P-Shapley: Shapley Values on Probabilistic Classifiers

Haocheng Xia
Zhejiang University
xiahc@zju.edu.cn

Xiang Li
Zhejiang University
lixiangzx@zju.edu.cn

Junyuan Pang
Zhejiang University
junyuanpang@zju.edu.cn

Jinfei Liu*
Zhejiang University
ZJU-Hangzhou Global Scientific and Technological Innovation Center
jinfeiliu@zju.edu.cn

Kui Ren
Zhejiang University
kuiren@zju.edu.cn

Li Xiong
Emory University
lxiong@emory.edu

ABSTRACT

The Shapley value provides a unique approach to equitably gauge each player's contribution within a coalition and has extensive applications with various utility functions. In data valuation for machine learning, particularly for classification tasks, using classification accuracy as the utility function has become a de facto standard. However, accuracy can be an imprecise metric, potentially missing finer details crucial for valuation. In this paper, we propose the probability-based Shapley (P-Shapley) value, which leverages predicted probabilities to heighten utility differentiation. Several convex calibration functions are further incorporated for probability calibration. We prove that the P-Shapley value outperforms Shapley values based on accuracy or other coarse metrics in approximation stability and the discrimination of marginal utility change can be further improved by convex calibration functions. Extensive experiments on four real-world datasets demonstrate the effectiveness of our approaches.

PVLDB Reference Format:
Haocheng Xia, Xiang Li, Junyuan Pang, Jinfei Liu, Kui Ren, and Li Xiong. P-Shapley: Shapley Values on Probabilistic Classifiers. PVLDB, 17(7): 1737 - 1750, 2024.
doi:10.14778/3654621.3654638

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/ZJU-DIVER/P-Shapley.

1 INTRODUCTION

The renowned Shapley value [36] offers a unique approach of distributing gains fairly to contributors based on their contribution towards a collective utility in a cooperative game. This approach satisfies all four desirable properties of fairness, including efficiency, symmetry, zero element, and additivity. The Shapley value is general owing to the flexible choices of utility functions [10, 17, 27]. Consequently, it has been extensively employed in various applications, such as explainable artificial intelligence [15, 27], data/feature selection [5, 17], cloud computing pricing [38], and data product pricing in data markets [3, 25, 26, 30]. When the utility function is model performance (e.g., classification accuracy), the Shapley value becomes one of the most prevalent data valuation methods for machine learning (ML) [29], showing superior capability on multiple downstream ML tasks, including identifying mislabeled examples [17], detecting outliers [20], and model debugging over end-to-end ML pipelines [34].

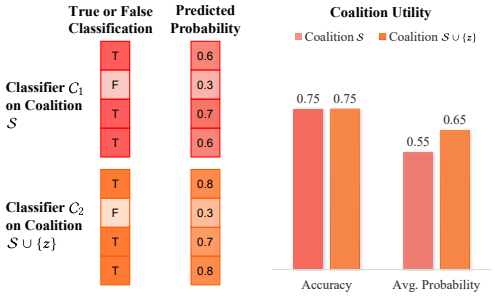

Figure 1: A motivating example on the drawback of using accuracy as the utility function for the Shapley value.

Given a training set comprised of n data points for a classification problem, the ML task (i.e., training a classifier) can be regarded as a cooperative game. The Shapley value views each data point as a player and the utility function can be defined as the classification accuracy of the classifier trained on the collective data. The Shapley value of a data point z is the expectation of the marginal utility contribution that z makes for a coalition of data points $\mathcal{S}$ ($z \notin \mathcal{S}$), that is, $\mathcal{U}(\mathcal{S} \cup \{z\}) - \mathcal{U}(\mathcal{S})$, where $\mathcal{U}(\mathcal{S})$ is the classification accuracy of the classifier trained on $\mathcal{S}$ over the entire validation set. When computing the Shapley value in classification problems, the vast majority of the previous work [3, 5, 17, 21, 25, 26, 35, 41, 43] uses classification accuracy as the utility function.

However, using accuracy as the utility function may not sufficiently capture and differentiate the marginal contributions of training data points for data valuation. Accuracy is the percentage of correctly predicted instances. Thus it overlooks valuable details for valuation. For example in Figure 1, given a binary classification problem where the classification threshold is 50% and a validation set $\mathcal{V}$ containing four data points. Consider two coalitions of training data points $\mathcal{S}$ and $\mathcal{S} \cup \{z\}$ ($z \notin \mathcal{S}$). We observe that classifier C_1 trained on $\mathcal{S}$ provides confidence scores (i.e., predicted probabilities) of $\{60\%, 30\%, 70\%, 60\%\}$ for the ground truth labels of validation points, while classifier C_2 trained on $\mathcal{S} \cup \{z\}$ provides scores of $\{80\%, 30\%, 70\%, 80\%\}$. Although C_2 demonstrates greater

*Jinfei Liu is the corresponding author.

Proceedings of the VLDB Endowment, Vol. 17, No. 7 ISSN 2150-8097.
doi:10.14778/3654621.3654638

predictive confidence, suggesting non-negligible marginal contributions of z, both classifiers have an identical classification accuracy of $3/4 = 75\%$, which means the Shapley value based on accuracy would show 0 marginal contribution of z. While other metrics such as AUROC (Area Under the ROC Curve) [18] that consider the probability output can be more discriminating than accuracy, the aggregate nature of such metrics still does not capture the subtle differences of the probability output.

Can we design a new utility function that better captures the marginal contributions of the data points towards the probability output of the model? In this paper, we propose to use the average predicted probability [8], called *raw probability*, as the utility function. It is defined as $\overline{Pr}(\hat{y} = y) = \frac{1}{|\mathcal{V}|}\sum_{i=1}^{|\mathcal{V}|} Pr(\hat{y}_i = y_i)$, where $|\mathcal{V}|$ is the size of validation set, y_i is the ground truth label for the i^{th} validation data point, and $Pr(\hat{y}_i = y_i)$ indicates the predicted probability score that the model assigns the i^{th} validation data point to class label y_i. We show that the raw probability can be mapped to accuracy Acc by a piecewise function $f(x)$, where values greater than classification threshold θ are mapped to 1 and otherwise mapped to 0, i.e., $Acc = \frac{1}{|\mathcal{V}|}\sum_{i=1}^{|\mathcal{V}|} f(Pr(\hat{y}_i = y_i))$. A distinctive property of raw probability is that even if the accuracy is the same, the raw probability might considerably differ. Therefore, the raw probability serves as a powerful metric to discern the utility of models trained over different coalitions and results in a *probability-based Shapley* (P-Shapley) value for data valuation on probabilistic classifiers.

To further improve the efficacy of raw probability, we delve into the intrinsic property of predicted probability. From the perspective of information theory, as an event becomes more likely to happen, the potential for additional information decreases [7]. This implies that the higher the predicted probability, the harder the improvement. For instance, a shift in predicted probability score from 60% to 70% differs significantly from the shift from 90% to 100%. In most scenarios, the latter is perceived as more valuable and tougher to achieve. To capture the non-linearity in the improvement of the predicted probability score, we combine the utility function with calibration functions to re-evaluate the improvement in predicted probability. Specifically, we utilize convex calibration functions to enlarge the utility change when the predicted probability is high. Our experimental results on four real-world datasets show that the proposed P-Shapley value outperforms existing Shapley value-based data valuation approaches across various downstream ML tasks. Raw probability enjoys stable superiority for data valuation compared to prevalent metrics for classification evaluation [18] including accuracy, F_1 score, and AUROC (aka AUC). Besides, using calibration functions for contribution re-evaluation can further improve the performance on the same tasks.

The Shapley value is one of the most prevalent data valuation approaches, so there is a rich body of studies on the variants of the Shapley value. The main novelty of the paper is that we propose a new utility function, raw probability, for rethinking the selection of utility function for probabilistic classifiers, which utilizes the more nuanced predicted probability, thereby sharpening utility differentiation. We summarize our contributions as follows.

- We propose P-Shapley value by introducing a new utility function, raw probability, which allows for a more refined use of predicted probability and enhances utility distinction (Section 3).
- We further offer several convex calibration functions for probability calibration to effectively quantify the marginal utility contribution of each data point (Section 4).
- Experiments on various evaluation tasks with different machine learning models are conducted, which verify the effectiveness of our proposed P-Shapley value and the enhancements introduced by calibration functions (Section 5).

The rest of the paper is organized as follows. Section 2 reviews the related work on data valuation and activation functions that can be used to enlarge the utility change when the predicted probability is high. We review the concept of Shapley value, develop the P-Shapley value, and analyze the benefits of raw probability in Section 3. In Section 4, we introduce calibration functions to further improve the utility discrimination and present the theoretical analysis of the advantages of the calibrated P-Shapley value. We report the experimental results and findings in Section 5. Finally, we conclude the paper in Section 6.

2 RELATED WORK

In this section, we discuss related work on data valuation and activation functions that can be used to enlarge the utility change when the predicted probability is high.

Data valuation. The emergence of data markets [3, 23, 25, 26, 30] and federated learning [22, 24, 44] significantly expand data sharing. For data sharing towards machine learning tasks [2, 13, 42], data valuation plays a pivotal role in measuring the contribution of each participant's data, aiding in valuing data for data products as well as improving training efficiency and effectiveness by pinpointing high-value data points. A common way for data valuation is the leave-one-out (LOO) method [6], which evaluates data through alterations in model loss upon the modification of a data point. However, it lacks fairness guarantees. To address this issue, the Shapley value has recently been used to quantify the contributions of data points towards training machine learning models [14]. The performance of a model trained using a subset of the training data and tested on a holdout validation set is often used as the utility function. The Shapley value measuring the average marginal contribution of individual data points [11, 17, 33, 34, 41] has been extensively studied and used in compensation allocation [14, 26], outlier detection [20], and data selection [34]. Ghorbani et al. [16] propose distributional Shapley to measure the value of data points where the dataset is drawn independently and identically from the underlying distribution. Moreover, Song et al. [37] introduce Shapley value to federated learning to evaluate the contribution of each client. Recent advancements in research have brought about a diversification in the computation of the Shapley value for data valuation. Beta Shapley [21] refines the Shapley axioms to reduce noise, and CS-Shapley [35] offers a nuanced class-wise accuracy metric, underscoring the data point's contribution to its class. To overcome the increasing model bias and unfairness [1, 4, 45], Pombal et al. [31] propose fairness-aware Shapley value with a novel utility function combining model performance and fairness.

The most related literature to our work is [20], which deals with the data valuation for the specific task on K-nearest neighbor (KNN) classifier. In particular, the study mainly focuses on the unweighted KNN classifier and determines the utility of a KNN classifier based

Table 1: The summary of frequently used notations.

Notation	Definition
n	the size of dataset
$\mathcal{U}(\cdot)$	the accuracy-based utility function
$\mathcal{U}_p(\cdot)$	the probability-based utility function
$f(\cdot)$	the piecewise function for classification
$\mathcal{SV}$	the Shapley value
$\mathcal{PSV}$	the probability-based Shapley value
z_i	the i^{th} data point in the training set
z'_k	the k^{th} data point in the validation set

on the likelihood of the correct label, which can also be viewed as the predicted probability. However, they use the likelihood of correct classifications to efficiently estimate the Shapley value (for KNN) instead of improving the quality of data valuation results. Furthermore, [20] is specific for KNN classifiers and does not apply to general classifiers while we focus on general classification tasks. We note that this work focuses on data valuation for data management in the database community instead of model interpretability in the ML community commonly focusing on feature importance.

