Is Our Continual Learner Reliable? Investigating Its Decision Attribution Stability through SHAP Value Consistency

Yusong Cai*, Shimou Ling*, Liang Zhang, Lili Pan†, Hongliang Li
University of Electronic Science and Technology of China
Chengdu, China
lilipan@uestc.edu.cn

Abstract

In this work, we identify continual learning (CL) methods' inherent differences in sequential decision attribution. In the sequential learning process, inconsistent decision attribution may undermine the interpretability of a continual learner. However, existing CL evaluation metrics, as well as current interpretability methods, cannot measure the decision attribution stability of a continual learner. To bridge the gap, we introduce Shapley value, a well-known decision attribution theory, and define SHAP value consistency (SHAPC) to measure the consistency of a continual learner's decision attribution. Furthermore, we define the mean and the variance of SHAPC values, namely SHAPC-Mean and SHAPC-Var, to jointly evaluate the decision attribution stability of continual learners over sequential tasks. On Split CIFAR-10, Split CIFAR-100, and Split TinyImageNet, we compare the decision attribution stability of different CL methods using the proposed metrics, providing a new perspective for evaluating their reliability.

1. Introduction

Continual learning (CL) [3, 7, 20, 23, 25, 32] involves learning a sequence of tasks incrementally without significantly forgetting previously acquired knowledge [23]. Over recent years, various algorithms [1, 3, 4, 7, 20, 25, 31, 32] and network architectures [8, 18, 24] have been proposed to mitigate catastrophic forgetting in CL with varying degrees of success. Concurrently, numerous evaluation metrics have been proposed to assess the performance of CL methods, focusing on aspects like forgetting [31], accuracy [5], and transfer [21]. However, the interpretability of continual learning has received very little attention.

The interpretability of a machine learning model refers to how well humans can understand the reasons behind its decisions [19]. To effectively engage with humans, a machine learning model should align with human comprehension, acceptability, and intuition [30], necessitating good interpretability to establish trust among humans.

For a continual learner, existing interpretability methods can explain decisions for samples in the current task but cannot interpret the same sample's decisions after learning subsequent tasks, as the sample will not be available. Subsequent learning can alter decision attribution for samples in previous tasks as the model continually adapts to the data distribution of new tasks. As shown in Fig. 1, the decision attribution (highlighted area) for the same sample varies across different CL methods during the continual learning process. This indicates that the interpretability of learners changes during continual learning processes.

However, current CL evaluation metrics and interpretability methods cannot evaluate the decision attribution stability of continual learners, resulting in a decision attribution stability assessment gap (See the lower part of Fig. 1).

To bridge this gap, we introduce SHAP value consistency, building upon SHAP values [22], to measure the decision attribution stability of CL methods. Further, we propose SHAPC-Mean and SHAPC-Var to respectively assess the average SHAP value consistency throughout the entire CL process and the variation of SHAP value consistency across different samples. The main contributions of this work are summarized as follows:

- We, for the first time, disclose the truth that most CL methods have decision attribution changes across tasks. Furthermore, we show that various methods may result in varying degrees of decision behavior deviation.
- We introduce SHAP value and define SHAP value consistency (SHAPC), along with SHAPC-Mean and SHAPC-Var, as metrics to measure the decision attribution stability of continual learning methods.
- We conduct comprehensive experiments on three prominent CL datasets, demonstrating that SHAPC-Mean and SHAPC-Var address the deficiencies of traditional metrics in assessing decision attribution stability for CL

*Authors contributed equally.

†Corresponding author.

