Interpretable Probability Estimation with LLMs via Shapley Reconstruction

Anonymous ACL submission

Abstract

Large Language Models (LLMs) demonstrate potential to estimate the probability of uncertain events, by leveraging their extensive knowledge and reasoning capabilities. This ability can be applied to support intelligent decision-making across diverse fields, such as financial forecasting and preventive healthcare. However, directly prompting LLMs for probability estimation faces significant challenges: their outputs are often noisy, and the underlying predicting process is opaque. In this paper, we propose **PRISM: Probability Reconstruction via Shapley Measures**, a framework that brings transparency and precision to LLM-based probability estimation. PRISM decomposes an LLM's prediction by quantifying the marginal contribution of each input factor using Shapley values. These factor-level contributions are then aggregated to reconstruct a calibrated final estimate. In our experiments, we demonstrate PRISM improves predictive accuracy over direct prompting, and performs comparably with strong baselines across multiple domains including finance and healthcare. Beyond performance, PRISM provides a transparent prediction pipeline: our case studies visualize how individual factors shape the final estimate, helping build trust in LLM-based decision support systems.

1 Introduction

Estimating the probability of uncertain events (Berger, 2013; Winkler et al., 2019) is a critical task for intelligent decision makings in various domains such as financial investment (Lathief et al., 2024), healthcare (Rajkomar et al., 2019), and emergency management (Rostami-Tabar and Hyndman, 2025). However, in many real-world scenarios, either high-quality datasets are unavailable or mature Machine Learning (ML) techniques are lacking. Besides, there could also appear temporarily unexpected factors which are not considered when people building datasets (Chowdhury et al., 2021). As an example, in business analytics (Raghupathi and Raghupathi, 2021), people may encounter a variety of diverse estimation tasks, such as determining whether the price, production or market demand of a certain product will increase or not (Fildes et al., 2022). It could be difficult for them to collect sufficient task-specific data and build reliable predictive models promptly. In this scenario, the applicability of traditional ML approaches are severely limited.

As a potential solution, Large Language Models (LLMs) offer a promising alternative to address this challenge (Feng et al., 2024; Sui et al., 2024; Chung et al., 2024). They incorporate extensive world knowledge and exhibit strong reasoning capabilities (Wei et al., 2022), making them particularly valuable in data or model scarce settings. However, directly prompting LLMs to probability estimation still faces tremendous challenges: (i) LLMs typically produce noisy probability estimates that lack accuracy and stability. As an instance, according to the study (Nafar et al., 2025), when LLMs are asked to predict the same event in different forms, i.e., "whether it will happen" and "whether it will not happen", LLMs could provide conflicting answers. (ii) The "black-box" generative process provides little transparency regarding how each individual factor contributes to the final prediction, making the results difficult to interpret. As illustrated in Figure 1, when asked to predict the likelihood of a person having a certain disease, LLMs will not explicitly show how much weight each factor contributes to final prediction, but outputs a single score (with a partial explanation). It makes the final outcome difficult to interpret and less trustworthy.

To overcome the challenges, we propose a novel framework **Probability Reconstruction via Shapley Measures (PRISM)**, inspired by Shapley Value for ML explanation (Lundberg and Lee, 2017). Specifically, we decompose the estimation task into quantifying the marginal contribution of each fac-

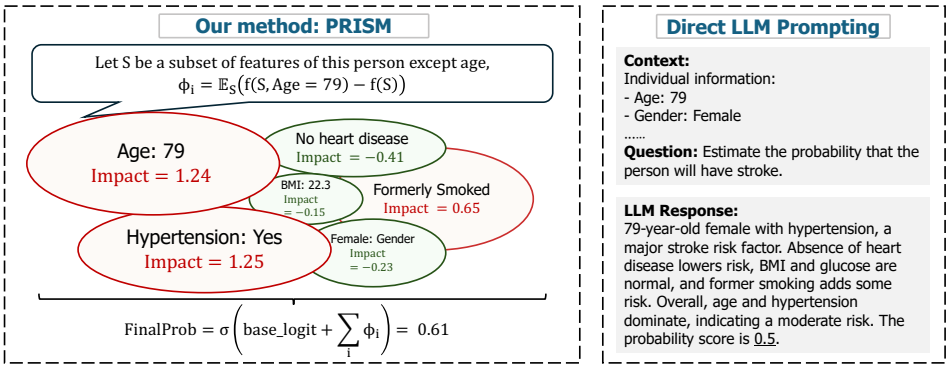


Figure 1: Illustration of direct LLM prompting and PRISM. PRISM first estimates Shapley values (factor contributions) and aggregates them to reconstruct final probability. We use red to represent the factors found by PRISM to have positive contribution to the positive outcome (have a stroke), green are factors found to be negative. The size reflects the contribution's absolute value. $f(\cdot)$ is from LLM prediction and S is a background set, $\sigma(\cdot)$ is the sigmoid function (see details in Section 3).

tor independently. As illustrated in Figure 1, consider the task of predicting whether a person will experience a stroke. For a given factor such as "Age=79", we compare the LLM's estimated probability when the model has access to this factor versus when it does not have access to it, paired with a random subset S of the remaining factors (see Section 3 for details). The *marginal contribution of a factor*, referred to as its *Shapley value*, is computed by averaging over different subsets S. These values capture both the direction and the intensity of the features' contribution to the prediction outcome. However, unlike traditional Shapley methods which solely focus on model output attribution (Lundberg and Lee, 2017), PRISM goes further: we reconstruct a new probability estimate by aggregating the factor contributions. For example, in Figure 1, the aggregated contributions yield a probability of 0.61, and we can clearly interpret this estimate originates from key risk-increasing factors such as advanced age and having hypertension. This process can make the prediction process transparent and allow human users to diagnose and interpret the model's reasoning.

In our experiments, we validate the effectiveness of our proposed method under various settings. In Section 4.1, we compare PRISM with other LLM-based probability estimation methods, on binary classification tasks under benchmark tabular datasets (such as Adult Census Income (Becker and Kohavi, 1996), Heart Disease Prediction (Chicco and Jurman, 2020), Stroke Prediction (Kaggle, 2021) and Loan Default Prediction (LendingClub, 2007–2018)). We find that PRISM demonstrates higher prediction performance than directly LLM prompting and performs similarly to stronger baselines. Furthermore, in Section 4.2, we evaluate PRISM on real-world prediction tasks involving textual and numerical inputs. For instance, we predict whether the a patient will be re-admitted to hospital, based on the doctor notes. Together, these studies demonstrate PRISM's ability to handle heterogeneous, context-rich factors—well beyond simple tabular data. Overall, the *experiments highlight PRISM's potential as a practical and reliable tool* for a wide range of real-world prediction problems.

2 Related Works

2.1 Probability Estimation via Traditional ML and LLMs

Traditional machine learning models such as Bayesian inference (Berger, 2013), multilayer perceptrons (MLPs) (Rumelhart et al., 1986) and decision trees (Quinlan, 1986) have long been widely used for probability estimation, typically cast as binary classification tasks on tabular data. These methods often rely on large amount of data and well-designed models. To overcome this limitation, recent works have explored using Large Language Models (LLMs) for probability estimation by leveraging their rich prior knowledge. In this paper, we focus on LLM-based probability estimation **under zero-shot setting** which makes estimations only based on the LLM's own knowledge. Under this setting, previous studies validate the potential especially for tabular format datasets (Sui et al., 2024; Chung et al., 2024; Ren et al., 2025; Xie et al., 2024). However, although many of these methods show decent estimation precision, their estimation process usually relies on direct LLM prompting. Thus, it can be difficult for the users to exactly measure the contribution of each factor to the final prediction, different from the case in traditional ML models. As a more relevant paper

to our work, BIRD (Feng et al., 2024) is proposed to integrate Bayesian networks (Berger, 2013) to explicitly calculate the probability of the prediction outcome conditioning on each factor. However, it assumes the factors are independent to each other, and it has to transform the factors into categorical values. Notably, beyond the zero-shot setting, there are related studies focusing on few-shot prediction with LLMs (Hegselmann et al., 2023; Brown et al., 2020). However, in such cases, the interpretation can become more complicated, as it should disentangle whether a prediction arises from the own knowledge of LLMs or from the provided demonstrations. Therefore, we leave the exploration of few-shot prediction tasks to future work.

2.2 Explainable probability estimation

For traditional ML explanation, Shapley-value (Shapley et al., 1953; Lundberg and Lee, 2017; Štrumbelj and Kononenko, 2014) is widely used to attribute predicted probabilities to input factors. In general, it quantifies each factor's marginal contribution to the final prediction outcome. Building on this line, recent works adapt Shapley values to explain the behavior of LLMs. For instance, TokenSHAP (Goldshmidt and Horovicz, 2024) estimates token-level contributions to LLM outputs in general question answering tasks. SyntaxSHAP (Amara et al., 2024) moves beyond token-level attributions by capturing the contribution of higher-level syntactic units. Unlike these studies, our work specifically targets on probability estimation tasks, where we quantify the contributions of each "influencing factor" in the probability estimation. Besides,, our approach goes beyond conventional LLM or ML interpretation tools. We reconstruct new probability estimates through a principled approach of factor aggregation, rather than only interpreting the original model outputs.

3 Methodology

In this section, we formally introduce and describe our proposed method Probability Reconstruction via Shapley Measures (PRISM). In Section 3.1, we first introduce the necessary notations and definitions, with particular emphasis on the Shapley value and its importance for ML explanations. Section 3.2 provides a description of the PRISM algorithm, clarifying how factor contributions are combined to reconstruct probabilities. Finally, Section 3.3 and Section 3.4 introduces an efficient variant of PRISM specifically designed for tabular tasks, enabling faster computation in such scenarios.

3.1 Definitions and Notations

Notations. For a probability estimation task, we denote an **instance** by $x = (x_1, \ldots, x_m)$ and its true outcome by $y_{true} \in \{0, 1\}$, where each x_i is a factor that can influence the outcome. We let $f(\cdot)$ denote a general model output, which may come from either a traditional ML binary classifier or from LLM-based predictions. Depending on the setting, $f(\cdot)$ can represent either the predicted probability or the model logit (before sigmoid function). In general, it is desirable to obtain the estimation $f(x)$ to be close or aligned to y_{true}. In our paper, an important concept is Shapley value (Shapley et al., 1953) that applied in ML prediction explanation (Lundberg and Lee, 2017). In the next, we first introduce the basic definition of Shapley value (for ML models) in general, and we discuss one important property.