Activation function. As we mentioned in Section 1, the increase of the predicted probability score is non-linear, i.e., the higher the predicted probability score, the harder the improvement. The primary advantage of using activation functions in neural networks lies in their ability to introduce non-linearity. To capture the non-linearity in the improvement of the predicted probability score, we combine the raw probability with convex calibration functions selected from activation functions [12, 28, 32], which can enlarge the utility change when the predicted probability is high. The simplistic Square x^2 [12] meets our criteria of being convex within interval $[0, 1]$ with an output range of $[0, 1]$, but has a pronounced derivative increase in $[0, 1]$ and swift value growth due to its constant second derivative of 2. Among renowned activation functions [12] like ReLU, Leaky ReLU, Parametric ReLU, Sigmoid, Tanh, Mish, and Swish, only Mish [28] and Swish [32] exhibit convexity within interval $[0, 1]$ with an output range of $[0, 1]$. Therefore, we employ Mish and Swish to enlarge the utility change when the predicted probability is high. Specifically, Mish is characterized by consistent higher-order derivatives within $[0, 1]$ and Swish requires a customized parameter with akin derivative saturation traits. The properties facilitate more refined calibrations and potentially optimize the differentiation of marginal contribution for data valuation.

3 PROBABILITY-BASED SHAPLEY VALUE

In this section, we review the notion of the Shapley value, propose the P-Shapley value based on raw probability, and analyze the advantages of the P-Shapley value. For reference, Table 1 summarizes the frequently used notations.

3.1 Shapley Value

Consider a set of data points $\mathcal{D} = \{z_1, \ldots, z_n\}$. A *coalition* $\mathcal{S}$ is a subset of $\mathcal{D}$ that cooperates to complete an ML task, for instance, training an ML model. A utility function $\mathcal{U}(\mathcal{S})$ $(\mathcal{S} \subseteq \mathcal{D})$ is the utility of a coalition $\mathcal{S}$ for an ML task, which is typically the classification accuracy of the model trained on $\mathcal{S}$. The *marginal contribution* of data point z_i with respect to a coalition $\mathcal{S}$ $(z_i \notin \mathcal{S})$ is $\mathcal{U}(\mathcal{S} \cup \{z_i\}) - \mathcal{U}(\mathcal{S})$. The traditional Shapley value typically employs an accuracy-based utility function, denoted as $\mathcal{U}(\cdot)$,

$$\mathcal{U}(\mathcal{S}) = \frac{1}{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{V}|} f(Pr(\hat{y}_k = y_k)), \tag{1}$$

where y_k represents the ground truth label of the k^{th} data point z'_k in the validation set, $\hat{y}_k$ stands for the predicted label of z'_k by the probabilistic classifier trained on $\mathcal{S}$, $Pr(\hat{y}_k = y_k)$ represents the predicted probability associated with the ground truth label for z'_k, and $f(\cdot) : [0, 1] \rightarrow \{0, 1\}$ is defined with classification threshold θ for binary classification and multi-label multi-class classification as follows

$$f(x; \theta) = \begin{cases} 1, & \text{if } x \geq \theta, \\ 0, & \text{if } x < \theta. \end{cases} \tag{2}$$

By default, θ is set to $\frac{1}{|C|}$, where $|C|$ denotes the number of classes in the classification problem.

The Shapley value $\mathcal{SV}_i$ measures the expectation of marginal contribution by data point z_i in all possible coalitions over $\mathcal{D}$. That is,

$$\mathcal{SV}_i = \frac{1}{n} \sum_{\mathcal{S} \subseteq \mathcal{D} \setminus \{z_i\}} \frac{\mathcal{U}(\mathcal{S} \cup \{z_i\}) - \mathcal{U}(\mathcal{S})}{\binom{n-1}{|\mathcal{S}|}}. \tag{3}$$

Computing the exact Shapley value has to enumerate the utilities of all coalitions which is proved to be #P-hard [9].

3.2 P-Shapley Value

The vast majority of prior work [3, 5, 17, 21, 25, 26, 35, 41, 43] uses classification accuracy as the utility function. However, as mentioned in Section 1, using raw probability as the utility function is expected to capture the impact of one datum better than classification accuracy. We propose the first *probability-based utility function* that allows us to better measure the performance of probabilistic classifiers for general classification tasks.

Given a classification problem, for any data point $z_i \in \mathcal{D}$ $(1 \leq i \leq n)$, we need to quantify the contribution of data point z_i to the probabilistic classifier for the binary classification task. Let $\mathcal{V}$ be the validation set. For a given data coalition $\mathcal{S}$ $(\mathcal{S} \subseteq \mathcal{D})$, the probability-based utility function $\mathcal{U}_p(\cdot)$ is defined as follows.

$$\mathcal{U}_p(\mathcal{S}) = \overline{Pr}(\hat{y} = y) = \frac{1}{|\mathcal{V}|} \sum_{i=k}^{|\mathcal{V}|} Pr(\hat{y}_k = y_k). \tag{4}$$

Compared with $\mathcal{U}(\mathcal{S})$, $\mathcal{U}_p(\mathcal{S})$ removes the piecewise function $f(\cdot)$ to reveal the raw probability of the classifier trained on $\mathcal{S}$.

We accumulate a data point z_i's marginal contribution in all possible coalitions over $\mathcal{D}$ as the P-Shapley value $\mathcal{PSV}_i$. That is,

$$\mathcal{PSV}_i = \frac{1}{n} \sum_{\mathcal{S} \subseteq \mathcal{D} \setminus \{z_i\}} \frac{\left(\mathcal{U}_p(\mathcal{S} \cup \{z_i\}) - \mathcal{U}_p(\mathcal{S})\right)}{\binom{n-1}{|\mathcal{S}|}}. \tag{5}$$

In terms of computational efficiency, since computing the P-Shapley value requires enumerating the utilities of all coalitions as computing the Shapley value, it is easy to see that both have the same computational complexity, i.e., #P-hard. Compared to using accuracy in computing the Shapley value, the only difference of using raw probability is avoiding the step of determining whether the classification probability of the right class is above the classification

threshold. The time cost of this step is negligible compared to the cost of model training which is required by all methods.

P-Shapley value algorithm. Similar to the computation of Shapley value, calculating the exact P-Shapley value requires exponential time complexity. Therefore, we adopt an approximate truncated Monte Carlo algorithm [17] to tackle the computational challenge of estimating P-Shapley values. The pseudocode is shown in Algorithm 1. Specifically, we randomly sample m permutations of the training set (Lines 2-3). For each permutation, we scan the data points progressively and evaluate the utility of the coalition consisting of the scanned data points (Lines 6-10). We then accumulate each data point's marginal contribution (Lines 11-12). To reduce the computational cost, we adopt a truncated threshold τ such that the gap between the utility of the coalition consisting of the scanned data points and the utility of the entire training set falls below τ (Lines 7-8). Finally, we return the average marginal contribution from all m permutations as an approximation of P-Shapley values (Lines 13-14).

Algorithm 1: Truncated Monte Carlo for P-Shapley value.

input : Training set $\mathcal{D} = \{z_1, \ldots, z_n\}$, number of total permutations m, truncated threshold τ.
output: P-Shapley value of training data points $\mathcal{PSV}_1, \ldots, \mathcal{PSV}_n$.

1 $\mathcal{PSV}_i \leftarrow 0$ $(1 \le i \le n)$;
2 **for** t = 1 to m **do**
3 $\pi^t \leftarrow$ random permutation of the training set $\mathcal{D}$;
4 $\mathcal{U}_p(\emptyset) = 0$;
5 Calculate $\mathcal{U}_p(\pi^t)$ using Equation 4;
6 **for** j = 1 to n **do**
   // Denote the first j data points in π^t as $\pi^t[:j]$
7 **if** $\mathcal{U}_p(\pi^t[:j]) - \mathcal{U}_p(\pi^t) < \tau$ **then**
8 $\mathcal{U}_p(\pi^t[:j]) = \mathcal{U}_p(\pi^t[:j-1])$;
9 **else**
10 Calculate $\mathcal{U}_p(\pi^t[:j])$ using Equation 4;
11 **for** i = 1 to n **do**
12 $\mathcal{PSV}_i$ += $\mathcal{U}_p(\pi^t[:j]) - \mathcal{U}_p(\pi^t[:j-1])$;
13 **for** i = 1 to n **do**
14 $\mathcal{PSV}_i$ /= m;
15 **return** $\mathcal{PSV}_1, \ldots \mathcal{PSV}_n$;

3.3 Theoretical Analysis

In this section, we show that the probability-based utility function is superior in preserving more Shannon entropy, differentiating clean and noisy data points, and enhancing computational stability.

3.3.1 Shannon Entropy. We use Shannon Entropy as the metric of the amount of uncertainty or randomness in a set of data. It's widely used in information theory to represent the average information content one can expect to gain from observing a random variable. Here, we show the definitions of Shannon entropy for discrete random variables and continuous random variables, respectively.

Definition 1 (Shannon Entropy for discrete Random Variables). *Let X^d be a discrete random variable with probability mass function $p^d(x)$, defined over a set $\mathcal{X}$ of all possible outcomes. The Shannon entropy $H(X^d)$ of X^d is defined as*

$$H(X^d) = -\sum_{x \in \mathcal{X}} p^d(x) \log_2 p^d(x),$$

where the sum is taken over all x such that $p^d(x) > 0$.

Definition 2 (Shannon Entropy for Continuous Random Variables). *Let X^c be a continuous random variable with probability density function $p^c(x) : \mathbb{R} \to \mathbb{R}^+ \cup \{0\}$. The Shannon entropy $H(X^c)$ of X^c is defined as*

$$H(X^c) = -\int_{-\infty}^{\infty} p^c(x) \log_2 p^c(x) dx,$$

provided the integral exists.

As mentioned in Section 3.2, the probability-based utility function, denoted as $\mathcal{U}_p(\cdot)$, can be transformed into the accuracy-based utility function, $\mathcal{U}(\cdot)$, utilizing a designated piecewise function $f(\cdot)$. By viewing the probability-based utility and the accuracy-based utility as a continuous random variable $\boldsymbol{X}$ and a discrete random variable $\boldsymbol{Y}$, respectively, we prove that the informational content embedded in the probability-based utility is greater than or equal to that in the accuracy-based utility as follows.

Theorem 3.1. *Let $\boldsymbol{X}$ be a continuous random variable defined on interval $[0, 1]$ with a probability density function $p^c(x)$. Define a new discrete random variable $\boldsymbol{Y}$ such that $\boldsymbol{Y} = 1$ if $\boldsymbol{X} > \theta$ and $\boldsymbol{Y} = 0$ if $\boldsymbol{X} \le \theta$, where θ is a given threshold $0 < \theta < 1$. Then, the entropy $H(\boldsymbol{Y})$ of $\boldsymbol{Y}$ satisfies $H(\boldsymbol{Y}) \le H(\boldsymbol{X})$.*

Proof. For ease of presentation, we move the brief proofs of the theorems in the paper to the appendix while providing more detailed proofs in our technical report [40]. □

According to Theorem 3.1, the probability-based utility function preserves more Shannon entropy than the accuracy-based utility function, making it more effective for data valuation.

3.3.2 discrimination& Stability. A key role of data valuation methods is differentiating clean and noisy data points. However, Kwon and Zou [21] point out that the marginal contribution generated by the accuracy-based utility function becomes indistinguishable rapidly as the increase of coalition size. Furthermore, approximating the Shapley value is time-consuming due to the repeated evaluation of the utility function. Enhanced computational stability implies that a required approximation can be achieved with fewer utility samples and consequently save computational resources. In this section, we explain why the P-Shapley value can better differentiate clean and noisy data points' contribution and why the P-Shapley value exhibits superior stability by introducing Theorem 3.2 and Theorem 3.3, respectively.