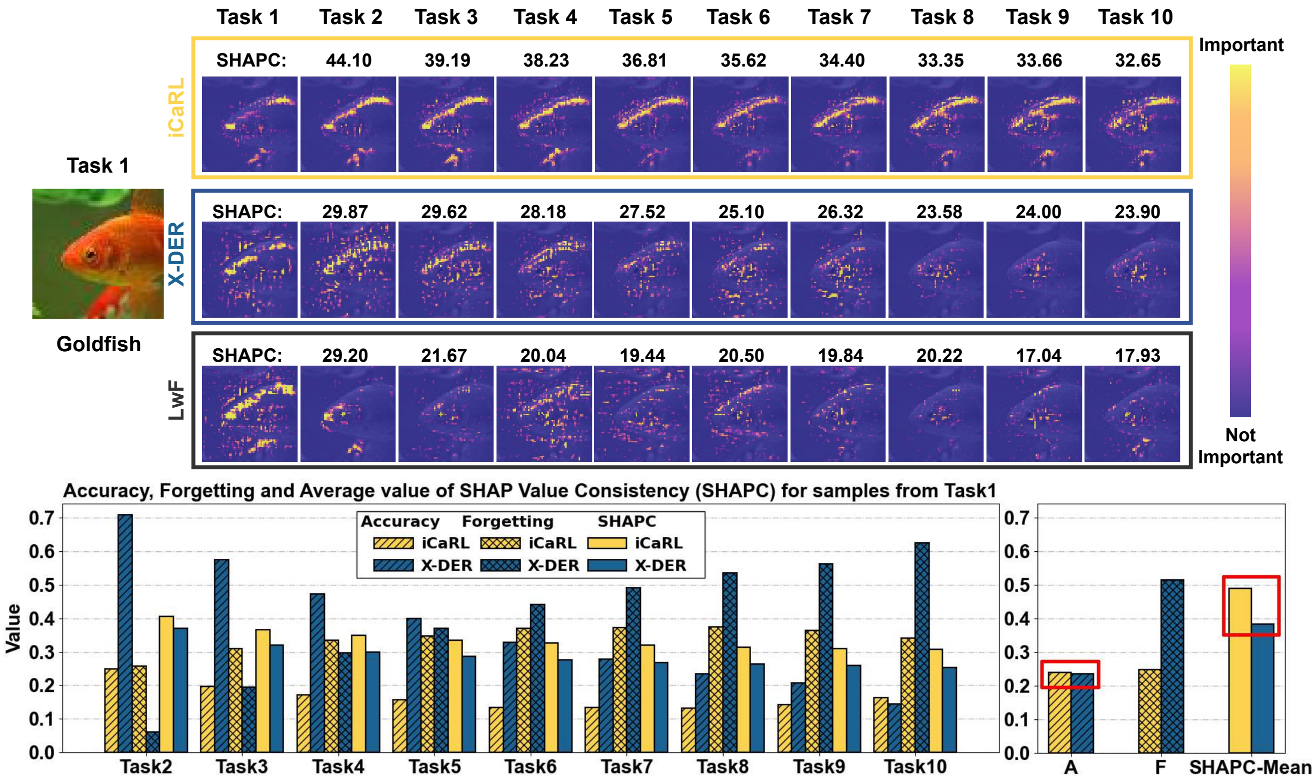

Figure 1. The decision attribution deviation. **The upper part** visualizes the decision attribution of three CL methods: iCaRL, X-DER, and LwF for Split TinyImageNet. The test sample is sampled from the test set of Task 1. The bright areas correspond to the larger SHAP value area. Three methods result in varying degrees of decision attribution deviation. **The lower left part** depicts the trends of the models' accuracy, forgetting, and SHAPC for samples from task 1 throughout the continual learning process. **The lower right part** displays the disparities among traditional continual learning evaluation metrics, including average accuracy (A) and average forgetting (F), alongside the proposed metric SHAPC-Mean outlined in Section 4.2.

methods. For instance, while iCaRL and BFP exhibit relatively similar average accuracy and forgetting rates on Split CIFAR-100, they have substantial disparities in SHAPC-Mean and SHAPC-Var.