Definition 1 (Shapley value)**.** *Let* $\mathcal{I}=\{1, 2, \ldots, m\}$ *denote the index set of factors, and let* $x = (x_1, \ldots, x_m)$*. The Shapley value of factor* $i \in \mathcal{I}$ *for instance* x *with respect to model* f *is:*

$$\phi_i(f, x) = \sum_{S \subseteq \mathcal{I} \setminus \{i\}} \frac{|S|!(m-|S|-1)!}{m!} \left[f(x_{S \cup \{i\}}) - f(x_S) \right], \quad (1)$$

where each S *is a subset of factors* $\mathcal{I}$ *excluding* i*, and the sum is taken over all* $S \subseteq \mathcal{I} \setminus \{i\}$*,* $f(x_S)$ *is the model output when only the factors in* S *are used (and* $f(x_{S \cup \{i\}})$ *is defined analogously). In our paper, we name each* S *as a "**background set**".*

Intuitively, Shapley value $\phi_i(f, x)$ measures the *marginal contribution* of factor i to the model output, by comparing the model's output when factor i is "included in the background set S" versus when it is "not included in the background set S", across all possible cases of S. In this way, one can interpret the model's prediction by separately examining the contribution of each individual factor. In practice, for traditional ML, the difference $\left(f(x_{S \cup \{i\}}) - f(x_S)\right)$ can be computed either by retraining models on different subsets of factors (Lipovetsky and Conklin, 2001), or by approximating the effect of missing factors using training data (Štrumbelj and Kononenko, 2014). Next, we discuss one important property of Shapley value in traditional MLs which inspires our method.

Property 1. *Additivity (Lundberg and Lee, 2017) Let* $\phi_0 = f(x_\emptyset)$ *be the expected model output when no factor information is available. Then, for models with deterministic scalar outputs, the sum of* ϕ_0 *and the Shapley value of the factors* $\phi_i(f, x)$ *must*

reconstruct *the model prediction itself.*

$$f(x) = \phi_0 + \sum_{i=1}^{m} \phi_i(f, x). \quad (2)$$

That is, the explanation exactly matches the original model output.

It suggests for traditional ML models, the output $f(x)$ can be decomposed to the sum of Shapley values and the base logit ϕ_0. However, Property 1 does not extend easily to Large Language Models (LLMs). This is because, when LLMs make predictions for uncertain events, they express their probability estimates through generated text (see Figure 1 (right)). Such expressed probabilities differ from the direct model outputs, such as the token probabilities produced by the LLM architectures.

3.2 PRISM: Probability Reconstruction via Shapley Measures

Nevertheless, inspired by Property 1, we devise our new estimation framework: we first obtain the Shapley values for each factor and then aggregate them to reconstruct a new prediction outcome. In the next, we elaborate two important steps to obtain the Shapley values through LLMs.

Comparative LLM Prompting. Given a factor of interest, such as "i = Age is 79" in the stroke prediction task, we first sample multiple background sets S from $\mathcal{I} \setminus \{i\}$ (following the permutation sampling strategy in the next part). Each S contains information of several factors that different from i, such as "hypertension, female and BMI=22.3" illustrated in Figure 2. Next, for a given background set S, we prompt the LLM to evaluate both $x_{S\cup\{i\}}$ and x_S in the same query. We denote $p(\cdot)$ as the LLM estimated probability for a given instance (e.g., by responding to a question as in Figure 2). We let $f(\cdot) = \sigma^{-1}(p(\cdot))$, where $\sigma^{-1}(\cdot)$ is the inverse sigmoid function, we have:

$$f(x_{S\cup\{i\}}) - f(x_S) = \sigma^{-1}\big(p(x_{S\cup\{i\}})\big) - \sigma^{-1}\big(p(x_S)\big), \quad (3)$$

This calculates the difference term in Shapley value defined in Eq.(1). We argue that: the single estimates derived by directly LLM prompting may not be exact, but the relative relation between the two cases can be well-captured. Recent studies also find that LLMs tend to perform more reliably on ranking or comparison tasks than absolute probability estimation (Qin et al., 2023; Liu et al., 2024), which supports the validity of our design.

Permutation Sampling. In PRISM, at each iteration we sample a background set S using the permutation sampling rule (Shapley et al., 1953; Štrumbelj and Kononenko, 2014). Concretely, we generate a random permutation of all factors in $\mathcal{I}$ including i, and choose S as the set of factors that precede i in this order. Then, we directly average the pair-wise differences (from Eq.(3)) across multiple (e.g., K times) such samplings to approximate the Shapley value:

$$\phi_i = \frac{1}{K}\sum_{k=1}^{K}\Big[\sigma^{-1}\big(p(x_{S^{(k)}\cup\{i\}})\big) - \sigma^{-1}\big(p(x_{S^{(k)}})\big)\Big], \quad (4)$$

The equivalence of Eq.(4) and Eq.(1) is shown in (Shapley et al., 1953). In practice, it is necessary to sample S for multiple times, ensuring each background factor is considered with a non-negligible probability. This will lead to a more comprehensive estimation by accounting for possible factor interactions (see Section 4.3 for further discussion). Finally, we set ϕ_0 as the base logit, either from the population average (e.g., $\phi_0 = \sigma^{-1}$(Average stroke prevalence), or simply set to 0 when no prior knowledge is available in other tasks. We then reconstruct the probability estimation as:

$$p_{\text{PRISM}}(x) = \sigma(\phi_0 + \sum_{i=1}^{m} \phi_i). \quad (5)$$

We provide the algorithm sketch in Appendix A.1.

Case A:
S = {has hypertension, female, BMI 22.3}
Case B:
S ∪ {i} = {has hypertension, female, BMI 22.3, Age = 79}
Question: Estimate the probability of each person that having stroke?

Figure 2: Comparative Prompting.

3.3 Tabular-PRISM

In this subsection, we discuss a variant of PRISM, when the factors can be represented as single values or categories and integrated in tabular datasets. For such tasks, instead of calculating the difference term in Eq.(3) for each S in a separate LLM query, we can consider multiple background sets in **one single query**. Refer to Figure 3a, we present x_S and $x_{S\cup\{i\}}$ for different S in the same table (adjacent rows are from the same S). Then, we prompt LLMs to evaluate each of them in one query. In this way, the query and token efficiency can be significantly improved. However, a challenge is that: if we leave those factors that are not presented as blank, "unknown" or "not provided", it will introduce bias to the LLM judgment. For example, LLM can over-estimate a person's risk of certain disease only because some information is "unknown".

To solve this issue, we leverage the strategy to introduce a **reference instance**, which is also

Age	Hyper.	Heart Disease	BMI
	Yes		22.3
79	Yes		22.3
		No	22.3
79		No	22.3
	Yes		
79	Yes		

(a) With blanks.

Age	Hyper.	Heart Disease	BMI
40	Yes	No	22.3
79	Yes	No	22.3
40	No	No	22.3
79	No	No	22.3
40	Yes	No	24.0
79	Yes	No	24.0

(b) With reference.

Figure 3: When calculating the Shapley value of the factor "Age=79", we put multiple S in one table. Factor values from reference instances are noted in green.

used in Shapley ML explanation (Štrumbelj and Kononenko, 2014). Specifically, we let r be a reference instance with each factor a fixed value (typically obtained from population average or majority). Then, use the information of this reference instance to impute missing factors, which is illustrated in Figure 3b, and calculate the Shapley values (with a new definition) based on this new table.

Definition 2. *For background set* S*, we define the model output of the imputed sample as* $v_r(S) = f([x_S, r_{\bar{S}}])$*, where* $\bar{S} = \mathcal{I} \setminus S$*, which means the factors in* S *are provided by* x*, and the remaining are provided by* r*. Then, for each factor* x_i*, the (reference-specific) Shapley value is*

$$\phi_i^{(r)} = \sum_{S \subseteq \mathcal{I} \setminus \{i\}} \frac{|S|!(m-|S|-1)!}{m!} \Big(v_r(S \cup \{i\}) - v_r(S) \Big). \quad (6)$$

This definition differs from Eq.(1) and focuses on comparisons against the reference sample. For instance, a large positive $\phi_i^{(r)}$ of "Age = 79" implies this factor substantially raises stroke risk relative to the reference age 40. Despite interpretive differences, predictions can still be reconstructed using these Shapley values.

Proposition 1. *Fix an instance* x *and a single reference sample* r*. Let* $\phi_i^{(r)}$ *be the Shapley value in Definition 2. Then, for models with deterministic scalar outputs, with* $\phi_0^{(r)} = v_r(\emptyset)$*, we still have:*

$$f(x) = v_r(\mathcal{I}) = \phi_0^{(r)} + \sum_{i=1}^{m} \phi_i^{(r)} \quad (7)$$

Similar to Property 1, this proposition indicates LLM-based estimation can be reconstructed by aggregating Shapley values: $p_{\text{PRISM}}(x) = \sigma(\phi_0^{(r)} + \sum_{i=1}^{m} \phi_i^{(r)})$. Detailed algorithm and proof are provided in Appendix A.1 and A.2.

3.4 Factor Extraction

In PRISM, it is also essential to determine which factors should be considered during prediction, especially for textual, unstructured inputs. We adopt two criteria for selecting these factors: **(1) Non-overlap**, which avoids redundancy by ensuring that the same information is not repeatedly considered, and **(2) Completeness**, which ensures that the union of all factors covers the necessary information used by the LLM. In practice, for tabular tasks, factors are directly obtained from the provided tabular features after removing duplicates. For unstructured tasks, we use an automated pipeline. We first query an LLM to propose the minimal set of aspects required to complete the task, with the constraints that these aspects must be non-overlapping and complete. For each aspect, we then extract a summary from the context, which serves as a "factor" within the PRISM framework.

4 Experiment and Discussion

In this section, we conduct experiments to validate the effectiveness of PRISM. In Section 4.1, we compare the estimation performance of PRISM and baselines on benchmark (tabular) datasets. Meanwhile, we provide case analysis to visualize the interpretation outcome of PRISM. In Section 4.2, we demonstrate that PRISM can also be applied in real-world estimation tasks which cannot be easily formed into tabular format. Finally, Section 4.3 provides a series of additional studies.

4.1 Probability Estimation on Benchmark Tabular Datasets

We first evaluate our method focused on benchmark **tabular** datasets. This is because probability estimation tasks are more frequently built in tabular format, for traditional ML studies. Therefore, we have various datasets with sufficient samples for fair and comprehensive comparisons. We later discuss scenarios beyond tabular format in Section 4.2.

Experiment setup. We involve four representative tabular datasets ranging from disease prediction, income prediction, and loan credit estimation. In details, Adult Census Income (Becker and Kohavi, 1996) is to predict whether a person has annual income over $50K, based on their occupation, education and family information. Stroke (Kaggle, 2021) and Heart Disease (Chicco and Jurman, 2020) predict whether a certain disease will appear based on the patients' health status. Lending (LendingClub, 2007–2018) predicts whether a loan default will happen, based on the loan application records and applicants' personal information. More details and pre-process procedures are in Appendix B.1.

In this study, we consider the zero-shot set-

tings where the estimations are solely relied on the LLM's own knowledge. For our method[1], we implement Tabular-PRISM (in Section 3.3) and sample background set S for 10 times for each Shapley value (see impacts of this number of sampling on the performance in Appendix C.5). Besides, we consider baselines:

- Directly prompting LLMs to predict likelihood **levels**. For example, we ask LLM to choose one in the options from "very unlikely" to "very likely". We also try multiple shots to obtain the self-consistency result (Wang et al., 2022) by making votes. They are denoted as "1shot_level, 5shot_level, 10shot_level" in Table 1.
- Directly prompting LLMs to predict likelihood **scores**, which are probability scores between [0-1], and we obtain the self-consistency result by taking their average. They are denoted as "1shot_score, 5shot_score, 10shot_score".
- *Contrast* (motivated by Nafar et al. (2025)) asks about the likelihood in the positive and the negated question form, and then unify the answers to get the final estimation.
- *BIRD* (Feng et al., 2024) builds Bayesian networks to evaluate the probability of the estimation outcome conditioning on various (categorized) factors.
- We also add In-Context Learning (ICL, (Brown et al., 2020)), which is beyond zero-shot setting. We randomly select 5 positive and 5 negative samples as demonstrations for each instance (or 10 positive and 10 negative respectively).