In this section, we use the setting of the binary classification and the multi-label multi-class classification since the conclusions can be applied to the multi-class classification by breaking down the multi-class classification problem into multiple binary classification problems. When we focus on the k^{th} data point in the validation set, the probability-based utility function of coalition $\mathcal{S}$ is

$$\mathcal{U}_p(\mathcal{S}; k) = Pr(\hat{y}_k = y_k). \tag{6}$$

Meanwhile, the accuracy-based utility function of coalition $\mathcal{S}$ for the k^{th} data point in validation set can be expressed as $\mathcal{U}(\mathcal{S}; k) = f(\mathcal{U}_p(\mathcal{S}; k); \frac{1}{|C|})$, where $f(\cdot)$ is the piecewise function in Equation 2 and $|C|$ is the number of classes in the classification problem. Note that we can interpret $\mathcal{U}_p(\cdot)$ as a function that starts at $\frac{1}{|C|}$

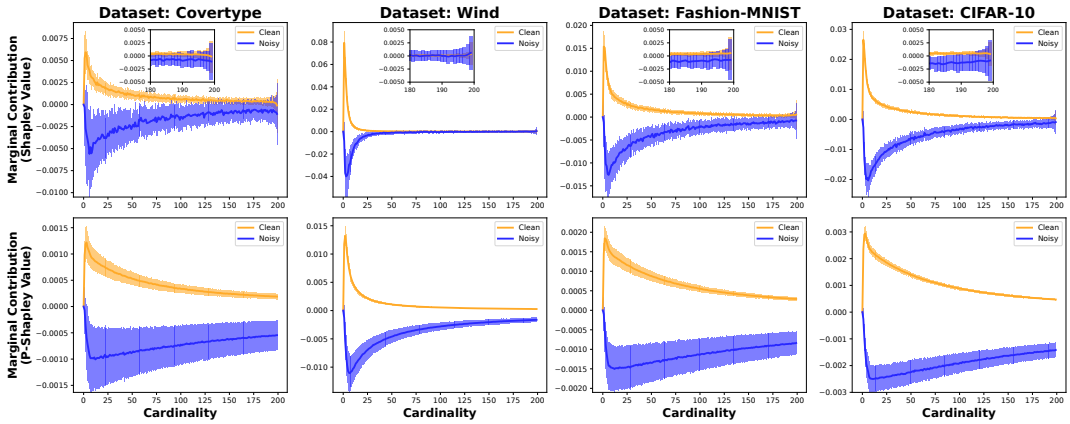

Figure 2: Illustrations of marginal contributions by cardinality on four real-world datasets [39]. The first row shows accuracy-based marginal contributions for Shapley values, while the second displays probability-based ones for P-Shapley values.

and increases (or sometimes decreases) to 1. We divide $\mathcal{U}_p(\cdot)$ into three terms as follows

$$\mathcal{U}_p(\mathcal{S};k) = \alpha(k;\mathcal{S})P(\mathcal{S}) + \frac{1}{|C|} + \epsilon,\ \epsilon \sim \mathcal{N}(0,\sigma^2), \tag{7}$$

where ϵ is an error term stemming from the model's intrinsic error and σ denotes its standard deviation. Notably, ϵ, which follows a zero-mean Normal distribution, is different and independent across varied coalitions and validation data points. $P(\mathcal{S})$ shows the contribution of subset $\mathcal{S}$, while $\alpha(k;\mathcal{S})$ shows the effect of the contribution of subset $\mathcal{S}$ on the k^{th} data point in the validation set. Specifically, $P(\emptyset) = 0$.

Assumptions. For Equation 7, we make the following assumptions, which hold in general scenarios.

Assumption 1. *Each data point z_k $(1 \le k \le |\mathcal{V}|)$ in the validation set is predictable, i.e., for all k, $\mathcal{S}$, $\alpha(k;\mathcal{S}) > 0$.*

Assumption 1 typically holds when the training and validation sets are not wholly irrelevant since the training set provides useful information for classification.

Assumption 2. *$P(\mathcal{S}) < P(\mathcal{S}\cup\{z\})$ and $|\alpha(k;\mathcal{S}') - \alpha(k;\mathcal{S})| \ll |P(\mathcal{S}') - P(\mathcal{S})|$ for all $k, \mathcal{S}, \mathcal{S}', z$ when $|\mathcal{S}|$ is large enough and $||\mathcal{S}'| - |\mathcal{S}|| \le 1$.*

Assumption 2 illustrates that the decline in the prediction results is attributed to the model's intrinsic error ϵ rather than issues with the quality of the dataset. And the rest part of the assumption can be accomplished by choosing a suitable $P(\cdot)$.

Assumption 3. *Only the predictions for $\eta|\mathcal{V}|$ data points (denoted by the set $\mathcal{A}$) in the validation set satisfy that $\alpha(k;\mathcal{S}) < 2\sigma$, and for the $\eta'|\mathcal{V}|$ data points (denoted by the set $\mathcal{A}'$) of $\mathcal{A}$, $\alpha(k;\mathcal{S}) \ll \sigma$.*

Assumption 3 shows that only a subset of data points in the validation set denoted as $\mathcal{A}$, frequently exhibit prediction errors. These points in $\mathcal{A}$, influenced by the model's intrinsic errors, still have some likelihood of prediction errors even when $|\mathcal{S}|$ is large. A subset of $\mathcal{A}$, denoted as $\mathcal{A}'$, is more profoundly affected by the model's intrinsic errors, maintaining a significant probability of prediction errors even with a large coalition size. For the other data points, their impact on the marginal contribution of the utility function in accuracy can be disregarded when $|\mathcal{S}|$ is large enough. We note that a scenario when two identical raw probabilities result in significantly different accuracy values is possible but is negligible compared to the probability of obtaining the same accuracy.

Higher discrimination. Based on Assumptions 1, 2, and 3, we present a theorem to illustrate that compared with Shapley values on accuracy-based utility function, P-Shapley values can effectively *mitigate the diminishing returns* of marginal contributions and *maintain the discrimination* of marginal contribution for clean and noisy data points.

Theorem 3.2. *For all i, we have $\mathrm{E}[\mathcal{PSV}_{i,j}] > \mathrm{E}[\mathcal{SV}_{i,j}]$ when cardinality j is large enough, where $\mathcal{SV}_{i,j}$ ($\mathcal{PSV}_{i,j}$) measures the expectation of marginal contribution generated by accuracy-based (probability-based) utility function by data point z_i in all possible coalitions with cardinality j over $\mathcal{D}$, called as stratified Shapley (P-Shapley) value.*

An intuitive illustration for Theorem 3.2 is that P-Shapley values capture all the changes in predicted probabilities caused by the addition of new data points. When the predicted probability is significantly greater than θ, the changes caused by the addition of new data points are hard for Shapley values to capture. We note that when the marginal contribution is consistently negative (the case of noisy data), due to the effect of multiplying by a negative number on both sides, the inequality in Theorems 3.2 will reverse. Therefore, the absolute value of P- Shapley values will be greater than that of Shapley values for a large cardinality, which helps differentiate clean and noisy data points.

Better stability. Based on Assumption 3, we present a theorem to illustrate that the variance of P-Shapley values is lower than that of Shapley values. This suggests that P-Shapley values exhibit superior performance against the intrinsic errors of the model, rendering it more stable than Shapley values.

Theorem 3.3. *For all i, we have $\mathrm{Var}\left(\mathcal{PSV}_{i,j}\right) < \mathrm{Var}\left(\mathcal{SV}_{i,j}\right)$ when cardinality j is large enough.*

The reduction in variance is mainly because the accuracy-based utility of a data point in the validation set is a discrete random variable with values $\{0, 1\}$. When it is greatly affected by the model's intrinsic errors, the variance will be large. This issue can be effectively circumvented by P-Shapley values, as the probability-based utility is continuous rather than discrete. Given that P-Shapley values will be larger than Shapley values as the cardinality increases (Theorem 3.2), a smaller variance of P-Shapley values (Theorem 3.3) implies that P-Shapley values possess better stability.

Case study. Now we attempt to empirically verify the higher discrimination and better stability through a case study. We directly compare the marginal contribution of mislabeled (noisy) and correctly labeled (clean) data points generated by accuracy (*resp.* raw probability) for Shapley values (*resp.* P-Shapley values). We set the training set size $n = 200$ and assume observed data can be mislabeled. Specifically, we flip the original label for a random 10% of data points in $\mathcal{D}$. Figure 2 shows the marginal contributions for clean and noisy samples as a function of the cardinality over four real-world datasets from OpenML [39]. Each color indicates the noisy (blue)/clean (yellow) data points' marginal contribution. We denote a 95% confidence band based on 50 independent runs. As the cardinality increases, the difference between the noise (blue) and clean (yellow) data points in terms of Shapley values becomes less distinct due to its small absolute value, and it is significantly perturbed by the intrinsic errors of the model. In contrast, P-Shapley values, with their larger absolute value, still exhibit a noticeable difference and are less affected by the model's intrinsic errors.

4 CALIBRATED P-SHAPLEY VALUE

In the P-Shapley value, the intrinsic non-linearity property of improving the predicted probability scores is ignored. For example, the increase in predicted probability score from 90% to 100% is more challenging than the increase from 60% to 70% in general. In this section, we incorporate several convex calibration functions to capture the non-linearity property of improving the predicted probability scores in Section 4.1 and prove the primary advantages of these calibration functions in Section 4.2.

4.1 Calibration Functions

As the predicted probability score nears its peak (i.e., 100%), subsequent enhancements become progressively difficult. This non-linearity can be explained as an effect of marginal diminishing law in ML [21]. We capture this non-linearity by slowing the rate of diminishing marginal contribution. Therefore, we underscore the importance of the marginal improvement made by data points when the predicted probability is high and propose the calibrated probability-based utility function $\mathcal{U}_p^+(\mathcal{S})$ as follows.

$$\mathcal{U}_p^+(\mathcal{S}) = \frac{1}{|\mathcal{V}|} \sum_{z_k \in \mathcal{V}} CF\left(Pr(\hat{y}_k = y_k)\right), \tag{8}$$

where $CF(\cdot)$ is the calibration function. Based on the calibrated probability-based utility function $\mathcal{U}_p^+(\cdot)$, we can measure the expectation of marginal contribution by data point z_i in all possible coalitions over $\mathcal{D}$ as z_i's calibrated P-Shapley value, $\mathcal{PSV}_i^+$,

Table 2: Mathematical Expressions for Selected Calibration Functions (*β defaults to 1).

Calibration Function	Mathematical Expression
Square	$y = x^2$
Mish	$y = x \tanh\left[\ln(1 + \exp(x)\right]$
Swish*	$y = x\left[1 + \exp(-\beta x)\right]^{-1}$

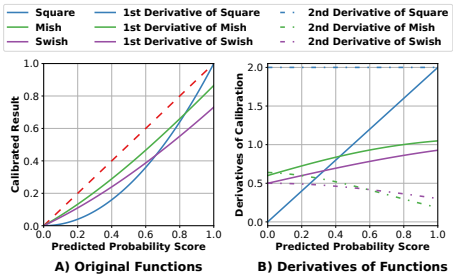

Figure 3: Graphical Expressions for Calibration Functions.

$$\mathcal{PSV}_i^+ = \frac{1}{n} \sum_{\mathcal{S} \subseteq \mathcal{D} \setminus \{z_i\}} \frac{\mathcal{U}_p^+(\mathcal{S} \cup \{z_i\}) - \mathcal{U}_p^+(\mathcal{S})}{\binom{n-1}{|\mathcal{S}|}}. \tag{9}$$

As computing the calibrated P-Shapley value is highly similar to computing the P-Shapley value, Algorithm 1 can be easily applied to approximate the calibrated P-Shapley value.