2. Related Work

Evaluation Metrics. Most commonly used CL evaluation metrics evaluate a model's overall performance based on average accuracy [5] and average forgetting [5]. The average accuracy is calculated by considering the model's classification accuracy for both the previous and current tasks, assessed after the model has completed training on the latter. Besides, task forgetting is assessed by calculating the variance between its maximum past performance and its current performance, and the average forgetting is computed as the mean of these differences [5]. Backward transfer (BWT) [21] evaluates the performance changes of all previous tasks after learning the current task. Both average forgetting and BWT are utilized to assess the memory stability of a model. In addition to assessing stability, researchers also introduced the intransigence measure(IM) [5] and forward transfer (FWT) [21] to evaluate learning plasticity. IM quantifies a model's inability to learn new tasks, determined by the difference in task performance between joint training and continual learning phases. In contrast, FWT assesses the collective impact of all previous tasks on the current task. It is noteworthy that the aforementioned evaluation metrics are typically applied to continual learners upon completion of the training phase and do not inherently assess the training process itself. De Lange *et al.* [17] observed a phenomenon termed the "stability gap" during the continual training process, occurring specifically during task switching. Building upon this discovery, they formulated four evaluation metrics for CL: average minimum accuracy, worst-case accuracy, windowed forgetting, and windowed plasticity. These metrics serve to assess the stability of continual learners amidst task switching.

Interpretability. While current evaluation metrics enhance the assessment of continual learning (CL), they predominantly emphasize model accuracy performance and overlook interpretability measurements. Interpretability methods currently exist in two main categories: intrinsic and post hoc methods [10]. Intrinsic interpretability involves models that are inherently interpretable, achieved through the imposition of constraints on the model. The commonly used intrinsic interpretability methods include the following: Classification and Regression Trees(CART) [2], Support Vector Machine(SVM) [29], K-Nearest Neighbors(KNN). On the other hand, post hoc interpretability involves methods applied after model training, providing explanations beyond the model's inherent interpretability. Established post-hoc interpretability methods, including Local Interpretable Model-Agnostic Explanations (LIME) [26], SHapley Additive exPlanations (SHAP) [22], and class activation mapping (CAM) [34], offer comprehensive explanations of model behavior. Following the selection of suitable evaluation criteria, post-hoc interpretable methods can effectively facilitate the evaluation of model interpretability. For instance, Zhang *et al.* [33] assesses model interpretability by utilizing human annotations as ground truth and comparing the consistency between the model's decision attribution and the ground truth. Therefore, this paper focuses on examining the attribution of model decisions through post-hoc interpretability methods.

Despite numerous interpretability assessment methods can evaluate the interpretability of model decision behavior, there is currently no method available to quantify the degrees of change in decision attribution in continual learning scenarios. This study aims to address this deficiency by introducing assessment metrics designed to measure the decision attribution stability in continual learning scenarios.

3. Preliminaries

3.1. Continual Learning

In CL, a continual learner is trained on an ordered set of tasks $\{1, ..., T\}$, each task contains a different dataset $\mathcal{D}_t = \{\mathcal{X}_t, \mathcal{Y}_t\}$, which contains the samples $\mathcal{X}_t$, as well labels $\mathcal{Y}_t$. For the task t, the dataset $\mathcal{D}_t$ is presented to the model f, while the previous data does not. The objective function for learning the current parameter $\mathbf{\Theta}_t$ is

$$\mathcal{L}_t = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}[\ell(f(\mathbf{x};\mathbf{\Theta}_t), y)], \tag{1}$$

where ℓ typically denotes the cross-entropy loss. In Class-Incremental Learning (Class-IL), unknowing task-specific classes in the test phase makes the learning process more challenging.

3.2. Shapley Values

Shapley value [27] is a widely recognized and equitable allocation method commonly used in cooperative game theory,

$$s_k(\nu) = \sum_{\mathcal{S}\subseteq\mathcal{N}\setminus\{k\}} \frac{(|\mathcal{N}| - |\mathcal{S}| - 1)!|\mathcal{S}|!}{|\mathcal{N}|!}(\nu(\mathcal{S}\cup\{k\}) - \nu(\mathcal{S})). \tag{2}$$

This formula represents the extent of the contribution of player k in team collaboration. Letting $\mathcal{N}$ represent the contributing players, ν represent the value function, and $\mathcal{S}$ represent the possible combination of remaining players after removing player k, *ie.* $\mathcal{N} \setminus \{k\}$, the allocation of contributions to player k can be calculated, which is also known as the Shapley value $s_k(\nu)$. The *higher* this value, the *greater* the contribution of player k.

