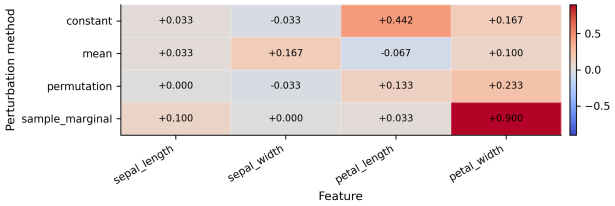
(b) *Breast Cancer*, GPT-4.1-mini: feature-level performance change under each perturbation operator.



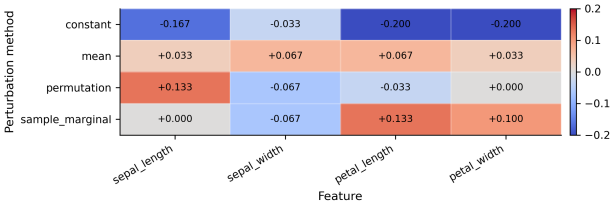
(c) *Breast Cancer*, Qwen3-8B: induced feature-ranking shifts across perturbation operators.



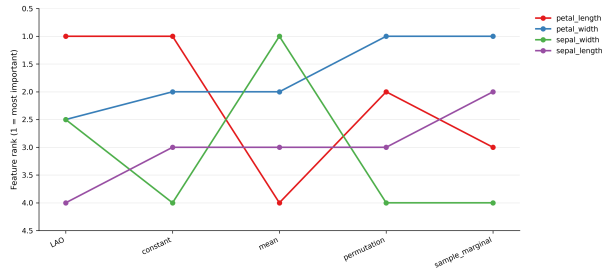
(d) *Breast Cancer*, GPT-4.1-mini: induced feature-ranking shifts across perturbation operators.



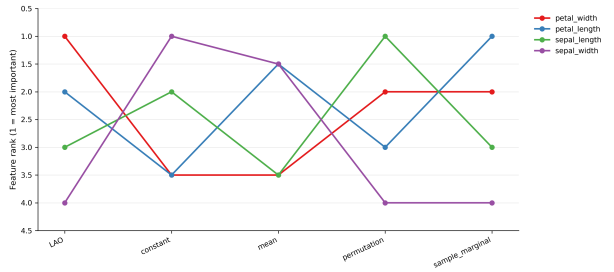
(e) Iris, Qwen3-8B: feature-level performance change under each perturbation operator.



(f) Iris, GPT-4.1-mini: feature-level performance change under each perturbation operator.