Selected Calibration Functions. We select several convex activation functions as the calibration functions, including Square, Mish, and Swish as listed in Table 2. These functions consistently output values within interval $[0, 1]$ when given inputs in the same range (corresponding to the range of predicted probabilities). With their positive second derivatives, they emphasize the significance of increases in predicted probability, particularly when it's already high, as depicted in Figure 3. Moreover, to maintain the positive and negative marginal contribution generated by the probability-based utility function, we require that the calibration function is always strictly monotonically increasing.

The most straightforward calibration function with a positive second derivative is x^2, which meets all our expected requirements for the calibration function, and its derivative exhibits a noticeable increase over interval $[0, 1]$. However, it may cause an overly rapid value growth when the predictive confidence score is high due to its constant second derivative of 2. The Mish function is convex over interval $[0, 1]$. Compared with the square function, its derivative exhibits a more gradual variation, and its n^{th} order derivatives do not degenerate to zero. Moreover, we can consider a convex activation function with parameters, Swish, allowing us to adjust its performance. Also, the derivatives of Swish saturate beyond a threshold, similar to Mish. The non-linear saturating form of these calibration functions' derivatives allows for a smoother value increment proportional to the predicted probability scores, yielding more accurate calibration results for the utility function. Therefore, we are also interested in their performance.

4.2 Theoretical Analysis

In this section, we first present several lemmas to show that convex functions can slow down the diminishing returns of marginal contributions. Subsequently, we show why P-Shapley values calibrated

by convex calibration functions can more effectively outperform in distinguishing the contributions of clean and noisy data points in terms of peak normalization and share in total utility, respectively.

Slowing diminishing rate. We introduce Lemma 4.1, positing that when a convex function encompasses a monotonically increasing function $U(\cdot)$ with a decreasing derivative (i.e., a concave function), the diminishing rate of its derivative is lower than that of $U(\cdot)$. This makes the convex function effective as a calibration function to slow down the diminishing rate of marginal contribution.

Lemma 4.1. *If $U(\cdot)$ is a strictly monotonically increasing derivable positive function and $CF(\cdot)$ is a convex positive calibration function, then for $\forall x_1 < x_2$, we have $\frac{(CF\circ U)'(x_2)}{(CF\circ U)'(x_1)} > \frac{U'(x_2)}{U'(x_1)}$.*

Based on Lemma 4.1, we can obtain Corollary 4.2.

Corollary 4.2. *If $U(\cdot)$ is a strictly monotonically increasing derivable positive function and $CF(\cdot)$ is a convex strictly monotonically increasing positive function, then for $\forall x_1 \leq x_2 - 1$, we have*

$$\frac{CF \circ U(x_2+1) - CF \circ U(x_2)}{CF \circ U(x_1+1) - CF \circ U(x_1)} > \frac{U(x_2+1) - U(x_2)}{U(x_1+1) - U(x_1)}, \tag{10}$$

Corollary 4.2 shows that when applying a convex calibration function, the diminishing returns of the utility function's marginal contribution (as the input increases) become less pronounced than before, meaning the marginal contributions become easier to capture when the cardinality is large.

Moreover, we need to consider whether the conclusion still holds after averaging over different coalitions of the same cardinality to apply it to the stratified P-Shapley value. We introduce Lemma 4.3 to demonstrate that the calibrated stratified P-Shapley values have a certain degree of consistency with the marginal contributions calculated by accumulating the P-Shapley values and then applying the calibration function.

Lemma 4.3. *$CF(\cdot)$ is a convex strictly monotonically increasing positive function. Define $F_i(j) = \sum_{p=1}^{j-1} \mathcal{PSV}_{i,p}$, we have*

$$\begin{aligned} \liminf_{j\to n, n\to\infty} \frac{CF(F_i(j+1)) - CF(F_i(j))}{\mathcal{PSV}^+_{i,j}} &= C_1, \\ \limsup_{j\to n, n\to\infty} \frac{CF(F_i(j+1)) - CF(F_i(j))}{\mathcal{PSV}^+_{i,j}} &= C_2, \\ \frac{C_2}{C_1} &\leq \frac{CF'(U_{max})}{CF'(U_{min})}, \end{aligned} \tag{11}$$

where $U_{max} = \lim_{n\to\infty} \sup_k \mathcal{U}_p(\mathcal{D};k)$, $U_{min} = \lim_{n\to\infty} \inf_k \mathcal{U}_p(\mathcal{D};k)$.

Peak normalization. Based on the aforementioned lemmas and corollaries, we can prove that the diminishing returns of the calibrated P-Shapley values will be more gradual compared with P-Shapley values. In other words, the calibrated P-Shapley values can better differentiate clean and noisy data points as the cardinality increases. Theorem 4.4 states that after normalizing the marginal contribution with cardinality j_0 (we typically select the peak), calibrated P-Shapley values will be greater than P-Shapley values when cardinality after cardinality threshold N.

Theorem 4.4. *Given that $CF(\cdot)$ is a convex calibration function, if for cardinality j_0, $\mathcal{PSV}^+_{i,j_0} = \mathcal{PSV}_{i,j_0}$ and $\frac{CF'(n\mathcal{PSV}_i)}{CF'(F_i(j_0))} > \frac{CF'(U_{max})}{CF'(U_{min})}$, we can find threshold N, $\forall j > N$, $\mathcal{PSV}^+_{i,j} > \mathcal{PSV}_{i,j}$.*

Utility normalization. Unlike Theorem 4.4, we consider this issue from the perspective of the model's overall utility. Given that the calibration functions we selected have a value of zero at the origin, we derive Theorem 4.5 which states that after normalizing the total utility, calibrated stratified P-Shapley values will be greater than stratified P-Shapley values after cardinality threshold j_0.

Theorem 4.5. *If a convex calibration strictly monotonically increasing function s.t. $CF(0) = 0$, we can find threshold $N, \forall j > N$, we have*

$$\frac{\mathcal{PSV}^+_{i,j}}{CF(\mathcal{U}_p(\mathcal{D}))} > \frac{\mathcal{PSV}_{i,j}}{\mathcal{U}_p(\mathcal{D})}. \tag{12}$$

In summary, Theorem 4.4 and Theorem 4.5 are complementary ways of stating the diminishing rate of marginal contribution can be further slowed down by the introduction of a convex calibration function, respectively. Theorem 4.4 compares the peak normalized P-Shapley values before and after the introduction of the convex calibration function, while Theorem 4.5 considers the share of each in the total utility. We conclude that the introduction of the convex calibration function further helps us to distinguish between the contributions of clean and noisy data points.

Furthermore, if the conditions in Lemma 4.1 change from convex functions to concave functions, then due to the opposite trend in the derivatives, the direction of the inequalities in the conclusion will reverse. Consequently, the direction of the inequalities in Theorem 4.4 and Theorem 4.5 will also reverse. This implies that the result will be counterproductive if we choose the concave function as the calibration function.

5 EXPERIMENTS

In this section, we present the empirical evaluation of the proposed algorithms on diverse classification datasets and compare their performance with existing accuracy-based data valuation methods. In Section 5.1, we provide details of the experimental setup including the datasets and compared methods. In Section 5.2, we propose detailed metrics for measuring the predictive confidence score in the data removal experiment as well as the median coefficient of variation in the stability comparison experiment. We conduct experiments on computational stability, high-value data removal, and noisy data detection in Sections 5.3, 5.4, and 5.5, respectively. In Section 5.6, we empirically demonstrate the advantages and potential applications of the probability-based utility function used by the P-Shapley value in record-level model interpretability.

5.1 Datasets and Experimental Setup

In this section, we describe the datasets, base models, and experimental setup.

Datasets and Base Models. We employ four real-world datasets from OpenML [39] that are commonly used to benchmark classification methods and implement a logistic regression classifier. We follow the standard methodology used in previous work [21, 35] to extract features from image datasets including Fashion-MNIST and CIFAR-10. Specifically, we utilize the pre-trained ResNet-18 [19] model available in PyTorch to extract image representations. We then perform principal component analysis (PCA) on the representations and select the top 32 principal components as features.

Throughout the experiments, we use a logistic regression model (LR) and a support vector machine model (SVM) as the base models.

Compared Methods. We augment the proposed P-Shapley value with three different calibration functions including Square, Mish [28], and Swish [32] and named as P-Shapley (Raw), P-Shapley (Square), P-Shapley (Mish), P-Shapley (Swish) accordingly. We compare them with the following baseline algorithms: Leave-One-Out [6], truncated Monte Carlo approximated Shapley (TMC-Shapley) [17], and Beta Shapley ($\alpha = 1, \beta = 16$) [21]. We truncate in the same iteration when estimating P-Shapley value, TMC-Shapley value, and Beta-Shapley value with the truncated Monte Carlo algorithm as mentioned in Algorithm 1.

5.2 Evaluation Metrics

In this section, we introduce the metrics for the experiment.

Median Coefficient of Variation (Median CV). To assess the computational stability of the P-Shapley value, we adopt the median coefficient of variation (Median CV) to capture the degree of variability across various data points. Given a set of estimated Shapley values $\left\{\overline{\mathcal{SV}_i^1}, \ldots, \overline{\mathcal{SV}_i^t}\right\}$ $(1 \le i \le n)$ obtained by computing t times using the same algorithm under the same experiment setting, where $\overline{\mathcal{SV}_i^t}$ denotes the t^{th} estimated Shapley value of z_i computed by the algorithm, the median coefficient of variation is

$$Median\ CV = \underset{1\le i\le n}{\text{Med}} \left\{ \frac{\sqrt{\frac{1}{t}\sum_{j=1}^{t}\left(\overline{\mathcal{SV}_i^t} - \frac{1}{t}\sum_{m=1}^{t}\overline{\mathcal{SV}_i^m}\right)^2}}{\left|\frac{1}{t}\sum_{j=1}^{t}\overline{\mathcal{SV}_i^j}\right|} \right\}. \tag{13}$$

Weighted Accuracy Drop (WAD). To quantify the overall accuracy drop and its rate for various data valuation methods, we adopt the weighted accuracy drop (WAD) [35] as a metric. Given a training set $\mathcal{D}$ in descending order by data value and removing data points progressively starting with the highest value data point, WAD is calculated by aggregating the prediction accuracy decrease in each round, with weight inversely proportional to the number of rounds.

$$WAD = \sum_{j=1}^{n}\left(\frac{1}{j}\sum_{i=1}^{j}\left(ACC_{\mathcal{D}[i-1:]} - ACC_{\mathcal{D}[i:]}\right)\right), \tag{14}$$

where $\mathcal{D}[i:]$ represents the slice of $\mathcal{D}$ starting from the i^{th} data point, indicating that the first $i-1$ data points have been removed. $ACC_{\mathcal{D}}[i:]$ represents the corresponding prediction accuracy of the probabilistic classifier trained on the remaining data. For boundary cases, we define $\mathcal{D}[0:]$ as the entire training set.