In interpretable machine learning, when using Shapley values, the model f is treated as a value function, and each feature $x_k, \forall k \in \mathcal{N}$ in its input $\mathbf{x}$ is considered as a player, where $\mathcal{N} = \{1, 2, \cdots, K\}$ represents the index set of K input features. Consequently, Shapley values can be used to interpret the model's prediction by assigning a value denoting importance to each input feature to indicate its impact on the model's final prediction. In this scenario, Shapley values can be expressed as:

$$s_k(f, \mathbf{x}) = \sum_{\mathcal{S}\subseteq\mathcal{N}\setminus\{k\}} \frac{(|\mathcal{N}| - |\mathcal{S}| - 1)!|\mathcal{S}|!}{|\mathcal{N}|!} \left[f^{\mathcal{S}\cup\{k\}}\left(\mathbf{x}^{\mathcal{S}\cup\{k\}}\right) - f^{\mathcal{S}}\left(\mathbf{x}^{\mathcal{S}}\right)\right], \tag{3}$$

where $\mathbf{x}^{\mathcal{S}}$ represents the sub-vector of $\mathbf{x}$ with the feature subset $\mathcal{S}$ as its index [14], and $f^{\mathcal{S}}$ is the model trained using the sub-vector $\mathbf{x}^{\mathcal{S}}$; the definitions of the sub-vector $\mathbf{x}^{\mathcal{S}\cup\{k\}}$ and the model $f^{\mathcal{S}\cup\{k\}}$ are consistent. In order to compute Shapley values, it is necessary to figure out how to calculate $f^{\mathcal{S}}(\mathbf{x}^{\mathcal{S}})$.

3.3. SHAP Values

Recalling Sec. 3.2, in order to calculate the Shapley values, we need to figure out how to compute $f^{\mathcal{S}}(\mathbf{x}^{\mathcal{S}})$. The vector $\mathbf{x}^{\mathcal{S}}$ is a sub-vector of $\mathbf{x}$, which means that some elements in $\mathbf{x}^{\mathcal{S}}$ are excluded, and most models cannot handle inputs with arbitrary missing values. SHAP values [22] address the issue of missing inputs by treating an input element as ”absent” by assigning a specific value to the element during the calculation.

Additionally, as indicated by Eq. (3), calculating Shapley values involve iterating over all possible subsets $\mathcal{S}$, requiring extensive computation. Expected Gradients (EG) [11], an extension of the Integrated Gradients(IG) [28] method, can approximate SHAP values and enhance the efficiency of SHAP by averaging attribution results estimated by the IG method over multiple baseline points. Consequently, to mitigate computational complexity, we employ EG for SHAP value approximation, adhering to the guidelines provided by the SHAP package [1].

4. Decision Attribution Stability Evaluation

In this section, we present the mathematical definitions for SHAP Value Consistency, the mean of SHAPC, and the variance of SHAPC.