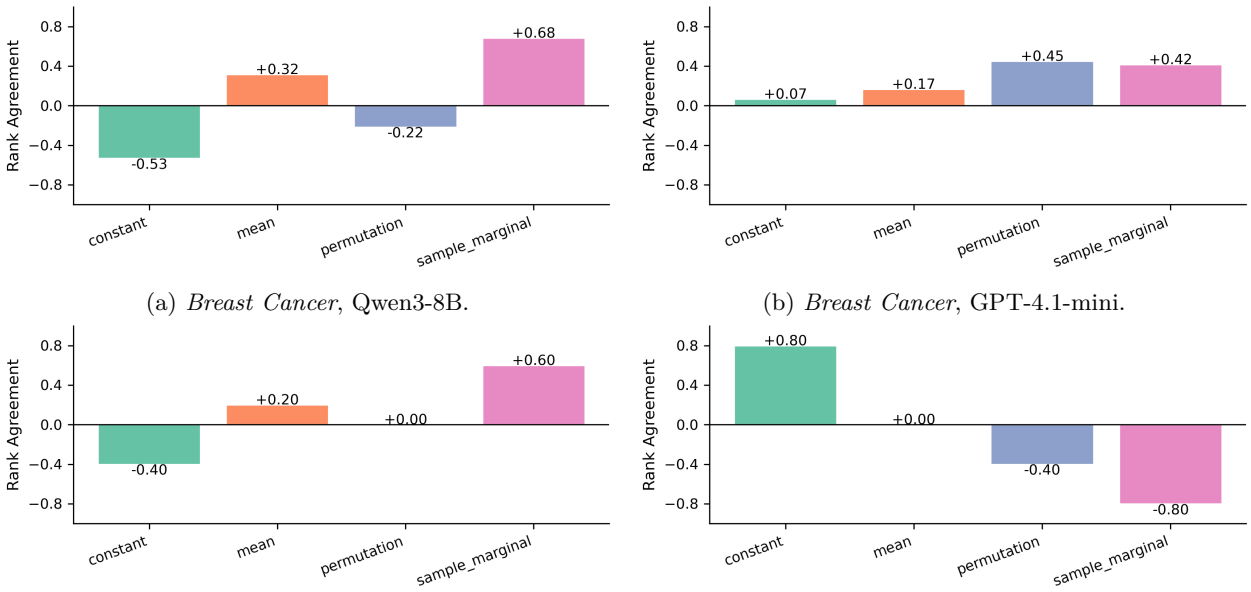


(g) Iris, Qwen3-8B: induced feature-ranking shifts across perturbation operators.



(h) Iris, GPT-4.1-mini: induced feature-ranking shifts across perturbation operators.

Figure 18: Perturbation-operator sensitivity of feature-reliance estimates. Heatmaps show feature-level performance change Δ under each perturbation operator, and rank-bump plots show how the induced feature-importance ordering changes across operators. The substantial movement in both sign and rank shows that deletion, replacement, marginal sampling, and permutation instantiate different counterfactual interventions in prompted tabular LLM evaluation.



(a) *Breast Cancer*, Qwen3-8B.

(b) *Breast Cancer*, GPT-4.1-mini.

(c) Iris, Qwen3-8B.

(d) Iris, GPT-4.1-mini.

Figure 19: Rank agreement between alternative perturbation operators and deletion-based LAO. Bars report Spearman correlation between each operator-induced feature ranking and the deletion-LAO ranking. The spread of values, including negative correlations in some settings, shows that estimated reliance rankings can change substantially when the perturbation operator changes. Deletion-LAO is therefore used as the primary missing-information intervention, while the remaining operators serve as diagnostics of operator sensitivity.

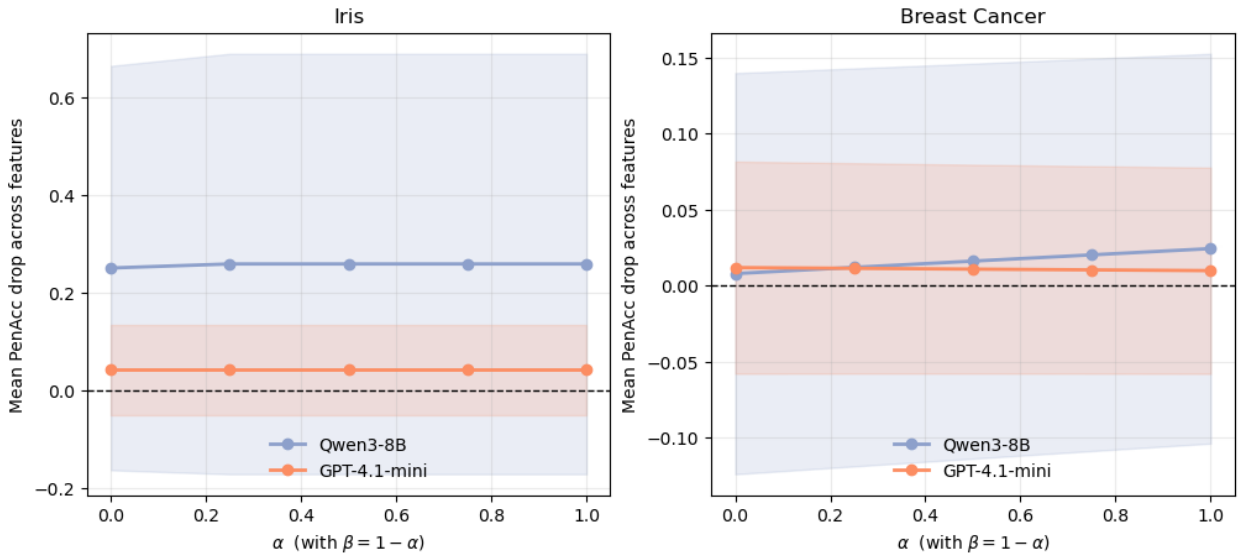


Figure 20: Sensitivity of mean LAO-induced PenAcc degradation to the penalty trade-off α with $\beta = 1 - \alpha$. Across Iris and *Breast Cancer*, the average degradation varies smoothly rather than erratically, indicating that the qualitative conclusions of the LAO analysis are stable across reasonable penalty-weight choices.