For each setting and method, we conduct experiments on GPT-4.1-mini (OpenAI, 2024) and Gemini-2.5-Pro (Google DeepMind, 2025). All runs use temperature 1.0 by default.

Experiment result. In our result shown in Table 1, for each dataset, we randomly choose 300 test samples (150 positive and 150 negative), and we report AUROC (shown as "ROC" in Table 1), AUPRC (shown as "PRC"), and the best F1 (when select the threshold for maximized TPR + TNR). From the result, we can see the PRISM consistently demonstrates reliable estimation performance. Specifically, PRISM presents highest performance or it is close to the strongest baselines in most datasets, especially under Adult Census, Heart Disease and Lending, under both LLMs. In these datasets, we see PRISM has more obvious improvement in AUROC and AUPRC than F1-scores. This suggests that the baselines can sometimes effectively separate positive and negative samples when an appropriate threshold is chosen, whereas the high AUROC and AUPRC of PRISM indicate its strong discriminative ability across all possible thresholds.

Under Stroke Dataset, our method is comparable to strong baselines or slightly lower than those strong baselines. For example, under Gemini-2.5-pro, "1shot_score" has higher AUROC, AUPRC and F1 than PRISM. We conjecture it may be because the LLM itself has well-educated knowledge in the relevant domain. However, PRISM demonstrates stable performance across various datasets and shows competency against the strongest baselines, if not surpassing them. In Appendix C, we show PRISM's predicted probability also has **a good calibration** (Bella et al., 2010) if a proper base logit is selected.

Examination of Interpretations. In Table 2, we present instances under Stroke dataset to help understand the interpretation process of PRISM. In detail, we present the factor values as well as the estimated Shapley values (Eq.(6)). Notably, the Shapley value here represents the relative contribution of each factor, comparing to the reference instance (see Section 3.3). For example, in Case 1, a Shapley value of 0.66 for "Heart Disease" suggests: compared with "No Heart Disease" (from reference instance), this factor increases the risk of having stroke. Similarly, in Case 2, because the person does not have heart disease which is the same as reference instance, the Shapley value is 0. In Appendix C, we provide the interpretation results similar to Table 2 of other datasets.

4.2 Probability Estimation on Unstructured Data

Beyond tabular data, there are often cases where the influencing factors cannot be easily transformed into single values to be inputted into tables. For example, the factors can be in form of descriptive facts that are extracted from news articles, financial reports, or social media. Probability estimation in this scenario is also of great importance.

Predicting 30-day Readmission after Discharging. We evaluate an unstructured-text prediction task on the MIMIC-III dataset (Johnson et al., 2016), using discharge summaries (doctor notes) to

[1]We set the factor values in the reference instance to be around the population average (for continuous values), and population majority (for categorical values). We query an LLM for the base logit ϕ_0. In practice, one can choose ϕ_0 from more reliable sources such as historical statistics or human experts, and the selection of ϕ_0 will not impact our evaluation of AUROC, AUPRC and F1.

Model / Method	Adult Census			Heart Disease			Stroke			Lending		
	ROC	PRC	F1	ROC	PRC	F1	ROC	PRC	F1	ROC	PRC	F1
GPT-4.1-mini												
1shot_level	0.777	0.773	0.709	0.722	0.706	0.615	0.767	0.694	0.726	0.629	0.606	0.478
5shot_level	0.795	0.779	0.701	0.759	0.723	0.664	0.780	0.716	0.730	0.617	0.576	0.648
10shot_level	0.799	0.779	0.701	0.759	0.718	0.672	0.792	0.727	0.732	0.627	0.591	0.443
1shot_score	0.795	0.792	0.709	0.799	0.761	0.772	0.804	0.763	0.753	0.612	0.600	0.558
5shot_score	0.819	0.820	0.755	0.807	0.779	0.782	**0.816**	0.783	0.780	0.629	0.621	0.612
10shot_score	0.816	0.820	0.734	0.806	0.776	**0.793**	0.813	0.781	0.785	0.636	**0.631**	0.621
ICL-5+5	0.807	0.788	0.707	0.803	0.761	0.762	0.801	0.751	0.775	0.598	0.567	0.668
ICL-10+10	0.754	0.758	0.645	0.776	0.744	0.760	0.788	0.736	0.763	0.591	0.569	0.642
Contrast	0.790	0.813	0.697	0.769	0.729	0.738	0.790	**0.813**	0.697	0.485	0.543	0.169
BIRD	0.813	0.804	0.730	0.777	0.748	0.712	0.778	0.740	0.729	0.610	0.564	0.533
PRISM (Ours)	**0.851**	**0.874**	**0.770**	**0.816**	**0.799**	**0.793**	0.814	0.783	**0.790**	**0.655**	0.626	**0.671**
Std. Err. (Ours)	0.0215	0.0221	0.0296	0.0254	0.0350	0.0250	0.0311	0.0430	0.0283	0.0313	0.0428	0.0300
Gemini-2.5-Pro												
1shot_level	0.818	0.797	0.699	0.732	0.668	0.780	0.793	0.744	0.720	0.555	0.538	0.518
5shot_level	0.822	0.794	0.694	0.758	0.686	0.794	0.804	0.749	0.727	0.541	0.526	0.511
10shot_level	0.821	0.797	0.688	0.738	0.663	0.791	0.803	0.745	0.716	0.564	0.543	0.571
1shot_score	0.855	0.876	0.767	0.812	0.739	0.797	**0.836**	0.812	0.802	0.536	0.538	0.465
5shot_score	0.864	0.879	0.823	0.816	0.749	0.795	0.834	0.810	**0.804**	0.529	0.534	0.533
10shot_score	0.864	0.878	**0.826**	0.815	0.753	**0.799**	0.835	**0.814**	0.803	0.528	0.526	0.547
ICL-5+5	0.842	0.834	0.744	0.834	0.785	0.770	0.815	0.767	0.728	0.615	0.576	0.575
ICL-10+10	0.812	0.793	0.722	0.807	0.756	0.739	0.818	0.776	0.738	0.626	0.584	**0.690**
Contrast	0.835	0.830	0.757	0.831	0.779	0.789	0.806	0.768	0.727	0.593	0.588	0.520
BIRD	0.799	0.794	0.754	0.810	0.767	0.762	0.803	0.779	0.735	0.555	0.528	0.625
PRISM (Ours)	**0.876**	**0.893**	**0.826**	**0.844**	**0.832**	0.787	0.826	0.788	0.783	**0.654**	**0.614**	0.685
Std. Err. (Ours)	0.0198	0.0194	0.0239	0.0220	0.0298	0.0257	0.0245	0.0369	0.0252	0.0314	0.0426	0.0277

Table 1: Performance comparison across various tabular datasets. The highest and second-highest results are in red and orange.

Factor	Case 1		Case 2		Case 3		Case 4	
	Value	Shapley	Value	Shapley	Value	Shapley	Value	Shapley
Gender	Female	-0.08	Female	0.00	Male	0.00	Female	0.00
Age	82	1.17	52	0.57	68	1.15	29	-0.77
Hypertension	Yes	1.64	Yes	1.41	No	0.00	No	0.00
Heart Disease	Yes	0.66	No	0.00	Yes	1.17	No	0.00
Marital Status	Never Married	0.00	Ever Married	0.00	Ever Married	0.00	Ever Married	0.00
Work Type	Government job	0.00	Private sector	0.00	Private sector	0.00	Private sector	0.00
Residence Type	Rural	0.00	Urban	0.00	Urban	0.00	Urban	0.00
Glucose Level	84.03	0.00	94.98	0.51	223.83	0.62	116.98	0.98
BMI	25.60	0.00	23.80	0.00	31.90	0.53	23.40	0.00
Smoking Status	Smokes	0.85	Never smoked	0.00	Formerly smoked	0.60	Never smoked	0.00
Sum Shapley	4.240		2.490		4.070		0.210	
Sum logit	-0.355		-2.105		-0.525		-4.385	
Predicted prob	0.413		0.109		0.372		0.012	
True label	Yes		No		Yes		No	

Table 2: Shapley values of 4 instances in Stroke. Reference instance: *gender=Male; age=40; hypertension=No; heart disease=No; marital status=Never Married; residence=Rural; average glucose=90.0; BMI=24.0; work type=Private; smoking status=never smoked*. A same base logit is set: $\phi_0 = \sigma^{-1}(0.01)$=-4.5951.

Method	GPT-4.1-mini			Gemini-2.5-Pro		
	ROC	PRC	F1	ROC	PRC	F1
1shot_level	0.62	0.57	0.57	0.64	0.63	0.60
5shot_level	**0.64**	0.63	0.66	0.66	0.67	0.66
1shot_score	**0.64**	0.61	0.50	0.66	0.66	0.59
5shot_score	**0.64**	0.62	0.63	**0.67**	**0.68**	**0.67**
Contrast	0.63	0.59	0.56	0.65	0.64	0.60
PRISM	0.63	**0.65**	**0.69**	0.66	0.67	**0.67**

Table 3: Performance on MIMIC-III dataset. The highest and second-highest results are in red and orange.

predict whether a patient will be readmitted to the hospital within 30 days. In PRISM, we consider eight distinct factors, including the patients' overall health status and their recent hospital course, and so on. We evaluate 200 samples (100 positive and 100 negative cases) with GPT-4.1-mini and Gemini-2.5-Pro and compare PRISM against directly prompting and Contrast baselines. Results in table 3 show that PRISM consistently achieves higher performance, particularly on PRAUC and F1. Overall, PRISM substantially outperforms single-shot direct prompting and achieves performance comparable to multiple-sampling baselines. For direct-prompting baselines, we find that LLMs such as GPT-4.1-mini often provide the same values (0.35 and 0.65) for different samples, reducing their ability to provide more fine-grained ranking. Moreover, Figure 4 shows an attribution example on a true readmission case. Direct prompting yields only a coarse estimate (0.35) without explaining the underlying clinical drivers. In contrast, PRISM produces a calibrated prediction (0.79) and distributes contributions across eight clinical factors, revealing how baseline disease burden, treatment intensity, psychosocial risks, and post-discharge complexity elevate the patient's risk.