Weighted Brier Score Drop (WBD). To assess the impact on predictive confidence scores more accurately, we propose the incorporation of predicted class probabilities with weighted performance drops. Brier score (BS) is a measure of the accuracy of predicted class probabilities made by a probabilistic classifier. As in Equation 15, it is calculated as the mean squared difference between the predicted class probabilities $\Pr(\hat{y}_k = y_k)$ and target label y_k.

$$BS = \frac{1}{|\mathcal{V}|}\sum_{k=1}^{|\mathcal{V}|}(\Pr(\hat{y}_k = y_k) - y_k)^2 \tag{15}$$

By combining the Brier score and WAD metrics, we introduce the probability-level Weighted Brier Score Drop (WBD) measure. This metric offers a probability-based approach to evaluating model performance that considers both the effect of data point removal on model performance and its predictive confidence scores.

$$WBD = -\sum_{j=1}^{n}\left(\frac{1}{j}\sum_{i=1}^{j}\left(BS_{\mathcal{D}[i-1:]} - BS_{\mathcal{D}[i:]}\right)\right). \tag{16}$$

Weighted Cross Entropy Drop (WCD). Similarly, we introduce cross-entropy (CE) to calculate the cumulative change in the model's predictive confidence scores and calculate Weighted Cross Entropy Drop (WCD) accordingly.

$$CE = -\sum_{k=1}^{|\mathcal{V}|}\left(y_k \log \Pr(\hat{y}_k = y_k) + (1 - y_k)\log(1 - \Pr(\hat{y}_k = y_k))\right), \tag{17}$$

$$WCD = -\sum_{j=1}^{n}\left(\frac{1}{j}\sum_{i=1}^{j}\left(CE_{\mathcal{D}[i-1:]} - CE_{\mathcal{D}[i:]}\right)\right). \tag{18}$$

For WAD, WBD, and WCD, a higher value suggests that removing high-value data points leads to a steeper decline in model utility, which shows the enhanced capability of identifying the high-value data important for model utility.

5.3 Computational Stability Comparison

We first assess the computational stability of the P-Shapley value. We repeatedly calculate the SV values and use the Median CV as an indicator to measure the stability of Shapley values for all points. The Median CV quantifies the ratio of the standard deviation to the mean in data distribution. A lower Median CV indicates less variability. As is shown in Table 3, the PSV algorithm has demonstrated a superior level of stability compared with the traditional utility-based SV algorithm, as it consistently exhibits lower Median CV values across multiple datasets. This underscores the heightened stability of the PSV algorithm. In the compared methods, TMC-Shapley (AUC) and TMC-Shapley (LogLoss) denote the TMC-Shapley value replacing accuracy with AUC and LogLoss respectively. Moreover, Square and Mish perform better among the four P-Shapley values in computational stability since they can reduce the model's intrinsic errors more effectively with generally higher derivatives.

Table 3: Median CV for different datasets across methods.

Method	Covertype	Wind	FMNIST	CIFAR-10
Beta-Shapley	0.624	0.747	0.602	0.896
TMC-Shapley	0.300	0.217	0.262	0.276
TMC-Shapley (AUC)	0.321	0.173	0.246	0.227
TMC-Shapley (LogLoss)	0.197	0.188	0.223	0.271
P-Shapley (Raw)	0.243	0.169	0.209	0.242
P-Shapley (Square)	0.185	0.165	0.166	0.208
P-Shapley (Swish)	0.228	0.168	0.194	0.230
P-Shapley (Mish)	0.167	0.163	0.152	0.201

5.4 High-value Data Removal

We conduct high-value data removal experiments to evaluate the effectiveness of our proposed data valuation methods. In these experiments, we iteratively remove data points from the dataset in descending order of their assessed value. Training data points with higher valuation should contribute more to the model performance, so we measure the performance of each data valuation method with the performance drop after removing high-value data points.

Figures 4 and 5 depict a decrease in prediction accuracy as the highest value data point is sequentially removed. The proposed

Table 4: WAD, WBD, WCD Drop for High-value data removal (LR).

	Covertype			Wind			Fashion-MNIST			CIFAR-10		
	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑
Leave-One-Out	0.194	0.152	1.197	0.157	0.100	0.693	0.271	0.181	0.606	0.109	0.100	0.595
Beta-Shapley	0.210	0.175	1.630	0.225	0.186	4.191	0.251	0.180	0.757	0.110	0.078	0.323
TMC-Shapley	0.318	0.263	1.737	0.373	0.303	3.911	0.373	0.281	1.211	0.136	0.102	0.462
TMC-Shapley (AUC)	0.339	0.300	2.175	0.376	0.324	5.041	0.416	0.328	1.518	0.100	0.071	0.284
TMC-Shapley (LogLoss)	0.041	0.049	0.665	0.173	0.156	1.675	0.068	0.072	0.650	0.106	0.083	0.519
P-Shapley (Raw)	0.381	0.301	2.475	0.398	0.323	4.457	0.397	0.307	1.404	0.140	0.108	0.487
P-Shapley (Square)	0.428	0.342	2.478	0.439	0.371	5.389	0.475	0.377	1.778	0.224	0.176	0.843
P-Shapley (Swish)	0.401	0.318	**2.509**	0.413	0.341	4.768	0.424	0.335	1.571	0.166	0.129	0.591
P-Shapley (Mish)	**0.436**	**0.349**	2.403	**0.449**	**0.381**	**5.466**	**0.490**	**0.390**	**1.823**	**0.235**	**0.185**	**0.894**

Table 5: WAD, WBD, WCD for High-value data removal (SVM).

	Covertype			Wind			Fashion-MNIST			CIFAR-10		
	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑	WAD↑	WBD↑	WCD↑
Leave-One-Out	0.044	0.042	0.320	0.116	0.135	1.359	0.154	0.127	0.581	0.045	0.020	0.119
Beta-Shapley	0.181	0.143	1.799	0.152	0.079	0.552	0.174	0.105	0.343	0.055	0.028	0.118
TMC-Shapley	0.075	0.052	0.886	0.248	0.142	1.114	0.261	**0.168**	0.325	0.067	0.029	0.108
TMC-Shapley (AUC)	0.189	0.128	1.282	0.205	0.124	1.415	0.187	0.116	0.349	0.072	**0.044**	0.105
TMC-Shapley (LogLoss)	0.041	0.049	0.665	0.078	0.090	1.126	0.042	0.028	0.331	0.043	0.033	0.101
P-Shapley (Raw)	0.219	0.115	0.879	0.340	0.291	**3.409**	0.495	0.081	0.360	**0.405**	0.026	0.103
P-Shapley (Square)	**0.309**	0.172	3.862	0.342	**0.294**	2.902	0.494	0.088	0.393	0.383	0.032	**0.128**
P-Shapley (Swish)	0.240	0.134	1.352	0.342	0.291	3.158	**0.497**	0.082	0.367	0.404	0.027	0.107
P-Shapley (Mish)	0.303	**0.189**	**5.901**	**0.344**	0.291	2.707	0.484	0.099	**0.435**	0.387	0.031	0.124

P-Shapley value approach, utilizing all three calibration functions, exhibits a faster decrease in accuracy as data points are removed. This indicates that the P-Shapley value captures the importance of the data more precisely, allowing for more efficient data reduction. Moreover, the P-Shapley value with Square, Mish, and Swish calibration functions shows a faster reduction rate compared with ReLU, highlighting the efficacy of these non-linear calibration functions. The results of the P-Shapley series on SVM are similar. One possible reason is that compared to LR, the robustness of SVM weakens the performance difference among activation functions.

Tables 4 and 5 display the reduction rates of all compared methods using the WAD, WBD, and WCD metrics, as defined in Section 5.2. Upward arrows indicate that the larger the value, the better. P-Shapley values utilizing all three calibration functions consistently outperform the baselines across all four datasets. Notably, the P-Shapley value with the Swish calibration function achieves the highest WBD and WCD scores. One possible reason is that the Swish activation's soft clipping nature helps produce a utility function that varies smoothly with the changes in predictive confidence score, resulting in well-calibrated data valuation.

5.5 Noisy Data Detection

We now investigate the detection ability of P-Shapley. To introduce label noise, we randomly shuffle the labels of 20% in the training data. We compute value estimates on the noised training sets using each valuation method and then simulate manual inspection by checking data labels from the lowest value to the highest value. The expectation is that an effective data valuation method can assign low values to mislabeled instances relative to the correctly labeled instances. We compute the Area Under the Curve (AUC) of the precision-recall (PR) curve for quantitative results. The PR-AUC results for logistic regression and SVM are presented in Table 6. The larger the value, the better the capability in noisy data detection. We find that P-Shapley outperforms other SV-based baselines across various datasets, which verifies the superiority of P-Shapley values in noisy data detection. Notably, the ability of P-Shapley values with different activation functions to detect labels varies across datasets. This variation is related to the differences among datasets. Therefore, considering both experimental results in Section 5.4 and Section 5.5, we should determine different activation functions for P-Shapley values depending on the specific scenario and Mish enjoys the best compatibility.

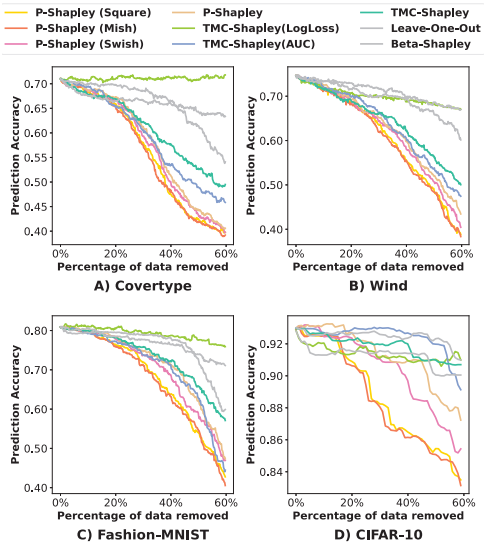

Figure 4: Results for High-value Data Removal (LR).

5.6 Record-level Model Interpretability

Besides the efficacy in data valuation, the P-Shapley value can be a tool for record-level model interpretability. Notably, record-level model interpretability is distinct from (feature-level) model interpretability in the ML community. While feature-level model interpretability aims to understand the contribution or importance of

Table 6: Area Under the Curve (PR-AUC) for LR and SVM.

	Logistic Regression				SVM			
	Covertype	Wind	Fashion-MNIST	CIFAR-10	Covertype	Wind	Fashion-MNIST	CIFAR-10
Leave-One-Out	0.622	0.502	0.498	0.554	0.510	0.475	0.426	0.536
Beta-Shapley	0.630	0.738	0.790	0.940	0.530	0.797	0.666	0.776
TMC-Shapley	0.678	0.758	0.828	0.982	0.547	0.842	0.701	0.961
TMC-Shapley (AUC)	0.635	0.744	0.841	0.978	0.552	0.852	0.663	0.968
TMC-Shapley (LogLoss)	0.621	0.734	0.819	0.975	0.526	0.793	0.682	0.933
P-Shapley (Raw)	0.680	0.800	0.857	0.981	0.547	0.892	0.848	0.970
P-Shapley (Square)	0.700	**0.806**	0.857	0.985	0.581	0.891	**0.854**	**0.972**
P-Shapley (Swish)	0.689	0.803	**0.858**	**0.986**	0.557	**0.892**	0.850	0.971
P-Shapley (Mish)	**0.707**	0.806	0.856	0.985	**0.590**	0.891	0.854	0.972

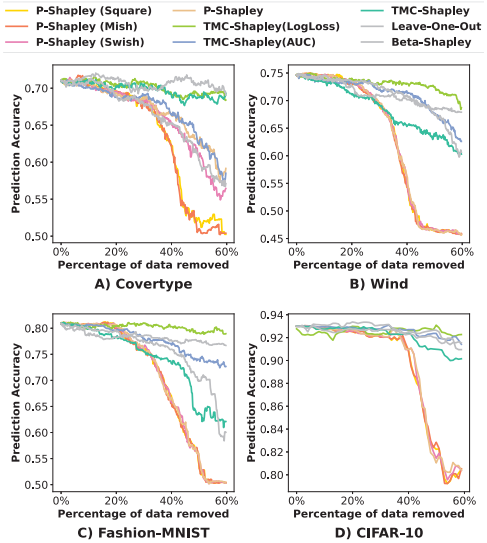

Figure 5: Results for High-value Data Removal (SVM).

each feature to the model output, record-level model interpretability is aimed at measuring the contribution of different data points (i.e., data records) to the model performance for a specific data point. For probabilistic classifiers, the final predicted probability of each validation data point often deviates from the base prediction before training with data, such as the neutral 0.5 (50%) in binary classification. It is therefore tempting to ask: how does the training data shape the model's output, specifically the predicted probability corresponding to the correct label for each validation data point?