[1] https://github.com/slundberg/shap

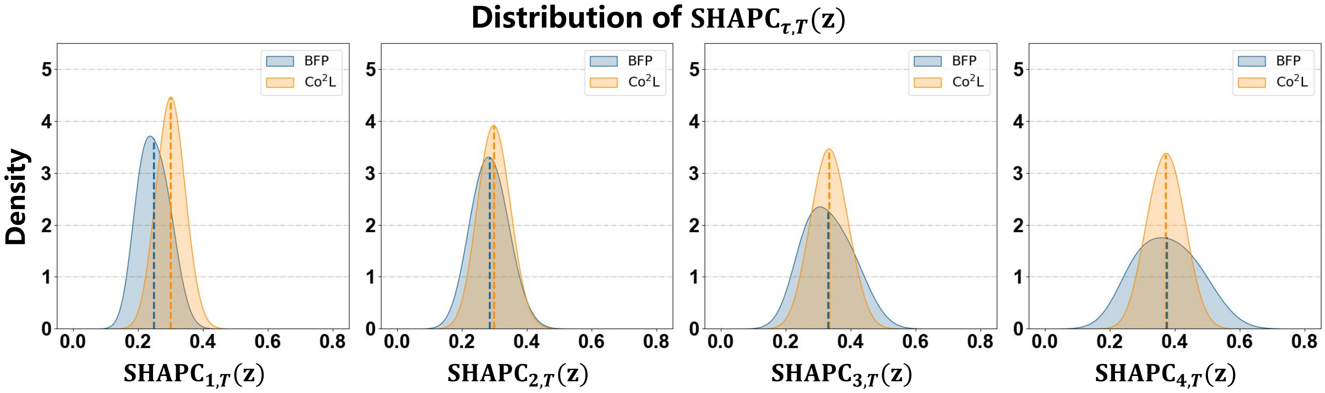

Figure 2. Distribution of SHAPC value on Task t and the final Task T for Split CIFAR-10. $\Pi_{\tau,T}$ for Co^{2}L and BFP are depicted by dashed lines. The values of $\Pi_{\tau,T}$ for Co^{2}L and BFP are highly comparable, indicating that these two methods are likely to exhibit closely aligned results in terms of the SHAPC-Mean metric, while a noticeable disparity exists in the SHAPC distribution (*i.e.* variance) between the two methods.

4.1. SHAP Value Consistency (SHAPC)

To quantify the decision attribution stability of a continual learner, we first define a novel metric termed **SHAP Value Consistency (SHAPC)** for each sample, which measures the consistency of the learner's decision attributions when the task switches. This metric relies on calculating the SHAP value for each feature element of the sample as an indication of the importance of that feature element to the prediction, thereby obtaining the learner's decision attribution for that sample.

The calculation of SHAPC is defined by Eq. (4). Specifically, given a sample $\mathbf{x}$, the term $s_{i,j}\left(f_t,\ \mathbf{x}\right)$ represents the SHAP value of each image pixel positioned at location (i,j) on the input image after the model has been trained on the t-th task, denoted as f_t. Here, differing from Eq. (3), we use (i,j) to index feature. We perform Min-Max Normalization on $s_{i,j}\left(f_t,\ \mathbf{x}\right)$, aiming to mitigate the order of magnitude discrepancies arising from diverse CL methods when computing SHAP values for the same sample. For the sample $\mathbf{x} \in \mathcal{X}_\tau$, where $\mathcal{X}_\tau$ represents the set of samples on task τ, we use $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ to denote the single channel SHAPC in decision-making across the τ-th task and t-th task ($t > \tau$). For multi-channel input image, the result is averaged across the channel dimension.

$$\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right) = \frac{\sum_{i,j \in \mathbf{p}_t(\mathbf{x}) \cap \mathbf{p}_\tau(\mathbf{x})} e^{-\left|s_{i,j}\left(f_t,\ \mathbf{x}\right) - s_{i,j}\left(f_\tau,\ \mathbf{x}\right)\right|}}{\sum_{i,j \in \mathbf{p}_t(\mathbf{x}) \cup \mathbf{p}_\tau(\mathbf{x})} e^{-\left|s_{i,j}\left(f_t,\ \mathbf{x}\right) - s_{i,j}\left(f_\tau,\ \mathbf{x}\right)\right|}}, \tag{4}$$

$\mathbf{p}_t\left(\mathbf{x}\right)$ denotes the important feature area of $\mathbf{x}$ on the t-th task, that is, based on features from which region the model predicts. In this context, the important feature region is the mask whose SHAP value exceeds a certain threshold. specifically, $\mathbf{p}_t\left(\mathbf{x}\right)$ can be expressed as:

$$p_t^{i,j}(\mathbf{x}) = \begin{cases} 1 & \text{if } s_{i,j}(f_t,\mathbf{x}) \geq s_{Th}(f_t,\mathbf{x}), \\ 0 & \text{if } s_{i,j}(f_t,\mathbf{x}) < s_{Th}(f_t,\mathbf{x}). \end{cases} \tag{5}$$

where $s_{Th}\left(f_t,\ \mathbf{x}\right)$ is the threshold of SHAP value.

In essence, for each feature element within the important feature area, $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ calculates the consistency of its SHAP value between task τ and t. Specifically, we calculate SHAP value absolute difference and use the natural exponential function, denoted as $e^{-|\cdot|}$, to restrict SHAPC range to $[0,1]$. $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ quantifies the consistency in the model's decision-making across the τ-th and t-th task. Greater value of $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ correspond to higher decision consistency while smaller $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ value relate to lower decision consistency.