4.3 Additional Studies

In this subsection, we study important questions regarding PRISM: (1) Can it handle feature interaction (Hall, 1999)? (2) How is the computational efficiency of PRISM? In appendix, we also conduct analysis to study: (3) Can PRISM generalize to multi-class prediction? (Appendix C.3); (4) How is PRISM performance compared to linear models if there existing (tabular) training data? (Appendix C.4)

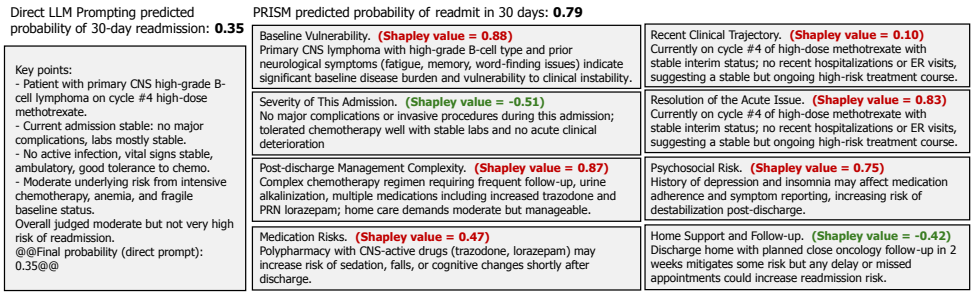


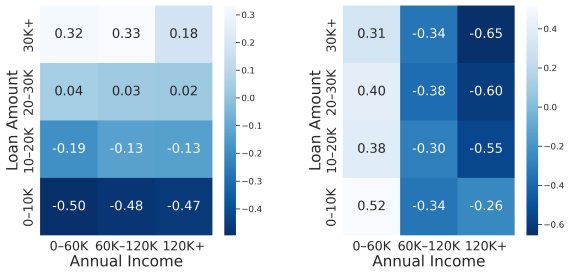
Figure 4: Example of PRISM attribution in MIMIC-III (True Label=1).

Metric	Query Complexity	LLM Evaluation Complexity
1-shot	$\Theta(1)$	$\Theta(1)$
n-shot	$\Theta(n)$	$\Theta(n)$
Tabular-PRISM	$\Theta(m)$	$\Theta(mK)$
PRISM	$\Theta(mK)$	$\Theta(mK)$

Table 4: Complexity of PRISM and baselines.

Can PRISM handle feature interaction? In ML probability estimation, feature interaction usually happens as the contribution of one factor to the outcome also highly depends on the condition of another factor. Taking this into consideration is necessary for precise and reliable estimation. Figure 5 demonstrates that PRISM can indeed consider feature interactions when calculating the Shapley values and the final estimation. In detail, we focus on the task to predict loan default in Lending dataset under GPT-4.1-mini (see Section 4.1), where feature interactions can naturally occur between "Loan Amount" and "Annual Income". In Figure 5a, we compute the average Shapley value of Loan Amount across different instances, conditioned on loan amount (vertical axis) and annual income (horizontal axis). From the result, we can see that the individuals with annual income 120K+ receive a Shapley value of 0.18 for the factor "having a loan amount over 30K+". It is lower than the values assigned to people with income in 0–60K or 60K–120K. This indicates that, within PRISM, for individuals earning above 120K, having a loan amount over 30K+ **is not considered as risky as those with lower incomes**. Similarly, Figure 5b computes the average Shapley value of Annual Income. We can see PRISM believes that having a factor "Annual Income over 120K+" can greatly reduce the risk (-0.65), if their loan amount is over 30K+, and it only moderately reduces the risk (-0.26), if the loan amount is low, e.g., below 10K. It shows a Shapely value in PRISM does not only rely on its factor of interest but also other factors.

Computational efficiency of PRISM. In this part, we analyze the computational efficiency of PRISM. Table 4 summarizes two notions of complexity: *Query Complexity* counts the number of API requests that each method issues, and *LLM Evaluation Complexity* counts the number of instances that each method need to evaluate. In the table, we compare PRISM with 1-shot direct prompting and n-shot direct prompting. For PRISM, it needs to calculate Shapley values for m factors one by one. For each factor, it makes K samplings of background set for comparison. Therefore, it has a query and evaluation complexity $\Theta(mK)$. Tabular-PRISM can input multiple S samplings into one query, so it has a query complexity $\Theta(m)$. In practice, we argue that Tabular-PRISM strategy can greatly reduce the time and token cost, as it saves API calls and evaluate multiple instances in one answer. For the actual time cost, in Stroke dataset (with m = 10, K = 10), Tabular-PRISM takes 92.7s for each instance on average under GPT-4.1-mini non-batched API calling. Apple price prediction using PRISM (GPT-4.1, m = 7, K = 5) takes around 330s on average, due to large amount of queries, long inputs (each factor is a paragraph) and larger model size. This efficiency can be acceptable if the evaluation size is not large.



(a) Loan Amount (b) Annual Income

Figure 5: Avg. Shapley under various conditions

5 Conclusion

In this work, we propose Probability Reconstruction via Shapley Measures (PRISM) for LLM-based probability estimation. PRISM reconstructs predictions from feature-level contributions, offering a more interpretable alternative to direct probability prompting. Experiments on multiple datasets demonstrate its effectiveness and suggest its potential for more transparent LLM-based prediction.

Limitations

This work has several limitations. First, our study focuses on the zero-shot setting. In real-world applications, historical records or labeled examples may be available, and extending PRISM to few-shot prediction remains an important future direction. However, few-shot settings may also complicate interpretation, since the model's prediction can be influenced by both its internal knowledge and the provided demonstrations. Second, PRISM requires multiple LLM queries to estimate feature-level contributions. As discussed in Section 4.3, this leads to higher computational cost than direct prompting, making the method more suitable for moderate-scale evaluation settings where interpretability and probability reliability are prioritized over inference efficiency.

References

Kenza Amara, Rita Sevastjanova, and Mennatallah El-Assady. 2024. Syntaxshap: Syntax-aware explainability method for text generation. *arXiv preprint arXiv:2402.09259*.

Barry Becker and Ronny Kohavi. 1996. Adult.

Antonio Bella, Cèsar Ferri, José Hernández-Orallo, and María José Ramírez-Quintana. 2010. Calibration of machine learning models. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, pages 128–146. IGI Global Scientific Publishing.

James O Berger. 2013. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Davide Chicco and Giuseppe Jurman. 2020. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making*, 20(1):16.

Priyabrata Chowdhury, Sanjoy Kumar Paul, Shahriar Kaisar, and Md Abdul Moktadir. 2021. Covid-19 pandemic related supply chain studies: A systematic review. *Transportation Research Part E: Logistics and Transportation Review*, 148:102271.

Philip Chung, Christine T Fong, Andrew M Walters, Nima Aghaeepour, Meliha Yetisgen, and Vikas N O'Reilly-Shah. 2024. Large language model capabilities in perioperative risk prediction and prognostication. *JAMA surgery*, 159(8):928–937.

Yu Feng, Ben Zhou, Weidong Lin, and Dan Roth. 2024. Bird: A trustworthy bayesian inference framework for large language models. *arXiv preprint arXiv:2404.12494*.

Robert Fildes, Shaohui Ma, and Stephan Kolassa. 2022. Retail forecasting: Research and practice. *International Journal of Forecasting*, 38(4):1283–1318.

Roni Goldshmidt and Miriam Horovicz. 2024. Tokenshap: Interpreting large language models with monte carlo shapley value estimation. *arXiv preprint arXiv:2407.10114*.

Google DeepMind. 2025. Gemini 2.5 pro — model card. Technical report.

Mark A Hall. 1999. *Correlation-based feature selection for machine learning*. Ph.D. thesis, The University of Waikato.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International conference on artificial intelligence and statistics*, pages 5549–5581. PMLR.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Kaggle. 2021. Stroke prediction dataset. https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset.

Jaheera Thasleema Abdul Lathief, Sunitha Chelliah Kumaravel, Regina Velnadar, Ravi Varma Vijayan, and Satyanarayana Parayitam. 2024. Quantifying risk in investment decision-making. *Journal of Risk and Financial Management*, 17(2):82.

LendingClub. 2007–2018. Lending club loan data. https://www.kaggle.com/datasets/wordsforthewise/lending-club. Accessed: 2025-09-20.

Stan Lipovetsky and Michael Conklin. 2001. Analysis of regression in game theory approach. *Applied stochastic models in business and industry*, 17(4):319–330.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024. Aligning with human judgement: The role of pairwise preference in large language model evaluators. *arXiv preprint arXiv:2403.16950*.

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

Aliakbar Nafar, Kristen Brent Venable, Zijun Cui, and Parisa Kordjamshidi. 2025. Extracting probabilistic knowledge from large language models for bayesian network parameterization. *arXiv preprint arXiv:2505.15918*.

OpenAI. 2024. Gpt-4.1 mini — openai api.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, and 1 others. 2023. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*.

J. Ross Quinlan. 1986. Induction of decision trees. *Machine learning*, 1(1):81–106.

Wullianallur Raghupathi and Viju Raghupathi. 2021. Contemporary business analytics: An overview. *Data*, 6(8):86.

Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane. 2019. Machine learning in medicine. *New England Journal of Medicine*, 380(14):1347–1358.

Kevin Ren, Santiago Cortes-Gomez, Carlos Miguel Patiño, Ananya Joshi, Ruiqi Lyu, Jingjing Tang, Alistair Turcan, Khurram Yamin, Steven Wu, and Bryan Wilder. 2025. Predicting language models'success at zero-shot probabilistic prediction.

Bahman Rostami-Tabar and Rob J Hyndman. 2025. Hierarchical time series forecasting in emergency medical services. *Journal of Service Research*, 28(2):278–295.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Lloyd S Shapley and 1 others. 1953. A value for n-person games.

Erik Štrumbelj and Igor Kononenko. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. 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*, pages 645–654.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Robert L Winkler, Yael Grushka-Cockayne, Kenneth C Lichtendahl Jr, and Victor Richmond R Jose. 2019. Probability forecasts and their combination: A research perspective. *Decision Analysis*, 16(4):239–260.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, and 1 others. 2024. Finben: A holistic financial benchmark for large language models. *Advances in Neural Information Processing Systems*, 37:95716–95743.

A Theory and Algorithm

A.1 Detailed Algorithm

Guided by the Shapley additivity property (Property 1), PRISM estimates each factor's marginal effect via paired contrasts (realized vs. baseline value of that factor), averages these effects over randomly sampled contexts, and then *reconstructs* the model output, finally mapping to a calibrated probability.

Setup. Let $\mathcal{I} = \{1, \ldots, m\}$ index factors, $x = (x_1, \ldots, x_m)$ be the instance, $b = (b_1, \ldots, b_m)$ be designated baselines , f be an evaluation oracle that returns either a *probability* in $[0, 1]$ or a *logit* in $\mathbf{R}$ when only a subset of factors is revealed, and Π_i be a distribution over background sets $S \subseteq \mathcal{I} \setminus \{i\}$. Given S, we write x_S for the partial specification that reveals $\{x_j : j \in S\}$.

Algorithm 1 PRISM (Probability Reconstruction via Shapley Measures)

Input: Instance $x \in \mathcal{X}^m$; oracle f; sampling distributions $\{\Pi_i\}_{i=1}^m$; budget K.
Output: Probability $\hat{p}(x)$ and factor attributions $\{\hat{\phi}_i(x)\}$.