The Shapley value can allocate the change in correctness to the contribution of different training data. However, the change in correctness is relatively rough compared to the change in predicted probability. Naturally, the P-Shapley value can be applied to refine the measurement. As illustrated in Figure 6, we select truck and automobile images from CIFAR-10 to form a binary classification problem, extract features with ResNet-50, and employ logistic regression as the classifier ($\theta = 0.5$). We then organize these images into five disjoint clusters per class using k-means clustering [18] and sample six validation images. Data clusters 0-4 and validation images 0-2 are trucks. Data clusters 5-9 and validation images 3-5 are automobiles. By regarding each data cluster as a participant in the cooperative game (i.e., model training) and using the raw probability (resp. accuracy) as the utility, we have the P-Shapley value (resp. Shapley value) of each data cluster for each validation image shown in the grids. The P-Shapley value measures the contribution of each data cluster on the final classification probability, e.g., for validation image 0, data cluster 0 makes a 44% increase in the predicted probability of the correct label (i.e., truck). The final probability can be derived by aggregating the effects of all data clusters over the base probability, for instance, the first row in Figure 6 indicates $0.5 + 0.440 - 0.038 - 0.058 + 0.083 + 0.063 - 0.270 - 0.055 + 0.035 + 0.045 + 0.126 = 0.871$. In contrast, Shapely values measure the contribution to the classification correctness.

We find that the prediction of validation image 4 is wrong as the predicted probability for the correct label is not larger than $\theta = 0.5$. According to the P-Shapley value, the predicted probability is most swayed by data cluster 2. When data cluster 2 is eliminated from the training data and then retraining the model, the prediction probability is boosted from 0.318 to 0.740 and consequently results in a correct prediction for validation image 4. It showcases the potential of the P-Shapley value in measuring data contribution finely. Besides, we conduct similar experiments on a basic deep learning model, multi-layer perceptron (MLP), and the results shown in Figure 7 verify the effectiveness of P-Shapley values.

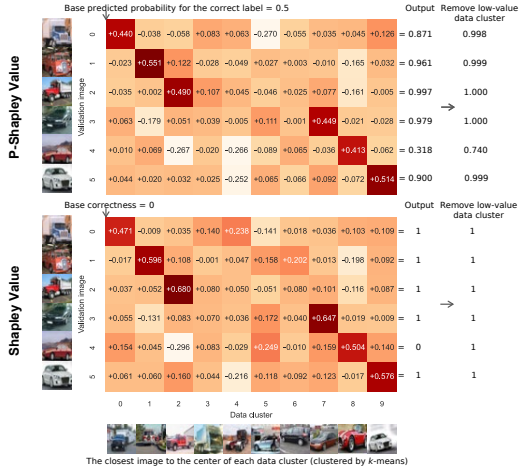

Figure 6: The contribution of each data cluster for the predicted probability (*resp.* prediction correctness) of each validation image on CIFAR-10, and darker colors indicate larger P-Shapley values (*resp.* Shapley values).

5.7 Discussion

In this section, we provide a concise overview of the benefits of employing the probabilistic utility function and compare it to the traditional utility based on accuracy. As mentioned in Section 5.3, P-Shapley values consistently exhibit strong stability when subjected to repeated computations. Also, P-Shapley's alignment with the true value of data is evident in the high-value point removal and

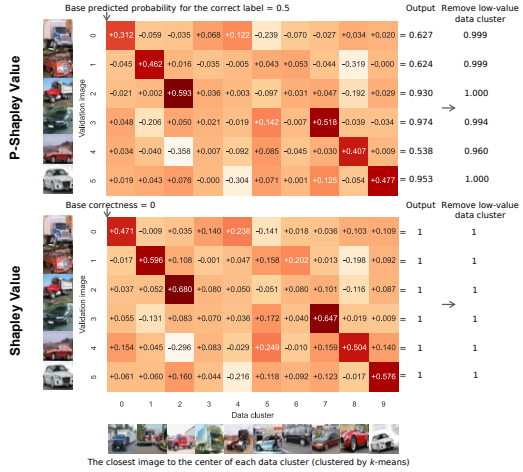

Figure 7: The contribution of each data cluster for the deep learning model's predicted probability (*resp.* prediction correctness) of each validation image on CIFAR-10, and darker colors indicate larger P-Shapley values (*resp.* Shapley values).

noisy data detection experiments, detailed in Sections 5.4 and 5.5. The interpretability of these values is demonstrated in Section 5.6.

In our exploration, we further ventured into extended Shapley Value with AUC-based, F_1-score-based, and log-loss-based utility. The AUC-based and log-loss-based Shapley values, in comparison to our P-Shapley, showed less optimal performance as shown in the high-value point removal and noisy data detection experiments. On the other hand, the F_1-score-based values displayed a significant class bias, favoring one class over the other. We believe this observed bias might be due to the imbalance in our dataset, causing the F_1 score to overvalue the majority class, given its heightened sensitivity. We further conducted experiments on perfectly balanced datasets using Shapley values based on the F_1 score. The results indicated that the aforementioned class bias was eliminated. However, the experimental performance still significantly lagged behind the Shapley values based on accuracy.

As for the selection of activation functions, according to the experimental results, Square and Mish perform better than Raw and Swish in most cases since they can reduce the model's intrinsic errors more effectively with generally higher derivatives. However, when the selected model is robust (e.g., SVM) and the dataset is easy to learn (e.g., Wind), Swish and Raw would perform better. Furthermore, all activation functions adopted in our paper possess convex properties (positive second derivatives). We also experimented with non-convex activation functions, such as the commonly used Sigmoid and Tanh. The WAD results from the high-value point removal experiments are illustrated in Table 7. The performance of non-convex functions is significantly inferior to that of convex functions. This empirically validates the correctness of our theoretical analysis presented in Section 4.2.

Table 7: WAD for P-Shapley with non-convex activation functions across datasets.

Method	Covertype	Wind	FMNIST	CIFAR-10
P-Shapley (Sigmoid)	0.102	0.165	0.167	0.063
P-Shapley (Tahn)	0.295	0.341	0.357	0.208

6 CONCLUSION

In this paper, we propose the P-Shapley value framework, a simple yet effective method for data valuation that incorporates a new probability-based utility function for a refined utility evaluation. We further suggest a series of convex calibration functions to capture the non-linearity property of improving the predicted probability scores by slowing the rate of diminishing marginal contribution. We prove that the P-Shapley value enjoys better computational stability as well as a higher ability to discern clean and noisy data. Experimental results on four real-world datasets show that the proposed approaches incorporating the probability-based utility function and the convex calibration functions outperform baseline methods in effectiveness.

There are several interesting directions for future research. While our TMC-based P-Shapley estimation is efficient to a certain degree, developing algorithms that enable the P-Shapley value to be effectively applied to large datasets remains a pivotal challenge. Meanwhile, exploring more utility functions for classification evaluation and applying the idea of raw probability to feature-level model interpretability can be further studied.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by the National Key RD Program of China (2021YFB3101100), NSFC grants (62102352, U23A20306), NSF grants (CNS-2124104, CNS-2125530), and NIH grants (R01LM013712, UL1TR002378).

A PROOFS OF THEOREMS

A.1 Proof of Theorem 3.1

Proof. Consider two random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ where $\boldsymbol{Y} = f(\boldsymbol{X})$. For each possible value of $\boldsymbol{Y}$, we have

$$H(\boldsymbol{Y}|\boldsymbol{X}) = \int_x p^c(x) H(\boldsymbol{Y}|\boldsymbol{X}=x) = \int_x p^c(x) H(f(\boldsymbol{X})|\boldsymbol{X}=x)$$
$$= \int_x p^c(x) \left[-p^d(f(x)) \log_2 p^d(f(x)|\boldsymbol{X}=x) \right] = 0 \tag{19}$$

where $f(\cdot)$ is a deterministic function. Using Bayes's rule, we have

$$H(\boldsymbol{X},\boldsymbol{Y}) = -\int \sum_y p^c(x) p^d(y|x) \log_2 [p^c(x) p^d(y|x)]\, dx$$
$$= H(\boldsymbol{X}) + H(\boldsymbol{Y}|\boldsymbol{X}) = H(\boldsymbol{X}), \tag{20}$$

where $H(\boldsymbol{Y}|\boldsymbol{X}) = 0$ according to Equation 19. Similarly, we have

$$H(\boldsymbol{X},\boldsymbol{Y}) = H(\boldsymbol{Y}) + H(\boldsymbol{X}|\boldsymbol{Y}). \tag{21}$$

Then we investigate $H(\boldsymbol{X}|\boldsymbol{Y})$,

$$H(\boldsymbol{X}|\boldsymbol{Y}) = \sum_y p^d(y) H(\boldsymbol{X}|\boldsymbol{Y}=y) \geq 0, \tag{22}$$

where $H(\boldsymbol{X}|\boldsymbol{Y}=y) = -\int_x p^c(x|y) \log_2 p^c(x|y) \geq 0$. According to Equations 20,21, and 22, we have $H(\boldsymbol{X}) = H(\boldsymbol{X},\boldsymbol{Y}) = H(\boldsymbol{Y}) + H(\boldsymbol{X}|\boldsymbol{Y}) \geq H(\boldsymbol{Y})$. This completes the proof. □

A.2 Proof of Theorem 3.2

A.2.1 Additional notations. we denote $\mathcal{S}^i = \mathcal{S} \cup \{z_i\}$, $p_k(\mathcal{S}) = \mathcal{U}_p(\mathcal{S};k)$, $\Delta_i(\mathcal{S}) = P(\mathcal{S}^i) - P(\mathcal{S})$ and $c_D = \frac{\mathcal{U}_p(\mathcal{D}) - \mathcal{U}_p(\emptyset)}{\mathcal{U}(\mathcal{D})}$, as well as assume all the intrinsic errors have same variance σ^2 for convenience. Specifically, we set $P(\mathcal{S}) = \mathcal{U}(\mathcal{S})$.

A.2.2 Key lemma.

Lemma A.1. *For all z, i, j, we have*

$$\lim_{j\to\infty} \sup_{|\mathcal{S}|=j-1} \left(\mathcal{U}_p(\mathcal{S} \cup \{z\}) - \mathcal{U}_p(\mathcal{S}) \right) = 0, \ \lim_{n\to\infty} n\mathcal{PSV}_i < \infty.$$

Consequently, $\mathcal{PSV}_{i,j} \to 0$ when $j \to \infty$.

Proof. This lemma is directly derived from Theorem 1 in [21]. □

A.2.3 Completing the proof of Theorem 3.2.