4.2. The Mean of SHAPC

In Section 4.1, we introduce $\text{SHAPC}_{\tau,t}\left(\mathbf{x}\right)$ to denote the SHAP value consistency of the model concerning the decision-making for a sample $\mathbf{x}$ between the previous task τ and the current task t. Upon the completion of the entire continual learning training process, it becomes imperative to assess holistically the performance of decision consistency throughout the entire continual learning process. We define the evaluation metric for this purpose as **SHAPC-Mean**, outlined as follows:

$$\Pi_{\tau,t} = \frac{1}{|\mathcal{X}_\tau|} \sum_{\mathbf{x} \in \mathcal{X}_\tau} \text{SHAPC}_{\tau,t}\left(\mathbf{x}\right), \tag{6}$$

$$\text{SHAPC-Mean} = \frac{1}{T-1} \sum_{\tau=1}^{T-1} \left(\frac{1}{T-\tau} \sum_{t=\tau+1}^{T} \Pi_{\tau,t} \right), \tag{7}$$

$\Pi_{\tau,t}$ in Eq. (6) is the average SHAPC over all samples $\mathbf{x} \in \mathcal{X}_\tau$, where $|\mathcal{X}_\tau|$ is the number of samples. The term SHAPC-Mean in Eq. (7) is the mean of $\Pi_{\tau,t}$ across sequential tasks, capturing the average SHAP value consistency throughout the entire continual learning process. This metric evaluates the stability of decision-making in continual learning. A higher SHAPC-Mean indicates more stable decision attributions.

4.3. The Variance of SHAPC

SHAPC-Mean cannot capture the variation of SHAPC across samples but only calculates the average. Intuitively, a model with high decision attribution stability ought to exhibit not only high SHAPC across tasks but also small variations of SHAPC across samples. Therefore, to characterize SHAPC variation across all samples, we propose the evaluation metric **SHAPC-Var**.

SHAPC-Var is computed by initially defining $\varLambda_{\tau,t}$ as the variance of SHAPC across all samples $\mathbf{x} \in \mathcal{X}_\tau$, post-training on task t,

$$\varLambda_{\tau,t} = \frac{\left[\frac{1}{|\mathcal{X}_\tau|}\sum_{\mathbf{x}\in\mathcal{X}_\tau}\left(\text{SHAPC}_{\tau,t}(\mathbf{x}) - \Pi_{\tau,t}\right)^2\right]^{\frac{1}{2}}}{\Pi_{\tau,t}}, \tag{8}$$

$$\text{SHAPC-Var} = \frac{1}{T-1}\sum_{\tau=1}^{T-1}\left(\frac{1}{T-\tau}\sum_{t=\tau+1}^{T}\varLambda_{\tau,t}\right), \tag{9}$$

SHAPC-Var is the average of $\varLambda_{\tau,t}$ across sequential tasks. A lower value of SHAPC-Var suggests smaller variations of SHAPC across different samples in the given task.

In Fig. 2, we show $\Pi_{\tau,T}$ of different methods, for example, Co^{2}L and BFP. It is evident that even $\Pi_{\tau,T}$ are closely aligned for Co^{2}L and BFP, the distributions of SHAPC differ significantly (in variance). As previously discussed, a lower variance of SHAPC indicates a similar level of decision consistency across different samples, thereby implying a higher decision attribution stability.

5. Experiments

We conduct extensive experiments to validate the proposed SHAPC metric. The quantitative as well as qualitative results are presented in the subsequent sections.

5.1. Experimental Setup

Datasets. In our experiment, we choose class-incremental learning and test our proposed decision attribution stability evaluation metric on three typical datasets: (i) Split CIFAR-10. The Split CIFAR-10 is constructed by splitting the CIFAR-10 dataset [15] into 5 tasks. Each task contains 2 classes, and each of which has $5,000$ and $1,000$ images of size 32×32 for training and testing, respectively. (ii) Split CIFAR-100. The Split CIFAR-100 is constructed by splitting the CIFAR-100 dataset [16] into 10 tasks. Each task contains 10 classes, and each of which has 500 and 100 images of size 32×32 for training and testing, respectively. (iii) Split TinyImageNet. The Split TinyImageNet is constructed by splitting the TinyImageNet dataset [9] into 10 tasks. Each task contains 20 classes, and each of which has 500 and 100 images of size 64×64 for training and testing, respectively.