$\phi_0 \leftarrow f(x_\emptyset)$
for $i \in \mathcal{I}$ **do**
  $\Delta \leftarrow 0$
  for k = 1 to K **do**
    Sample $S \sim \Pi_i$
    $\Delta \leftarrow \Delta + \big(f(x_{S\cup\{i\}}) - f(x_S)\big)$
  $\hat{\phi}_i(x) \leftarrow \Delta/K$
$\hat{z}(x) \leftarrow \phi_0 + \sum_i \hat{\phi}_i(x), \quad \hat{p}(x) \leftarrow \sigma(\hat{z}(x))$
return $\hat{p}(x)$ and $\{\hat{\phi}_i(x)\}$

Explanation for Alg. 1 . (1) Query the oracle on the empty specification to obtain the intercept $\phi_0 = f(x_\emptyset)$. (2–3) Start looping over factors $i \in \mathcal{I}$ and initialize the accumulator $\Delta \leftarrow 0$ for factor i. (4–6) For $k = 1, \ldots, K$, draw a background set $S \sim \Pi_i$ and accumulate the presence/absence contrast $f(x_{S\cup\{i\}})$–$f(x_S)$ into Δ. (7) Average the K contrasts to estimate the contribution of factor i: $\hat{\phi}_i(x) \leftarrow \Delta/K$. (8) Reconstruct the score by additivity, $\hat{z}(x) = \phi_0 + \sum_i \hat{\phi}_i(x)$, and map through the logistic link to get the probability $\hat{p}(x) = \sigma(\hat{z}(x))$. (9) Return the probability and the per-factor attributions, i.e., $\hat{p}(x)$ and $\{\hat{\phi}_i(x)\}$.

Explanation for Alg. 2 . (1) Obtain the intercept by querying $f(x_\emptyset)$ so $\phi_0 = f(x_\emptyset)$. (2–3) For each factor i, sample K background contexts $S_i^{(k)} \sim \Pi_i$ to form the evaluation batches. (4) In each sampled context, evaluate a realized/baseline pair in batch: $f(x_{S_i^{(k)}\cup\{i\}} \mid i := x_i)$ and $f(x_{S_i^{(k)}\cup\{i\}} \mid i := b_i)$. (5) Average the K realized–baseline differences to obtain $\hat{\phi}_i(x) = \frac{1}{K}\sum_{k=1}^{K}\big(f(x_{S_i^{(k)}\cup\{i\}} \mid i := x_i) - f(x_{S_i^{(k)}\cup\{i\}} \mid i := b_i)\big)$. (6) Reconstruct the score $\hat{z}(x) = \phi_0 + \sum_i \hat{\phi}_i(x)$ and apply the logistic link to produce $\hat{p}(x) = \sigma(\hat{z}(x))$. (7) Return $\hat{p}(x)$ together with the attributions $\{\hat{\phi}_i(x)\}$.

Algorithm 2 Tabular-PRISM (Batched realized vs. baseline contrasts)

Input: Instance $x \in \mathcal{X}^m$; oracle f; designated baselines b; sampling distributions $\{\Pi_i\}_{i=1}^m$; budget K.
Output: Probability $\hat{p}(x)$ and factor attributions $\{\hat{\phi}_i(x)\}$.

$\phi_0 \leftarrow f(x_\emptyset)$
for $i \in \mathcal{I}$ **do**
  For $k = 1, \ldots, K$, sample $S_i^{(k)} \sim \Pi_i$
  Batch-Query: evaluate all pairs
$$\Big\{f(x_{S_i^{(k)}\cup\{i\}} | i{:=}x_i),\ f(x_{S_i^{(k)}\cup\{i\}} | i{:=}b_i)\Big\}_{k=1}^{K}$$
  $\hat{\phi}_i(x) \leftarrow \frac{1}{K}\sum_{k=1}^{K}\Big(f(x_{S_i^{(k)}\cup\{i\}} | i{:=}x_i) - f(x_{S_i^{(k)}\cup\{i\}} | i{:=}b_i)\Big)$
$\hat{z}(x) \leftarrow \phi_0 + \sum_i \hat{\phi}_i(x), \quad \hat{p}(x) \leftarrow \sigma(\hat{z}(x))$
return $\hat{p}(x)$ and $\{\hat{\phi}_i(x)\}$

A.2 Theorem proof

Proposition 1. *Fix an instance* x *and a single reference sample* r. *Let* $\phi_i^{(r)}$ *be the Shapley value in Definition 2. Then, for models with deterministic scalar outputs, with* $\phi_0^{(r)} = v_r(\emptyset)$*, we still have:*

$$f(x) = v_r(\mathcal{I}) = \phi_0^{(r)} + \sum_{i=1}^{m} \phi_i^{(r)} \tag{7}$$

Proof. We use the permutation form of the Shapley value, which is equivalent to the subset form. Let π be a permutation of $\mathcal{I}$, and let $\Pi(\mathcal{I})$ be the set of all m! permutations. For a permutation π and an index i, let $\mathrm{Pre}_i(\pi)$ denote the set of features that appear before i in π. The permutation form of the Shapley value of i for the game $v_r(S) = f\big([x_S,\ r_{\bar{S}}]\big)$

can be written as

$$\phi_i^{(r)} = \frac{1}{m!} \sum_{\pi \in \Pi(\mathcal{I})} \Big(v_r\big(\mathrm{Pre}_i(\pi) \cup \{i\}\big) - v_r\big(\mathrm{Pre}_i(\pi)\big) \Big). \tag{8}$$

Taking equation (8) to the $\sum_{i=1}^{m} \phi_i^{(r)}$ part yields

$$\sum_{i=1}^{m} \phi_i^{(r)} = \frac{1}{m!} \sum_{\pi \in \Pi(\mathcal{I})} \sum_{i=1}^{m} \Big(v_r\big(\mathrm{Pre}_i(\pi) \cup \{i\}\big) - v_r\big(\mathrm{Pre}_i(\pi)\big) \Big). \tag{9}$$

For a fixed permutation $\pi = (\pi_1, \pi_2, \ldots, \pi_m) \in \Pi(\mathcal{I})$, we have $\mathrm{Pre}_i(\pi) \cup \{i\} = \mathrm{Pre}_{i+1}(\pi)$ for $1 \leq i \leq m-1$. Therefore

$$\sum_{i=1}^{m} \Big(v_r\big(\mathrm{Pre}_i(\pi) \cup \{i\}\big) - v_r\big(\mathrm{Pre}_i(\pi)\big) \Big) = v_r(\mathcal{I}) - v_r(\emptyset). \tag{10}$$

Substituting equation (10) into (9) gives

$$\sum_{i=1}^{m} \phi_i^{(r)} = \frac{1}{m!} \sum_{\pi \in \Pi(\mathcal{I})} \big(v_r(\mathcal{I}) - v_r(\emptyset) \big) = v_r(\mathcal{I}) - v_r(\emptyset). \tag{11}$$

Finally, given $\phi_0^{(r)} := v_r(\emptyset)$ and $v_r(\mathcal{I}) = f\big([x_{\mathcal{I}}, r_{\emptyset}]\big) = f(x)$, we have

$$f(x) = v_r(\mathcal{I}) = \phi_0^{(r)} + \sum_{i=1}^{m} \phi_i^{(r)}. \tag{12}$$

Proof complete. □

B Additional Details for Experiments

B.1 Datasets

Stroke: The stroke prediction dataset contains health, demographic, and lifestyle information for 5,110 patients, with the goal of predicting stroke occurrence. Each record includes variables such as age, hypertension, heart disease, marital status, work type, body mass index (BMI) and so on.

Heart Disease: The heart disease dataset integrates multiple clinical heart disease datasets and contains 918 patient records with 11 features related to demographics, clinical measurements, and lifestyle factors. The target variable indicates the presence or absence of heart disease.

Adult Census: The adult census contains 48,842 records from the 1994 U.S. Census, with 14 demographic and employment-related features such as age, education, occupation, work hours, and marital status. The target variable indicates whether an individual's annual income exceeds $50,000.

Lending: The lending dataset contains peer-to-peer loans issued from 2007–2018, with borrower and loan features such as income, debt-to-income ratio, FICO score, interest rate, loan amount, and purpose. The target variable indicates whether the loan will default.

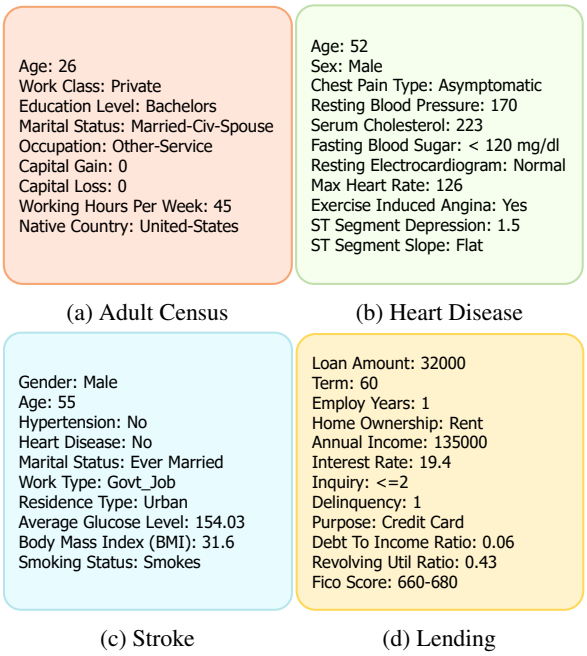
Before applying above datasets for evaluation, we preprocess the datasets as follows:

(1) In order to make LLMs better understand the datasets, we rename the names of columns of datasets. For example, for **stroke** dataset, we rename "avg_glucose_level" as "average glucose level" and rename "bmi" as "Body Mass Index".

(2) For some columns, the values may be some abbreviation or with unclear meanings. For example, the values of attribute "gender" in dataset **stroke** are {0, 1}, so we convert "0" into "Female" and "1" into "Male". Besides, for attribute "Chest Pain Type" in dataset **heart disease**, the values are abbreviations such as "ATA", "NAP". We also convert them into their full names, e.g., "ATA" → "Atypical Angina" and "NAP" → "Non-Anginal Pain".

(3) We dropped some attributes of the datasets so that them LLMs can evaluation each data point with more efficiency.

The final data points examples and columns of each dataset are summarized in Figure (6).



(a) Adult Census (b) Heart Disease

(c) Stroke (d) Lending

Figure 6: Data examples of the datasets.

B.2 Baseline Methods

1/5/10shot_level: We perform multiple shots or multiple trials on directly asking LLM for prob-

ability as in prompt Figure (9). The LLM is instructed to select a linguistic probability description to represent the probability. Before evaluation, we map the descriptions into numerical values as follows: “very unlikely": 0.05, “unlikely": 0.2, “somewhat unlikely": 0.35, “neutral": 0.5, “somewhat likely": 0.65, “likely": 0.8, “very likely": 0.95. For 1shot_level, we only perform one trial; for 5shot_level and 10shot_level we perform 5 and 10 trials, and select the most common value as output.

1/5/10shot_score: We perform multiple shots similar to 1/5/10shot_level. The difference is that we instruct the LLM to output the numerical probability directly, as shown in the prompt Figure (10). We take the average probability as output.