Proof. We can compute $\mathcal{PSV}_{i,j}$ directly by Equation 7 :

$$\begin{aligned}&\mathcal{PSV}_{i,j}\\&=\sum_{\mathcal{S}\subseteq\mathcal{D}\setminus\{i\},|\mathcal{S}|=j-1}\binom{n-1}{j-1}^{-1}\left(\mathcal{U}_p(\mathcal{S}^i)-\mathcal{U}_p(\mathcal{S})\right)\\&=\sum_{\mathcal{S}}\binom{n-1}{j-1}^{-1}\frac{1}{|\mathcal{V}|}\sum_{k\in\mathcal{V}}\left(P(\mathcal{S}^i)\alpha(k;\mathcal{S}^i)-P(\mathcal{S})\alpha(k;\mathcal{S})\right)+\epsilon^*.\end{aligned}\tag{23}$$

We have

$$\mathrm{E}[\mathcal{PSV}_{i,j}]\geq\binom{n-1}{j-1}^{-1}\frac{1}{|\mathcal{V}|}\sum_{\mathcal{S}}\left(P(\mathcal{S}^i)-P(\mathcal{S})\right)\left(\sum_{k\in\mathcal{V}}\alpha(k;\mathcal{S})\right).\tag{24}$$

Using Equation 7, we can obtain:

$$\binom{n-1}{j-1}^{-1}\frac{1}{|\mathcal{V}|}\left(\sum_{k\in\mathcal{V}}\alpha(k;\mathcal{S})\right)\geq\frac{\mathcal{U}_p(\mathcal{D})-\mathcal{U}_p(\emptyset)}{\mathcal{U}(\mathcal{D})}.\tag{25}$$

We now consider the marginal contribution of accuracy-based utility function for point k upon adding data i, and we consider only the points in the set $\mathcal{A}$ (mentioned in Assumption 3):

$$\begin{aligned}&\mathrm{E}[\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k)]\\&=Pr(p_k(\mathcal{S}^i)\geq\frac{1}{|C|})Pr(p_k(\mathcal{S})<\frac{1}{|C|})-Pr(p_k(\mathcal{S}^i)<\frac{1}{|C|})Pr(p_k(\mathcal{S})\geq\frac{1}{|C|}).\end{aligned}\tag{26}$$

According to Equation 7, $p_k(\mathrm{S})\sim\mathcal{N}(\alpha(k;\mathcal{S})P(\mathcal{S})+\frac{1}{2},\sigma^2)$, then

$$Pr(p_k(\mathcal{S})\geq\frac{1}{|C|})=\frac{1}{2}+\int_0^{\alpha(k;\mathcal{S})P(\mathcal{S})\sigma^{-1}}\psi(x)dx,\tag{27}$$

where $\psi(\cdot)$ denotes the PDF of standard normal distribution. Substituting this into the equation, we obtain:

$$\mathrm{E}[\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k)]\leq\frac{\psi(0)}{\sigma}\left(\alpha(k;\mathcal{S}^i)P(\mathcal{S}^i)-\alpha(k;\mathcal{S})P(\mathcal{S})\right).\tag{28}$$

According to Assumption 2 and Assumption 3, we can get:

$$\mathrm{E}[\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k)]\leq 2\psi(0)\Delta_i(\mathcal{S}).\tag{29}$$

Combing with Equation 25, we can get :

$$\mathrm{E}[\mathcal{SV}_{i,j}]\leq 2\eta_1\psi(0)\sum_{\mathcal{S}}\Delta_i(\mathcal{S})<c_D\sum_{\mathcal{S}}\Delta_i(\mathcal{S})\leq\mathrm{E}[\mathcal{PSV}_{i,j}].\tag{30}$$

This completes the proof. □

A.3 Proof of Theorem 3.3

Proof. According to Equation 7, we have:

$$\mathrm{Var}(\mathcal{U}_p(\mathcal{S}^i;k)-\mathcal{U}_p(\mathcal{S};k))=\mathrm{Var}(\epsilon)+\mathrm{Var}(\epsilon)=2\sigma^2.\tag{31}$$

Now we consider $X=\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k)$ with $k\in\mathcal{A}'$, which is a discrete random variable taking values of 1, 0, or -1 with $EX\approx 0$ and $Pr(0)\approx\frac{1}{2}$ since Equation 27. We can get:

$$\mathrm{Var}(X)=EX^2-(EX)^2\approx EX^2=Pr(1)+Pr(-1)\approx\frac{1}{2}.\tag{32}$$

In this case, we can assume that $2\sigma^2<\frac{1}{2}\eta'(\sigma<\frac{1}{2}\sqrt{\eta'})$, We have $\mathrm{Var}(\mathcal{U}_p(\mathcal{S}^i;k)-\mathcal{U}_p(\mathcal{S};k))<\eta'^2Var(\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k))$. Combing with Equation 23, we can get

$$\begin{aligned}&\mathrm{Var}(\mathcal{PSV}_{i,j})\\&=\sum_{\mathcal{S}}\binom{n-1}{j-1}^{-2}\frac{1}{|\mathcal{V}|^2}\sum_{k\in\mathcal{V}}\mathrm{Var}\left(\mathcal{U}_p(\mathcal{S}^i;k)-\mathcal{U}_p(\mathcal{S};k)\right)\\&<\sum_{\mathcal{S}}\binom{n-1}{j-1}^{-2}\frac{1}{|\mathcal{V}|^2}\sum_{k\in\mathcal{A}'}\mathrm{Var}\left(\mathcal{U}(\mathcal{S}^i;k)-\mathcal{U}(\mathcal{S};k)\right)\\&\leq\mathrm{Var}(\mathcal{SV}_{i,j}).\end{aligned}\tag{33}$$

This completes the proof. □

A.4 Proof of Lemma 4.1

Proof. Since $CF(\cdot)$ and $U(\cdot)$ are strictly monotonically increasing and positive with $CF(\cdot)$ convex, we have $\frac{(CF\circ U)'(x_2)}{(CF\circ U)'(x_1)},\frac{U'(x_2)}{U'(x_1)}>0\,CF'(U(x_2))>CF'(U(x_1))>0$. Since $(CF\circ U)'(x)=CF'(U(x))U'(x)$, we can get:

$$\frac{(CF\circ U)'(x_2)}{(CF\circ U)'(x_1)}=\frac{CF'(U(x_2))U'(x_2)}{CF'(U(x_1))U'(x_1)}=\frac{CF'(U(x_2))}{CF'(U(x_1))}\cdot\frac{U'(x_2)}{U'(x_1)}.$$

Since $CF'(U(x_2))>CF'(U(x_1))>0$, we have $\frac{CF'(U(x_2))}{CF'(U(x_1))}>1$, which implies $\frac{(CF\circ U)'(x_2)}{(CF\circ U)'(x_1)}>\frac{U'(x_2)}{U'(x_1)}$. □

A.5 Proof of Corollary 4.2

Proof. By Lemma 4.1, we can obtain that $\frac{(CF\circ U)'(x)}{U'(x)}$ is strictly monotonically increasing. According to Cauchy's mean value theorem, $\exists\check{\xi}_1\in(x_1,x_1+1),\check{\xi}_2\in(x_2,x_2+1)$ s.t.

$$\begin{aligned}\frac{CF\circ U(x_1+1)-CF\circ U(x_1)}{U(x_1+1)-U(x_1)}&=\frac{(CF\circ U)'(\check{\xi}_1)}{U'(\check{\xi}_1)},\\\frac{CF\circ U(x_2+1)-CF\circ U(x_2)}{U(x_2+1)-U(x_2)}&=\frac{(CF\circ U)'(\check{\xi}_2)}{U'(\check{\xi}_2)}.\end{aligned}\tag{34}$$

Since $\check{\xi}_1<\check{\xi}_2$ we have $\frac{(CF\circ U)'(\check{\xi}_1)}{U'(\check{\xi}_1)}<\frac{(CF\circ U)'(\check{\xi}_2)}{U'(\check{\xi}_2)}$, then we can get

$$\frac{CF\circ U(x_2+1)-CF\circ U(x_2)}{U(x_2+1)-U(x_2)}>\frac{CF\circ U(x_1+1)-CF\circ U(x_1)}{U(x_1+1)-U(x_1)},\tag{35}$$

which is consistent with the conclusion because every term in the inequality is greater than 0.

□

A.6 Proof of Theorem 4.4

Proof. We interpolate $F_i(\cdot)$ as a continuously derivable function. According to Corollary 4.2, we have $\forall j>j_0$,

$$\frac{CF\circ F_i(j+1)-CF\circ F_i(j)}{CF\circ F_i(j_0+1)-CF\circ F_i(j_0)}>\frac{\mathcal{PSV}_{i,j}}{\mathcal{PSV}_{i,j_0}}.\tag{36}$$

More specifically, the ratio of the left-hand side to the right-hand side of the above inequality tends to $\frac{CF'(F_i(j))}{CF'(F_i(j_0))}$. Since $\frac{CF'(F_i(n))}{CF'(F_i(j_0))}>\frac{CF'(U_{max})}{CF'(U_{min})}$, combing with Lemma 4.1, we can get $\exists N,\forall j>N$,

$$\frac{\mathcal{PSV}^+_{i,j}}{\mathcal{PSV}^+_{i,j_0}}>\frac{\mathcal{PSV}_{i,j}}{\mathcal{PSV}_{i,j_0}}.\tag{37}$$

Since $\mathcal{PSV}^+_{i,j_0}\geq\mathcal{PSV}_{i,j_0}$, we have

$$\mathcal{PSV}^+_{i,j}>\mathcal{PSV}_{i,j}.\tag{38}$$

This completes the proof. □

A.7 Proof of Theorem 4.5

Proof. Since $CF(0)=0$, we have $\frac{CF(y)-CF(x)}{CF(x)}>\frac{y-x}{x}$. Then, we can obtain that

$$\frac{\left(CF(\mathcal{U}_p(\mathcal{S}^i;k))-CF(\mathcal{U}_p(\mathcal{S};k))\right)}{CF(\mathcal{U}_p(\mathcal{S};k))}>\frac{\left(\mathcal{U}_p(\mathcal{S}^i;k)-\mathcal{U}_p(\mathcal{S};k)\right)}{\mathcal{U}_p(\mathcal{S};k)}.\tag{39}$$

According to Jensen's inequality,

$$\begin{aligned}\frac{\mathcal{PSV}^+_{i,j}}{CF(\overline{\mathcal{U}_p(\mathcal{S})})}>\frac{\mathcal{PSV}^+_{i,j}}{\overline{CF(\mathcal{U}_p(\mathcal{S}))}}&=\frac{\sum_{\mathcal{S},k}\left(CF(\mathcal{U}_p(\mathcal{S}^i;k))-CF(\mathcal{U}_p(\mathcal{S};k))\right)}{\sum_{\mathcal{S},k}CF(\mathcal{U}_p(\mathcal{S};k))}\\&>\frac{\sum_{\mathcal{S},k}\left(\mathcal{U}_p(\mathcal{S}^i;k)-\mathcal{U}_p(\mathcal{S};k)\right)}{\sum_{\mathcal{S},k}\mathcal{U}_p(\mathcal{S};k)}\,(*)\\&=\frac{\mathcal{PSV}_{i,j}}{\overline{\mathcal{U}_p(\mathcal{S})}}.\end{aligned}\tag{40}$$

Since $\lim_{j\to n}\overline{\mathcal{U}_p(\mathcal{S})}=\mathcal{U}_p(\mathcal{D})$, we can get that $\exists N,\forall j>N$, $\frac{\mathcal{PSV}^+_{i,j}}{CF(\mathcal{U}_p(\mathcal{D}))}>\frac{\mathcal{PSV}_{i,j}}{\mathcal{U}_p(\mathcal{D})}$. □

Remark 1. *The step $(*)$ does not hold again in any arbitrary case, but in this paper we consider it to hold given that $\mathcal{U}_p(\mathcal{S})$ is not very different.*

REFERENCES

[1] Mohit Bajaj, Lingyang Chu, Vittorio Romaniello, Gursimran Singh, Jian Pei, Zirui Zhou, Lanjun Wang, and Yong Zhang. 2022. Revealing Unfair Models by Mining Interpretable Evidence. *CoRR* abs/2207.05811 (2022). https://doi.org/10.48550/ARXIV.2207.05811 arXiv:2207.05811

[2] Luca Bonomi, Sepand Gousheh, and Liyue Fan. 2023. Enabling Health Data Sharing with Fine-Grained Privacy. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, Ingo Frommholz, Frank Hopfgartner, Mark Lee, Michael Oakes, Mounia Lalmas, Min Zhang, and Rodrygo L. T. Santos (Eds.). ACM, 131–141. https://doi.org/10.1145/3583780.3614864

[3] Lingjiao Chen, Paraschos Koutris, and Arun Kumar. 2019. Towards Model-based Pricing for Machine Learning in a Data Marketplace. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, Peter A. Boncz, Stefan Manegold, Anastasia Ailamaki, Amol Deshpande, and Tim Kraska (Eds.). ACM, 1535–1552. https://doi.org/10.1145/3299869.3300078

[4] Lingyang Chu, Lanjun Wang, Yanjie Dong, Jian Pei, Zirui Zhou, and Yong Zhang. 2021. FedFair: Training Fair Models In Cross-Silo Federated Learning. *CoRR* abs/2109.05662 (2021). arXiv:2109.05662 https://arxiv.org/abs/2109.05662

[5] Shay B. Cohen, Eytan Ruppin, and Gideon Dror. 2005. Feature Selection Based on the Shapley Value. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005*, Leslie Pack Kaelbling and Alessandro Saffiotti (Eds.). Professional Book Center, 665–670. http://ijcai.org/Proceedings/05/Papers/0763.pdf

[6] R. Dennis Cook. 1977. Detection of Influential Observation in Linear Regression. *Technometrics* 19, 1 (1977), 15–18. http://www.jstor.org/stable/1268249

[7] Thomas M Cover. 1999. *Elements of information theory*. John Wiley & Sons.