CL Methods for Evaluation. We conduct an empirical study on nine well-known continual learning methods, including regularization-based methods such as LwF [20] and SI [32], and rehearsal-based methods like iCaRL [25], A-GEM [7], DER [3], DER++ [3], Co^{2}L [4], X-DER [1] and BFP [12]. The hyperparameters for each method follow its original settings.

Traditional CL metrics. In addition to the metric introduced in this paper for measuring model decision attribution stability, we also employ traditional CL evaluation metrics to assess the performance of various CL methods. We choose two commonly used metrics: (1) Average Accuracy (A) [5] and (2) Average Forgetting (F) [6]. Average accuracy represents the final accuracy averaged over all tasks with respect to all past classes. Let $a_{t,\tau}$ denote the model's accuracy on task τ after learning task t. Then, the average accuracy is defined by: $\mathbf{A} = \frac{1}{T}\sum_{\tau=1}^{T} a_{T,\tau}$; which quantifies the average drop in task performance over all tasks. Besides, average forgetting is defined by: $\mathbf{F} = \frac{1}{T-1}\sum_{\tau=1}^{T-1}\max_{i\in\{1,\ldots,T-1\}}\left(a_{i,\tau} - a_{T,\tau}\right)$.

Implementation Details. On all three mentioned datasets, we employ Resnet-18[13] with a linear layer for classification. Our investigation of rehearsal-based methods involves using a fixed buffer capacity of 500 samples. We empirically select feature points within the top 30% of SHAP values to identify the important feature regions, as setting a higher threshold may lead to tiny important feature regions, while a lower threshold may inevitably introduce less significant features to important regions.

5.2. Test on SHAP Value Consistency

We validate SHAP value consistency over various CL methods, and combine this metric with average accuracy and average forgetting to comprehensively assess different CL methods.

In Table 1, we report SHAPC-mean, SHAPC-var, averaging accuracy, and average forgetting of various continual learning methods. From Table 1, it is observed that iCaRL demonstrates a higher SHAPC-Mean and a reduced SHAPC-Var across all the three datasets. The underlying cause of this phenomenon can be attributed to iCaRL's strategy of selecting and retaining samples whose features are closest to the class prototype, combined with the use of distillation loss for replay, thereby maintaining the model's representational stability for the samples. Therefore, iCaRL exhibits better decision attribution stability on samples from previous tasks. Despite this, iCaRL does not achieve the highest average accuracy among all methods. This could be due to the fact that iCaRL exhibits relatively low accuracy in the current task, and its decision-making stability results in consistently low accuracy throughout the continual learning process in this specific task (as shown in the lower left part of Fig. 1), ultimately leading to a lower average accuracy

Method	Pub.Venue	Split CIFAR-10				Split CIFAR-100				Split TinyImageNet			
		$A(\uparrow)$	$F(\downarrow)$	SHAPC		$A(\uparrow)$	$F(\downarrow)$	SHAPC		$A(\uparrow)$	$F(\downarrow)$	SHAPC	
				Mean(↑)	Var(↓)			Mean(↑)	Var(↓)			Mean(↑)	Var(↓)
LwF[20]	TPAMI 2017	19.61	96.16	25.08	11.99	9.96	89.04	28.46	10.85	7.89	76.22	26.41	9.12
SI‡[32]	ICML 2017	19.45	93.03	23.03	13.52	9.28	89.14	28.81	10.58	6.68	62.80	33.50	7.47
iCaRL[25]	CVPR 2017	63.24	26.71	**45.29**	**11.18**	46.55	**29.58**	**43.26**	**9.52**	23.94	**24.79**	**43.99**	**7.90**
A-GEM‡[7]	ICLR 2019	20.64	94.16	26.92	10.58	9.36	89.11	28.31	10.29	7.91	76.56	25.23	7.72
DER[3]	NeurIPS 2020	71.54	27.93	31.26	12.37	36.64	54.19	34.06	10.57	16.41	66.17	31.72	8.03
DER++[3]	NeurIPS 2020	73.36	23.42	31.18	12.68	39.63	52.18	33.23	10.28	18.73	59.63	32.72	8.29
Co^{2}L[4]	ICCV 2021	73.93	25.13	33.63	13.17	33.43	47.07	39.33	11.63	18.86	51.43	34.65	9.23
X-DER[1]	TPAMI 2022	67.02	15.93	34.37	10.62	46.19	25.35	38.08	10.19	23.60	51.38	35.18	8.16
BFP‡[12]	CVPR 2023	**77.14**	**12.66**	31.88	19.03	**47.45**	30.28	39.80	9.80	**25.78**	32.44	38.82	8.55