ICL-5+5, ICL-10+10: We perform in-context learning on the 4 datasets with available training data. There are 5 positive 5 negative data in ICL-5+5 and 10 negative 10 positive data in ICL-10+10. The training data are excluding the evaluation data, with large mutual differences from the arbitrary candidate set. The order of the training data inside the prompt is shuffled randomly to minimize confusion. Detailed prompt example is shown as follows 11.

Contrast: We perform two queries in opposite directions. The first query apply the same prompt as in 1shot_level Figure (9). The second query is trying to ask the question in the opposite way, such as "How likely is this patient to NOT have a stroke?" instead of "How likely is this patient to have a stroke?". Two queries are normalized so that they sum up to 1, and the normalized positive answer will be the output.

BIRD: We follow BIRD’s method with slight modification. First, since BIRD could not handle numerical datatype features, those features will be turned into bins before treated as input. In order to better utilize prior knowledge of LLM, binning will base on prior evidence that extract the most characteristic of the stage, therefore the interval of each bin may not have the same length. For example, regarding to the numerical feature “Resting Blood Pressure" in the Heart Disease dataset, the bins are: Normal ‘80-120’, Pre-hypertension ‘120-130’, Hypertension Stage 1 ‘130-140’, Hypertension Stage 2 ‘140-180’, Hypertensive crisis ‘180-200’. Second, instead of probability mapping in BIRD {“f_j supports outcome i": 75%, “f_j is neutral": 50%, “f_j supports opposite outcome ¬i": 25%}, we are using a denser mapping which is the same as introduced in 1shot_level.

In general, the baseline BIRD is performed in 4 steps: (1) Initializing the prediction probability given each independent feature, by querying LLM using similar prompt as in 1shot_level Figure (9), only replace “Person information: ..." by “Given that [*factor*] = [*value*]". (2) Generating stochastic training data, by randomly choice a bin for each feature, the prompt is the same as in 1shot_level Figure (9). (3) Training the BIRD constrained optimization method. (4) Inferring to the evaluation dataset.

B.3 Reference instances settings

Stroke	
Gender	Male
Age	40.0
Hypertension	No
Heart disease	No
Marital status	Never Married
Residence type	Rural
Average glucose level	90.0
Body Mass Index (BMI)	24.0
Work type	Private
Smoking status	never smoked
Adult	
Age	40
Workclass	Private
Education level	Some-college
Marital status	Married-civ-spouse
Occupation	Sales
Capital gain	0
Capital loss	0
Working hours per week	40
Native country	United-States
Loan	
Loan amount	20000
Term	36
Employ years	3
Home ownership	OWN
Annual income	60000
Interest rate	14.0
Purpose	car
Debt-to-income ratio	0.35
Revolving util ratio	0.30
FICO score	680–710
Inquiry	≤ 2
Delinquency	0
Heart Disease	
Age	53
Resting Blood Pressure	133
Serum Cholesterol	212
Max Heart Rate	137
ST Segment Depression	0.8
Sex	Male
Chest Pain Type	Asymptomatic
Fasting Blood Sugar	< 120 mg/dl
Resting Electrocardiogram	Normal
Exercise Induced Angina	No
ST Segment Slope	Flat

Table 5: Reference instance of each dataset.

We instantiate a per-dataset *reference instance* x_{ref} that serves as the baseline context for PRISM's contrastive evaluations. Reference values are chosen to be representative, i.e., close to the empirical mean for continuous variables and a prevalent category for discrete variables (the mode for multi-class features and the negative category for binary features). The concrete instances used in our experiments are listed in Table 5.

For each dataset's reference instance,, we use prompt 10, query the model five times, average the predicted probabilities, and convert the result to a base logit via $\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$. The resulting base probabilities and logits are reported in Table 6.

Dataset	p (mean over 5 runs)	logit
Stroke	0.001	-6.9068
Adult	0.354	-0.6015
Heart Disease	0.410	-0.3640
Loan	0.182	-1.5029

Table 6: Base probabilities p (from prompt 10, averaged over five queries) and corresponding base logits per dataset.

C Additional Results

C.1 Examination of Interpretations

We examine factor-level attributions on the tabular datasets in Table 7, Table 8, and Table 9.

Adult (Table 7). The attributions align with well-known socio-economic regularities for predicting income. Education exerts the largest and most consistent influence: *Masters* yields a strong positive contribution (Case 4: +1.31), while *HS-grad* is negative (Case 3: –0.45). Occupational roles are similarly informative: *Exec-managerial* is positive (Cases 1/4: +0.49/ + 0.99), whereas *Farming-fishing* is negative (Case 3: –0.84). *Capital gain* is highly predictive when present (Case 1: +1.04), and *age* contributes moderately with the expected direction (older age increasing odds in Cases 1/3). Marital status exhibits negative impacts for *Divorced* and *Never-married* (Cases 1/2), consistent with prior findings that marriage correlates with higher income. Some variables (e.g., *workclass*, *native country*) show near-zero effects in these cases, indicating either proximity to the anchor or low marginal power after conditioning on stronger factors.

Heart (Table 8). For heart disease risk, the factor impacts align with real-world clinical patterns *Exercise-induced angina* is strongly positive (Cases 1/2: +0.89/ + 0.87), and elevated *resting blood pressure* and *serum cholesterol* contribute positively (Case 1: +0.43/ + 0.45; Case 2: +0.38/ + 0.29). Conversely, higher *max heart rate* and *upsloping ST slope* decrease risk (Case 3: –0.45 and –0.48), aligning with cardiology practice that better exercise capacity and non-flat slopes are protective. *Sex=Female* is negative in Case 3 (–0.53), capturing lower risk in females. Note that the same feature can change sign across cases (e.g., *ST depression*: –0.59 in Case 2 vs. near-zero/positive elsewhere), reflecting interactions and the conditional nature of $\phi_i(x)$ under different covariate settings.

Loan (Table 9). For default risk, the patterns are intuitive. High *interest rate* increases risk (Cases 3/4: +0.67/ + 0.71), while lower rates reduce it (Case 1: –0.50). Past *delinquency* is the single most influential positive factor when present (Case 3: +1.15). *Home ownership=RENT* and lower *annual income* tend to raise risk (Case 3: +0.44/ + 0.48), while higher *FICO* mitigates risk and lower *FICO* elevates it (Case 1: –0.41 vs. Case 3: +0.45). Debt burden is captured by *DTI* and *revolving utilization*: lower values reduce risk (Case 1: –0.08 and –0.51), whereas moderate-to-high levels are less favorable across other cases.

Across these tabular datasets, predicted probabilities exhibit a label-consistent ordering with clear separation—for example, the positives cluster at higher values while the negatives fall into intermediate and low ranges—yielding an easily interpretable ranking. Per-instance factor impacts make the drivers of each estimate explicit and support auditability: dominant contributors can be inspected, implausible probabilities traced to specific factors, and questionable cases flagged for review.

C.2 Calibration Analysis

Setup. We assess calibration by plotting the weighted reliability curve. Given predictions $\{\hat{p}_i\}$ and labels $\{y_i \in \{0, 1\}\}$, we bin by *equal-count* quantiles of $\hat{p}$ and compute in-bin weighted means

$$\begin{aligned} \hat{p}_m &= \frac{1}{W_m} \sum_{i \in B_m} w_i \hat{p}_i, \\ \hat{y}_m &= \frac{1}{W_m} \sum_{i \in B_m} w_i y_i, \\ W_m &= \sum_{i \in B_m} w_i. \end{aligned} \tag{13}$$

Weights correct the gap between the *evaluation*

Factor	Case 1		Case 2		Case 3		Case 4	
	Value	Shapley	Value	Shapley	Value	Shapley	Value	Shapley
Age	51	0.17	25	-0.41	47	0.38	38	0.00
Workclass	Private	0.00	Private	0.00	Self-emp-not-inc	-0.44	Private	0.00
Education level	Bachelors	0.62	Bachelors	0.49	HS-grad	-0.45	Masters	1.31
Marital status	Divorced	-0.47	Never-married	-0.41	Married-civ-spouse	0.00	Married-civ-spouse	0.00
Occupation	Exec-managerial	0.49	Sales	0.00	Farming-fishing	-0.84	Exec-managerial	0.99
Capital gain	10520	1.04	0	0.00	0	0.00	0	0.00
Capital loss	0	0.00	1876	0.00	0	0.00	0	0.00
Working hours/wk	40	0.00	40	0.00	60	0.54	60	0.53
Native country	US	0.00	US	0.00	US	0.00	US	0.00
Sum Shapley	1.86		-0.33		-0.81		2.83	
Sum logit	1.26		-0.93		-1.41		2.22	
Pred prob	0.778		0.283		0.197		0.902	
True label	Yes		No		No		Yes	

Table 7: Shapley values for four instances in Adult dataset. Reference instance: *age=40; workclass=Private; education=Some-college; marital status=Married-civ-spouse; occupation=Sales; capital gain=0; capital loss=0; working hours=40; native country=US*. A single base logit is shared across cases, $\phi_0 = \sigma^{-1}(0.354) = -0.6015$.

Factor	Case 1		Case 2		Case 3		Case 4	
	Value	Shapley	Value	Shapley	Value	Shapley	Value	Shapley
Age	52	0.00	58	0.00	34	-0.73	48	-0.12
Sex	Male	0.00	Male	0.00	Female	-0.53	Male	0.00
Chest Pain Type	Asymptomatic	0.00	Non-Anginal Pain	0.70	Atypical Angina	0.42	Asymptomatic	0.00
Resting Blood Pressure	170	0.43	150	0.38	118	-0.49	132	0.00
Serum Cholesterol	223	0.45	219	0.29	210	0.00	272	0.37
Fasting Blood Sugar	< 120 mg/dl	0.00	< 120 mg/dl	0.00	< 120 mg/dl	0.00	< 120 mg/dl	0.00
Resting ECG	Normal	0.00	ST-T abn.	0.33	Normal	0.00	ST-T abn.	0.45
Max Heart Rate	126	0.19	118	0.00	192	-0.45	139	0.00
Exercise Induced Angina	Yes	0.89	Yes	0.87	No	0.00	No	0.00
ST Segment Depression	1.5	0.44	0.0	-0.59	0.7	0.00	0.2	-0.44
ST Segment Slope	Flat	0.00	Flat	0.00	Upsloping	-0.48	Upsloping	-0.44
Sum Shapley	2.40		1.99		-2.25		-0.63	
Sum logit	2.04		1.62		-2.61		-0.54	
Pred prob	0.885		0.835		0.068		0.367	
True label	Yes		Yes		No		No	

Table 8: Shapley values for four instances in Heart dataset. Reference instance: *Age=53; Resting BP=133; Serum Chol.=212; Max HR=137; ST Depression=0.8; Sex=Male; Chest Pain=Asymptomatic; Fasting Blood Sugar=*< 120 *mg/dl; Resting ECG=Normal; Exercise Angina=No; ST Slope=Flat*. A single base logit is shared across cases, $\phi_0 = \sigma^{-1}(0.41) = -0.363$.

split and the *deployment/population* class mix:

$$w_i = \begin{cases} \dfrac{\pi}{\hat{\pi}}, & y_i = 1, \\ \dfrac{1-\pi}{1-\hat{\pi}}, & y_i = 0, \end{cases} \tag{14}$$

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{W_m}{\sum_j W_j} |\hat{p}_m - \hat{y}_m|.$$

Here, $\hat{\pi}$ is the positive rate *in our evaluation split*, which is class-balanced ($\hat{\pi} = 0.5$). π is the *population* positive rate we aim to evaluate against. Since the true deployment prevalence is unknown, we use the *original dataset prevalence before balancing* as a proxy for π: *Stroke* 4.87%, *Adult Census* 24.88%, *Heart Disease* 55.28%. (If a practitioner knows their deployment prevalence, they can plug it in for π.) We visualize calibration by plotting $(\hat{p}_m, \hat{y}_m)$ against the identity line y = x.