[8] Olga V Demler, Nina P Paynter, and Nancy R Cook. 2015. Tests of calibration and goodness-of-fit in the survival setting. *Statistics in medicine* 34, 10 (2015), 1659–1680.

[9] Xiaotie Deng and Christos H. Papadimitriou. 1994. On the Complexity of Cooperative Solution Concepts. *Math. Oper. Res.* 19, 2 (1994), 257–266. https://doi.org/10.1287/MOOR.19.2.257

[10] Daniel Deutch, Nave Frost, Benny Kimelfeld, and Mikaël Monet. 2022. Computing the Shapley Value of Facts in Query Answering. In *SIGMOD '22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 - 17, 2022*, Zachary G. Ives, Angela Bonifati, and Amr El Abbadi (Eds.). ACM, 1570–1583. https://doi.org/10.1145/3514221.3517912

[11] Daniel Deutch, Amir Gilad, Tova Milo, and Amit Somech. 2020. ExplainED: Explanations for EDA Notebooks. *Proc. VLDB Endow.* 13, 12 (2020), 2917–2920. https://doi.org/10.14778/3415478.3415508

[12] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. 2022. Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing* 503 (2022), 92–108. https://doi.org/10.1016/J.NEUCOM.2022.06.111

[13] Liyue Fan. 2022. Privacy Challenges and Solutions for Image Data Sharing. In *4th IEEE International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications, TPS-ISA 2022, Atlanta, GA, USA, December 14-17, 2022*. IEEE, 55–57. https://doi.org/10.1109/TPS-ISA56441.2022.00017

[14] Raul Castro Fernandez. 2022. Protecting Data Markets from Strategic Buyers. In *SIGMOD '22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 - 17, 2022*, Zachary G. Ives, Angela Bonifati, and Amr El Abbadi (Eds.). ACM, 1755–1769. https://doi.org/10.1145/3514221.3517855

[15] Tianfan Fu, Tian Gao, Cao Xiao, Tengfei Ma, and Jimeng Sun. 2019. Pearl: Prototype learning via rule learning. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*. 223–232.

[16] Amirata Ghorbani, Michael P. Kim, and James Zou. 2020. A Distributional Framework For Data Valuation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research)*, Vol. 119. PMLR, 3535–3544. http://proceedings.mlr.press/v119/ghorbani20a.html

[17] Amirata Ghorbani and James Y. Zou. 2019. Data Shapley: Equitable Valuation of Data for Machine Learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA (Proceedings of Machine Learning Research)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.), Vol. 97. PMLR, 2242–2251. http://proceedings.mlr.press/v97/ghorbani19c.html

[18] Jiawei Han, Jian Pei, and Hanghang Tong. 2022. *Data mining: concepts and techniques*. Morgan kaufmann.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[20] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nezihe Merve Gürel, Bo Li, Ce Zhang, Costas J. Spanos, and Dawn Song. 2019. Efficient Task-Specific Data Valuation for Nearest Neighbor Algorithms. *Proc. VLDB Endow.* 12, 11 (2019), 1610–1623. https://doi.org/10.14778/3342263.3342637

[21] Yongchan Kwon and James Zou. 2022. Beta Shapley: a Unified and Noise-reduced Data Valuation Framework for Machine Learning. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event (Proceedings of Machine Learning Research)*, Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (Eds.), Vol. 151. PMLR, 8780–8802. https://proceedings.mlr.press/v151/kwon22a.html

[22] Haoyang Li, Qingqing Ye, Haibo Hu, Jin Li, Leixia Wang, Chengfang Fang, and Jie Shi. 2023. 3DFed: Adaptive and Extensible Framework for Covert Backdoor Attack in Federated Learning. In *44th IEEE Symposium on Security and Privacy, SP 2023, San Francisco, CA, USA, May 21-25, 2023*. IEEE, 1893–1907. https://doi.org/10.1109/SP46215.2023.10179401

[23] Yifan Li, Xiaohui Yu, and Nick Koudas. 2021. Data Acquisition for Improving Machine Learning Models. *Proc. VLDB Endow.* 14, 10 (2021), 1832–1844. https://doi.org/10.14778/3467861.3467872

[24] Zitao Li, Tianhao Wang, and Ninghui Li. 2023. Differentially Private Vertical Federated Clustering. *Proc. VLDB Endow.* 16, 6 (2023), 1277–1290. https://doi.org/10.14778/3583140.3583146

[25] Jinfei Liu, Qiongqiong Lin, Jiayao Zhang, Kui Ren, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. 2021. Demonstration of Dealer: An End-to-End Model Marketplace with Differential Privacy. *Proc. VLDB Endow.* 14, 12 (2021), 2747–2750. https://doi.org/10.14778/3476311.3476335

[26] Jinfei Liu, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. 2021. Dealer: An End-to-End Model Marketplace with Differential Privacy. *Proc. VLDB Endow.* 14, 6 (2021), 957–969. https://doi.org/10.14778/3447689.3447700

[27] Scott M. Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 4765–4774. https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

[28] Diganta Misra. 2019. Mish: A Self Regularized Non-Monotonic Neural Activation Function. *CoRR* abs/1908.08681 (2019). arXiv:1908.08681 http://arxiv.org/abs/1908.08681

[29] Jian Pei. 2022. A Survey on Data Pricing: From Economics to Data Science. *IEEE Trans. Knowl. Data Eng.* 34, 10 (2022), 4586–4608. https://doi.org/10.1109/TKDE.2020.3045927

[30] Jian Pei, Raul Castro Fernandez, and Xiaohui Yu. 2023. Data and AI Model Markets: Opportunities for Data and Model Sharing, Discovery, and Integration. *Proc. VLDB Endow.* 16, 12 (2023), 3872–3873. https://doi.org/10.14778/3611540.3611573

[31] José Pombal, Pedro Saleiro, Mário A. T. Figueiredo, and Pedro Bizarro. 2023. Fairness-Aware Data Valuation for Supervised Learning. *CoRR* abs/2303.16963 (2023). https://doi.org/10.48550/ARXIV.2303.16963 arXiv:2303.16963

[32] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. 2018. Searching for Activation Functions. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=Hkuq2EkPf

[33] Alon Reshef, Benny Kimelfeld, and Ester Livshits. 2020. The Impact of Negation on the Complexity of the Shapley Value in Conjunctive Queries. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2020, Portland, OR, USA, June 14-19, 2020*, Dan Suciu, Yufei Tao, and Zhewei Wei (Eds.). ACM, 285–297. https://doi.org/10.1145/3375395.3387664

[34] Sebastian Schelter, Stefan Grafberger, Shubha Guha, Bojan Karlas, and Ce Zhang. 2023. Proactively Screening Machine Learning Pipelines with ARGUSEYES. In *Companion of the 2023 International Conference on Management of Data, SIGMOD/PODS 2023, Seattle, WA, USA, June 18-23, 2023*, Sudipto Das, Ippokratis Pandis, K. Selçuk Candan, and Sihem Amer-Yahia (Eds.). ACM, 91–94. https://doi.org/10.1145/3555041.3589682

[35] Stephanie Schoch, Haifeng Xu, and Yangfeng Ji. 2022. CS-Shapley: Class-wise Shapley Values for Data Valuation in Classification. In *NeurIPS*. http://papers.nips.cc/paper_files/paper/2022/hash/df334022279996b07e0870a629c18857-Abstract-Conference.html

[36] Lloyd S Shapley. 1953. A value for n-person games. *Contributions to the Theory of Games* 2, 28 (1953), 307–317.

[37] Tianshu Song, Yongxin Tong, and Shuyue Wei. 2019. Profit Allocation for Federated Learning. In *2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019*, Chaitanya K. Baru, Jun Huan, Latifur Khan, Xiaohua Hu, Ronay Ak, Yuanyuan Tian, Roger S. Barga, Carlo Zaniolo, Kisung Lee, and Yanfang (Fanny) Ye (Eds.). IEEE, 2577–2586. https://doi.org/10.1109/BIGDATA47090.2019.9006327

[38] Prasang Upadhyaya, Magdalena Balazinska, and Dan Suciu. 2012. How to Price Shared Optimizations in the Cloud. *Proc. VLDB Endow.* 5, 6 (2012), 562–573. https://doi.org/10.14778/2168651.2168657

[39] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luís Torgo. 2013. OpenML: networked science in machine learning. *SIGKDD Explor.* 15, 2 (2013), 49–60. https://doi.org/10.1145/2641190.2641198

[40] Haocheng Xia, Xiang Li, Junyuan Pang, Jinfei Liu, Kui Ren, and Li Xiong. 2023. Shapley Values on Probabilistic Classifiers (Technical Report). *CoRR* abs/2306.07171 (2023). https://doi.org/10.48550/ARXIV.2306.07171 arXiv:2306.07171

[41] Haocheng Xia, Jinfei Liu, Jian Lou, Zhan Qin, Kui Ren, Yang Cao, and Li Xiong. 2023. Equitable Data Valuation Meets the Right to Be Forgotten in Model Markets. *Proc. VLDB Endow.* 16, 11 (2023), 3349–3362. https://doi.org/10.14778/3611479.3611531

[42] Min Xu, Bolin Ding, Tianhao Wang, and Jingren Zhou. 2020. Collecting and Analyzing Data Jointly from Multiple Services under Local Differential Privacy. *Proc. VLDB Endow.* 13, 11 (2020), 2760–2772. http://www.vldb.org/pvldb/vol13/p2760-xu.pdf

[43] Jiayao Zhang, Qiheng Sun, Jinfei Liu, Li Xiong, Jian Pei, and Kui Ren. 2023. Efficient Sampling Approaches to Shapley Value Approximation. *Proc. ACM Manag. Data* 1, 1 (2023), 48:1–48:24. https://doi.org/10.1145/3588728

[44] Huadi Zheng, Haibo Hu, and Ziyang Han. 2020. Preserving User Privacy for Machine Learning: Local Differential Privacy or Federated Machine Learning? *IEEE Intell. Syst.* 35, 4 (2020), 5–14. https://doi.org/10.1109/MIS.2020.3010335

[45] Zirui Zhou, Lingyang Chu, Changxin Liu, Lanjun Wang, Jian Pei, and Yong Zhang. 2021. Towards Fair Federated Learning. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, Feida Zhu, Beng Chin Ooi, and Chunyan Miao (Eds.). ACM, 4100–4101. https://doi.org/10.1145/3447548.3470814