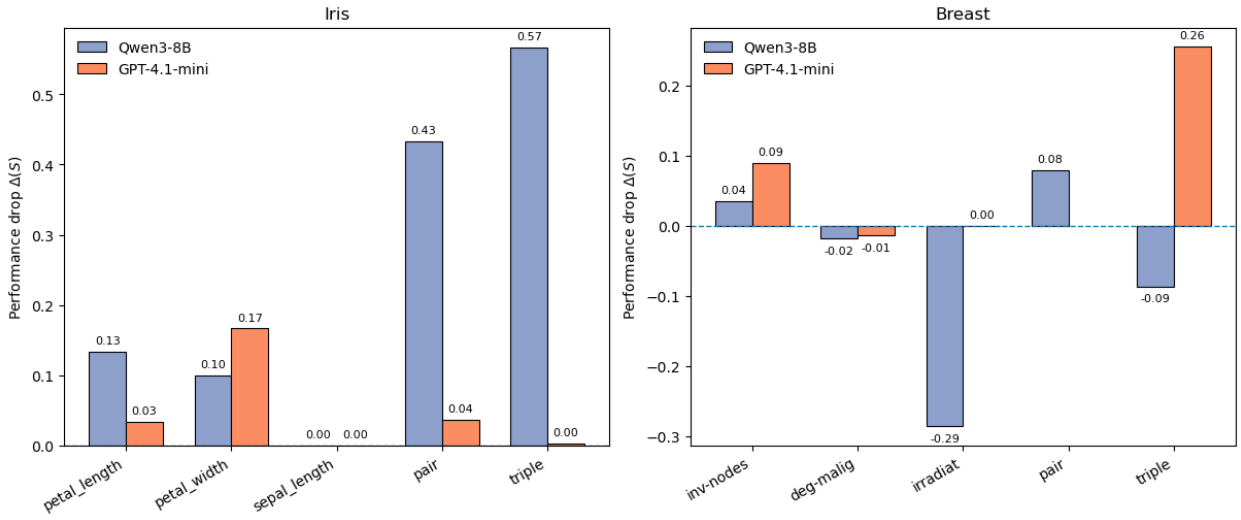


Figure 21: Standardized performance drops under single-feature and correlated group ablations. Larger values indicate stronger reliance on the removed feature set. Qwen3-8B on Iris shows a pronounced increase from single-feature to pair and triple ablations, consistent with joint reliance on correlated petal features. GPT-4.1-mini on Iris shows much weaker group effects, illustrating that higher-order reliance is model-dependent.