Why weighting matters. Our test splits are 1:1 balanced, but real-world prevalence is typically skewed. Without weighting ($w_i \equiv 1$), bin positive fractions reflect the artificial 50% mix rather than the population, biasing the curve upward on imbalanced tasks (e.g., *Stroke*). Importance weights re-create the population mix *within each bin*, so the reliability curve answers the practical question: "given this score in deployment, what fraction will be positive?" Therefore, Reweighting restores the population class mix in each bin, yielding reliability curves that reflect real–world deployment rather than the artificial test mix.

Results. Across all four datasets in Fig. 7, *PRISM* maintains strong calibration, remaining close to the y=x line on *Adult* and *Heart*, and staying closest to the diagonal on *Stroke* and *Loan* despite small deviations, while *consistently yielding a monotonically increasing reliability curve*. This monotonicity ensures that higher predicted proba-

Factor	Case 1 (ID=1)		Case 2 (ID=2)		Case 3 (ID=29)		Case 4 (ID=37)	
	Value	Shapley	Value	Shapley	Value	Shapley	Value	Shapley
Loan amount	30000	0.45	1500	0.00	10000	-0.38	30000	0.48
Term	60	0.45	36	0.00	36	0.00	60	0.51
Employ years	4	-0.43	< 1	0.45	6	-0.43	10+	-0.55
Home ownership	MORTGAGE	0.00	MORTGAGE	0.00	RENT	0.44	MORTGAGE	0.42
Annual income	65000	-0.27	55000	0.00	48000	0.48	42000	0.46
Interest rate	12.0	-0.50	10.4	-0.43	20.0	0.67	19.4	0.71
Inquiry	≤ 2	0.00	≤ 2	0.00	≤ 2	0.00	≤ 2	0.00
Delinquency	0	0.00	0	0.00	1	1.15	0	0.00
Purpose	Debt cons.	0.43	Home improv.	0.00	Debt cons.	0.44	Debt cons.	0.48
Debt-to-income ratio	0.25	-0.08	0.13	-0.55	0.29	0.00	0.17	0.00
Revolving util ratio	0.21	-0.51	0.29	0.00	0.08	-0.47	0.44	0.00
FICO score	710–740	-0.41	710–740	-0.77	660–680	0.45	660–680	0.42
Sum Shapley	-0.87		-1.30		2.35		2.93	
Sum logit	-2.37		-2.81		0.85		1.43	
Pred prob	0.086		0.057		0.701		0.806	
True label	No		No		Yes		Yes	

Table 9: Shapley values for four instances in Loan dataset. Reference instance: *Loan amount=20000; Term=36; Employ years=3; Home ownership=OWN; Annual income=60000; Interest rate=14.0; Purpose=car; Debt-to-income ratio=0.35; Revolving util ratio=0.30; FICO score=680–710; Inquiry=≤ 2; Delinquency=0*. A single base logit is shared across cases, $\phi_0 = \sigma^{-1}(0.182) = -1.504$.

bilities always correspond to higher empirical event rates (no local reversals), preventing rank inconsistencies and threshold instability that appear in the baselines when the fraction of positives decreases as the mean predicted probability increases, and it further shows that our method is better calibrated than the baselines.

C.3 Can PRISM generalize to multi-class prediction?

We provide a concise strategy to extend PRISM to multi-class prediction tasks. Consider a classification problem with M possible classes. In PRISM, we aim to quantify the marginal contribution of each factor i to each class $m \in \{1, \ldots, M\}$. Given a background set S and a target factor i (see Definition 1), we query the LLM to estimate the probability that the input belongs to each class m for both x_S and $x_{S\cup\{i\}}$. We denote these estimates as $p^m(x_S)$ and $p^m(x_{S\cup\{i\}})$, respectively. Following the same procedure as in Section 3, we compute the logit difference:

$$f^m(x_{S\cup\{i\}}) - f^m(x_S),$$

and aggregate these differences across background set samples to obtain the Shapley value ϕ_i^m for each class m. This formulation naturally generalizes the binary case by computing class-wise factor contributions without modifying the underlying sampling strategy.

For the UCI-Wine dataset, we use all 178 samples and consider a three-class prediction task corresponding to different wine categories. PRISM is compared with direct prompting baselines, where the LLM is asked to select the most likely class either once (1-shot) or by majority voting over ten independent queries (10-shot). For the Weather Type Classification dataset, we consider a four-class prediction task with labels *rainy*, *sunny*, *cloudy*, and *snowy*. We randomly sample a total of 100 instances from the dataset, with 25 samples per class to ensure a balanced evaluation. The same multi-class PRISM procedure is applied to estimate class-wise factor contributions and generate final predictions.

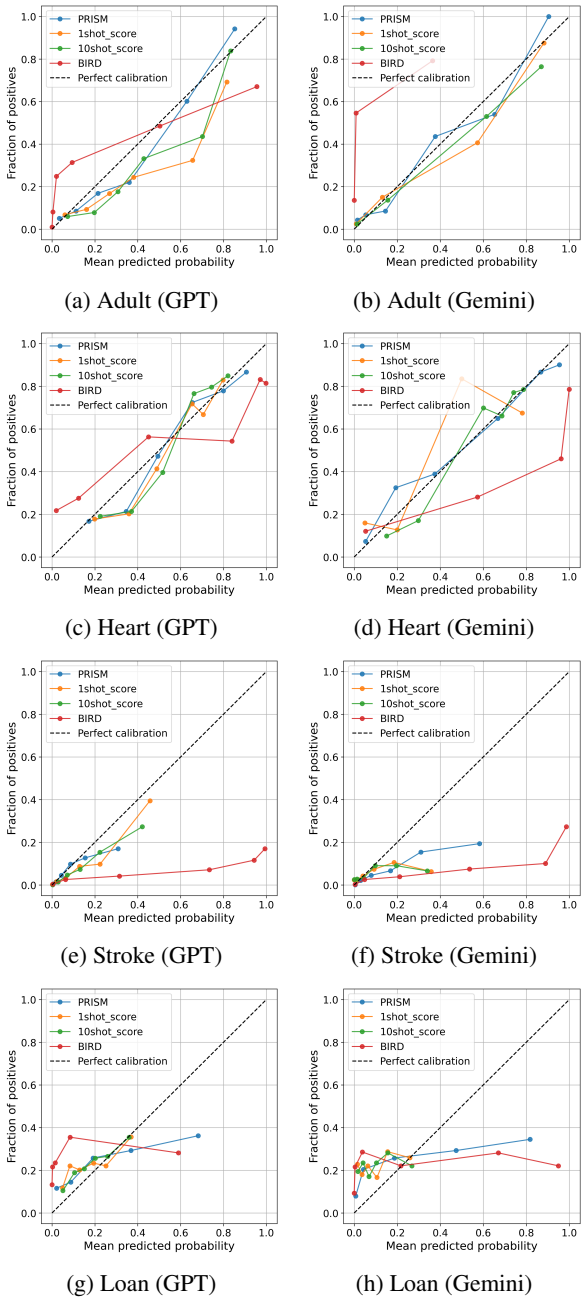
Method	Wine	Weather
1shot	0.437	0.720
10shot	0.453	0.760
PRISM (Ours)	**0.613**	**0.810**

Table 10: Accuracy on the UCI-Wine and Weather datasets.

C.4 Comparison with Supervised Logistic Regression

We include supervised baselines based on logistic regression trained with different amounts of labeled data for all four tabular datasets. For each dataset, we construct balanced training sets with varying sizes (20, 50, 100, 150, 200, 250 and 300 labeled samples), and report AUROC, AUPRC, and F1 scores on the same evaluation split used for PRISM.

Figure 8 summarizes the results. Across all four datasets, PRISM achieves performance comparable



(a) Adult (GPT) (b) Adult (Gemini)

(c) Heart (GPT) (d) Heart (Gemini)

(e) Stroke (GPT) (f) Stroke (Gemini)

(g) Loan (GPT) (h) Loan (Gemini)

Figure 7: Calibration (reliability) curves comparing PRISM, 1shot_score, 10shot_score, and BIRD on four datasets under GPT-4.1-mini and Gemini-2.5-Pro.

to logistic regression models trained with approximately 50–100 labeled samples. This observation suggests that PRISM persists a substantial amount of predictive signal without requiring explicit supervised training, effectively matching the performance of standard classifiers in low- to moderate-data regimes.

As expected, logistic regression continues to improve as more labeled data become available. However, in settings where labeled data are scarce or expensive to obtain, PRISM provides a competitive alternative by leveraging structured factor-wise reasoning through LLM queries rather than supervised optimization.

C.5 Sensitivity to the Number of Background Samplings K

We analyze the sensitivity of PRISM to the number of background set samplings K on the Adult Census and Lending datasets using GPT-4.1-mini. Table 11 reports the performance under different values of K, ranging from 1 to 10.

On the Adult Census dataset, PRISM achieves stable performance across a wide range of K, indicating low sensitivity to the sampling size. This behavior is consistent with the relatively weak feature interactions in this dataset, where individual factors tend to influence the prediction independently. In contrast, on the Lending dataset, performance degrades noticeably when K is small and improves steadily as K increases. This can be attributed to stronger feature interactions in Lending. For example, the effect of a high loan amount on default risk depends critically on the borrower's annual income level. With insufficient background samplings, PRISM may fail to capture such interactions accurately, leading to degraded prediction and attribution quality. Based on these observations, we use K = 10 as a default setting across all tabular datasets in our experiments to ensure robust performance and consistent evaluation.

D Prompts

We provide the details of all prompts we used.