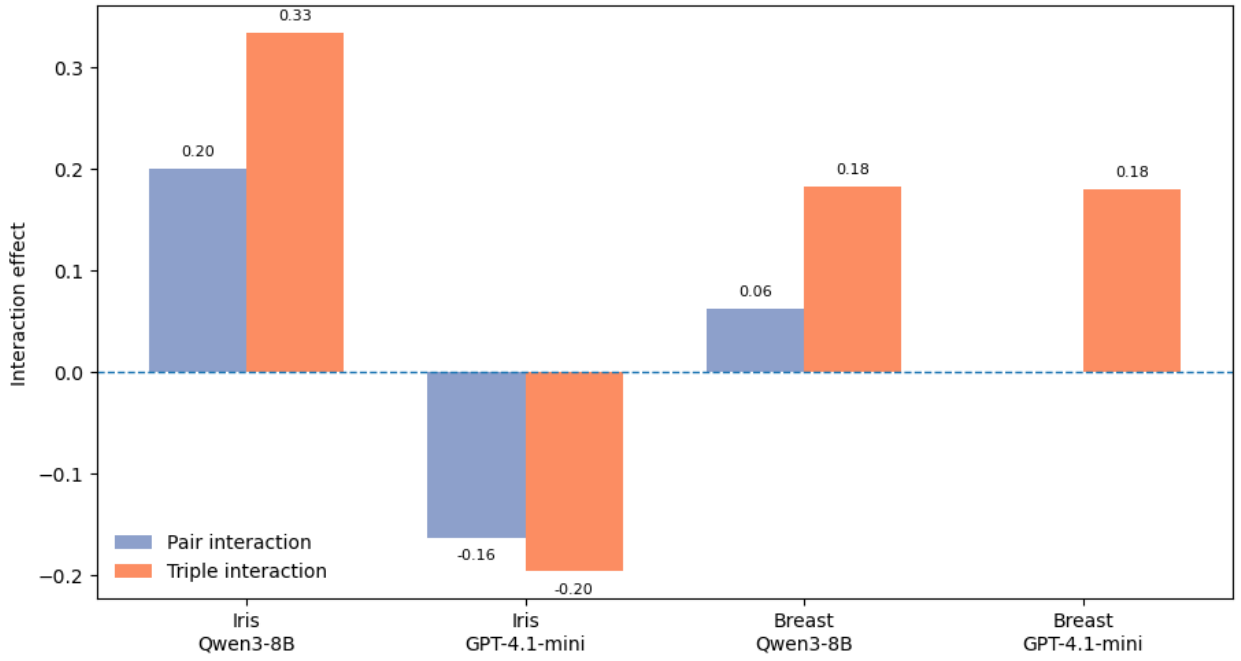


Figure 22: Non-additive interaction effects for correlated feature groups. Each value is computed as the observed group-ablation drop minus the sum of the corresponding single-feature drops. Positive values indicate super-additive reliance, whereas negative values indicate redundancy or overlap. The direction and magnitude of the interaction differ across models and datasets, showing that higher-order reliance is itself model-dependent.

Table 39: Group ablations on correlated feature subsets. $\Delta(f)$ denotes the standardized performance drop after removing feature f, while $\Delta(f_1, f_2)$ and $\Delta(f_1, f_2, f_3)$ denote drops under pair and triple removal. $I(\cdot)$ is the corresponding non-additive interaction effect relative to the sum of single-feature drops.

Dataset	Model	$\Delta(f_1)$	$\Delta(f_2)$	$\Delta(f_3)$	$\Delta(f_1, f_2)$	$\Delta(f_1, f_2, f_3)$	$I(f_1, f_2)$ / $I(f_1, f_2, f_3)$
Iris	Qwen3-8B	0.133	0.100	0.000	0.433	0.567	0.200 / 0.333
Iris	GPT-4.1-mini	0.033	0.167	0.000	0.037	0.003	-0.163 / -0.197
Breast	Qwen3-8B	0.036	-0.018	-0.286	0.080	-0.086	0.062 / 0.182
Breast	GPT-4.1-mini	0.090	-0.013	0.000	0.205	0.256	0.129 / 0.180

For Iris, f_1 = `petal_length`, f_2 = `petal_width`, and f_3 = `sepal_length`. For *Breast Cancer*, f_1 = `inv-nodes`, f_2 = `deg-malig`, and f_3 = `irradiat`.

Table 40: MedGemma-4B case-study summary on two healthcare datasets. We compare the model's self-reported decision factors with behavioral reliance measured by deletion-LAO. The table reports the most salient self-attributed features, the features with strongest measured LAO effects, and the resulting self–LAO alignment.

Dataset	Self-attributed factors	Strongest LAO factors	Alignment	Main observation
Breast Cancer	`inv-nodes`, `tumor-size`, `age`, `node-caps`, `deg-malig`	`age`, `tumor-size`, `deg-malig`, `breast`	Weak positive	MedGemma identifies medically interpretable variables, but only partially overlaps with measured reliance.
Pima Diabetes	`Glucose`, `Insulin`, `Pregnancies`, `Age`, `BMI`	`SkinThickness`; weak or negative effects for several self-attributed features	Negative / weak	Stated importance diverges from behavioral reliance; some self-attributed factors do not show corresponding LAO dependence.