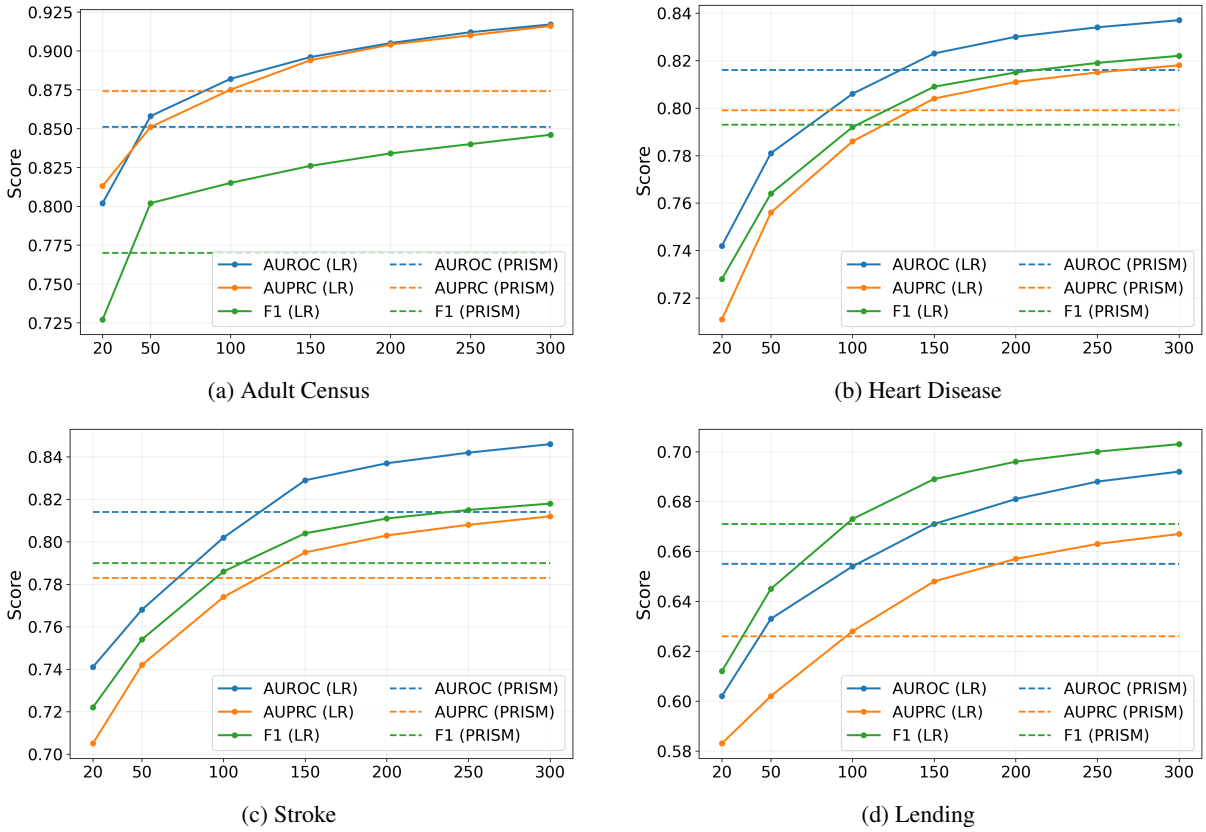


Figure 8: Performance comparison with logistic regression when training data is available. Dashed lines represent the performance of PRISM under the zero-shot setting.

K	Adult Census			Lending		
	AUROC	AUPRC	F1	AUROC	AUPRC	F1
1	0.844	0.870	0.759	0.591	0.552	0.579
2	0.849	0.874	0.763	0.603	0.567	0.610
3	0.850	0.873	0.767	0.618	0.583	0.638
4	0.849	0.873	0.766	0.631	0.597	0.654
5	0.850	0.875	0.762	0.643	0.608	0.662
6	0.849	0.874	0.770	0.650	0.615	0.666
7	0.848	0.873	0.770	0.653	0.621	0.669
8	0.849	0.874	0.773	0.655	0.624	0.670
9	0.850	0.874	0.770	0.656	0.626	0.670
10	0.850	0.874	0.770	0.655	0.626	0.671

Table 11: Sensitivity analysis of PRISM with respect to the number of background set samplings K on Adult Census and Lending datasets using GPT-4.1-mini.

```
You are an income prediction expert. Estimate the probability that the following person has an annual income
greater than $50,000.
This person lived in 1994; please base your judgment on the U.S. economic and social context of that year.

Person information:
[personal information in json]

How likely is this person to have income >$50,000?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(a) 1/5/10shot_level on Adult Census

```
You are a medical risk assessment expert. Estimate the probability that the following person will have heart
disease.

Person information:
[personal information in json]

How likely is this patient to have heart disease?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(b) 1/5/10shot_level on Heart Disease dataset

```
You are a medical risk assessment expert. Estimate the probability that the following person will have a stroke.

Person information:
[personal information in json]

How likely is this patient to have a stroke?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(c) 1/5/10shot_level on Stroke dataset

```
You are a loan-risk analyst. Estimate the probability that the following applicant will default on their loan.

Person information:
[personal information in json]

How likely is this applicant to be a defaulter?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(d) 1/5/10shot_level on Lending dataset

Figure 9: Prompts for baseline method 1shot_level, 5shot_level, 10shot_level. Where [personal information in json] is replaced by the actual data in json format.

```
You are an income prediction expert. Estimate the probability that the following person has an annual income
greater than $50,000.
This person lived in 1994; please base your judgment on the U.S. economic and social context of that year.

Person information:
[personal information in json]

How likely is this person to have income >$50,000?
First, provide a short explanation of your reasoning.
Then provide the final result strictly in the format:
##Final Result##: <a single number between 0 and 1>
```

(a) 1/5/10shot_score on Adult Census

```
You are a medical risk assessment expert. Estimate the probability that the following person will have heart
disease.

Person information:
[personal information in json]

How likely is this patient to have heart disease?
First, provide a short explanation of your reasoning.
Then provide the final result strictly in the format:
##Final Result##: <a single number between 0 and 1>
```

(b) 1/5/10shot_score on Heart Disease dataset

```
You are a medical risk assessment expert. Estimate the probability that the following person will have a stroke.

Person information:
[personal information in json]

How likely is this patient to have a stroke?
First, provide a short explanation of your reasoning.
Then provide the final result strictly in the format:
##Final Result##: <a single number between 0 and 1>
```

(c) 1/5/10shot_score on Stroke dataset

```
You are a loan-risk analyst. Estimate the probability that the following applicant will default on their loan.

Person information:
[personal information in json]

How likely is this applicant to be a defaulter?
First, provide a short explanation of your reasoning.
Then provide the final result strictly in the format:
##Final Result##: <a single number between 0 and 1>
```

(d) 1/5/10shot_score on Lending dataset

Figure 10: Prompts for baseline method 1shot_score, 5shot_score, 10shot_score. Where [personal information in json] is replaced by the actual data in json format, and the last line of each prompt is the format guideline.

```
You are an income prediction expert. Estimate whether a person in 1994 would have an annual income greater than
$50K, based on U.S. economic and social context of that year. Please use your own knowledge and the examples below
as references.
Examples:
- Features: [personal information in json]
  Label: [yes/no]
- Features: [personal information in json]
  Label: [yes/no]
…

Now predict the following case:
Features: [personal information in json]
Question: How likely is this person to have income $50,000?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(a) ICL on Adult Census

```
You are a medical risk assessment expert.
Estimate whether the following person will have a heart disease. Please use your own knowledge and the examples
below as references.
Examples:
- Features: [personal information in json]
  Label: [yes/no]
- Features: [personal information in json]
  Label: [yes/no]
…

Now predict the following case:
Features: [personal information in json]
Question: How likely is this patient to have a heart disease?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(b) ICL on Heart Disease dataset

```
You are a medical risk assessment expert. Estimate whether the following person will have a stroke. Please use your
own knowledge and the examples below as references.
Examples:
- Features: [personal information in json]
  Label: [yes/no]
- Features: [personal information in json]
  Label: [yes/no]
…

Now predict the following case:
Features: [personal information in json]
Question: How likely is this patient to have a stroke?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(c) ICL on Stroke dataset

```
You are a loan-risk analyst. Estimate the probability that the following applicant will default on their loan.
Please use your own knowledge and the examples below as references.
Examples:
- Features: [personal information in json]
  Label: [yes/no]
- Features: [personal information in json]
  Label: [yes/no]
…

Now predict the following case:
Features: [personal information in json]
Question: How likely is this applicant to be a defaulter?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(d) ICL on Lending dataset

Figure 11: Prompts for baseline method ICL-5+5, ICL-10+10. Where [personal information in json] is replaced by the actual data in json format. For ICL-5+5, we have 5 positive and 5 negative training data randomly ordered as Features and Values in the prompt, and similar for ICL-10+10 except we have 10 positive and 10 negative training data.

```
You are an income prediction expert. Estimate the probability that the following person has an annual incomeat most
$50,000.
This person lived in 1994; please base your judgment on the U.S. economic and social context of that year.

Person information:
[personal information in json]

How likely is this person to have income <=$50,000?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, verylikely.
```

(a) Contrast on Adult Census

```
You are a medical risk assessment expert. Estimate the probability that the following person will NOThave heart
disease.

Person information:
[personal information in json]

How likely is this patient to NOT have heart disease?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, verylikely.
```

(b) Contrast on Heart Disease dataset

```
You are a medical risk assessment expert. Estimate the probability that the following person will NOT have a stroke.

Person information:
[personal information in json]

How likely is this patient to NOT have a stroke?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, very likely.
```

(c) Contrast on Stroke dataset

```
You are a loan-risk analyst. Estimate the probability that the following applicant will NOT default on their loan.

Person information:
[personal information in json]

How likely is this applicant to be a non-defaulter?
First, provide a short explanation of your reasoning.
Then answer with one of: very unlikely, unlikely, somewhat unlikely, neutral, somewhat likely, likely, verylikely.
```

(d) Contrast on Lending dataset

Figure 12: Negative prompts for baseline method Contrast. Where [personal information in json] is replaced by the actual data in json format. The positive prompt is the same as in 1shot_label prompt 9.

```
[A Markdown table with 10 pairs of records]

Your task is to evaluate the likelihood that each person has annual wage over 50K dollars in the United
States in 1994.

For each person:
- Conduct a brief analysis.
- Give a risk score from 1 to 10 (1 = very unlikely, 10 = very likely).

Provide the final result in the format ##Final Result##: @ID1: 1-10; @ID2: 1-10...
```

(a) Tabular-PRISM on Adult Census

```
[A Markdown table with 10 pairs of records]

Your task is to evaluate the likelihood that each person has heart disease.

For each person:
- Conduct a brief analysis.
- Give a risk score from 1 to 10 (1 = very unlikely, 10 = very likely).

Provide the final result in the format ##Final Result##: @ID1: 1-10; @ID2: 1-10...
```

(b) Tabular-PRISM on Heart Disease dataset

```
[A Markdown table with 10 pairs of records]

Your task is to evaluate the likelihood that each person has stroke.

For each person:
- Conduct a brief analysis.
- Give a risk score from 1 to 10 (1 = very unlikely, 10 = very likely).

Provide the final result in the format ##Final Result##: @ID1: 1-10; @ID2: 1-10...v
```

(c) Tabular-PRISM on Stroke dataset

```
[A Markdown table with 10 pairs of records]

Your task is to evaluate the likelihood that each loan application will default.

For each application:
- Conduct a brief analysis.
- Give a risk score from 1 to 10 (1 = very unlikely, 10 = very likely).

Provide the final result in the format ##Final Result##: @ID1: 1-10; @ID2: 1-10...
```

(d) Tabular-PRISM on Lending dataset

Figure 13: Prompt templates for Tabular–PRISM across four datasets. Each panel shows the standardized instruction used to score instances from a Markdown table with 10 pairs of records; each pair comprises a target case and its matched baseline case.