Table 1. Quantitative results on Split CIFAR-10, Split CIFAR-100 and Split TinyImageNet, respectively. "‡" indicates that certain experimental results of the method are not included in the original publication, and are subsequently derived by us through the replication of the paper's open-source code. **All the results are presented in the form of percentages (%)**.

for iCaRL.

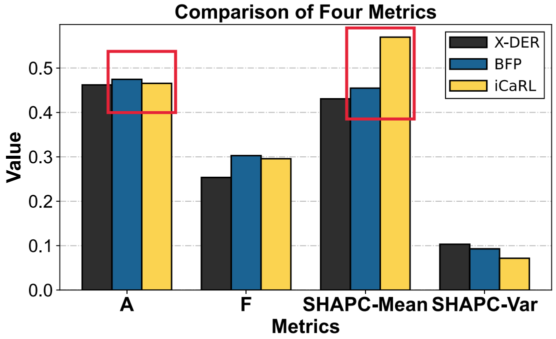

Figure 3. Evaluation of X-DER, BFP, and iCaRL on **Split CIFAR-100** using four evaluation metrics: average accuracy (A), average forgetting (F), SHAPC-Mean, and SHAPC-Var.

Additionally, we discover that BFP sustains the highest accuracy across all three datasets (77.14% on Split CIFAR-10, 47.45% on Split CIFAR-100, and 25.78% on Split TinyImageNet). This is because BFP introduces a linear transformation matrix, relaxing the constraints on features, thereby increasing its plasticity on current tasks at the expense of representational stability, which affects its decision attribution stability on previous tasks, resulting in lower SHAPC-Mean.

Furthermore, it can be observed that while some methods exhibit *similar* results in SHAPC-Mean, they demonstrate significant *differences* in SHAPC-Var. For instance, on Split CIFAR-10, both DER++ and BFP exhibit SHAPC-Mean values of approximately 31%, whereas in SHAPC-Var, DER++ is about 6% lower than BFP. This suggests that SHAPC-Var can be utilized to further characterize the differences in decision consistency across different samples, thereby addressing the limitation of SHAPC-Mean in evaluating consistency at the sample level. Therefore, the evaluation metrics introduced in this study are both necessary and complementary.

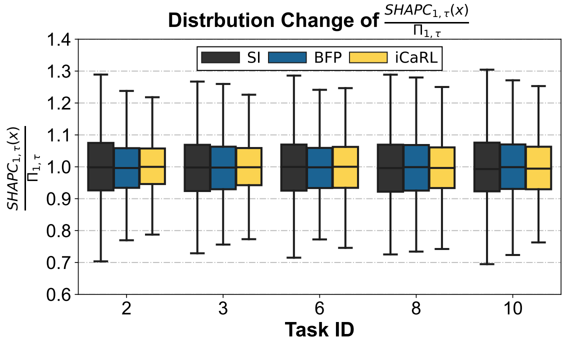

Figure 4. Evaluating the normalized distribution of SHAPC$_{1,\tau}(\mathbf{x})$ during the training of **Split CIFAR-100** for three different CL methods. The box represents the interquartile range (IQR), with the median marked by the central line, where whiskers extend to 1.5 times the IQR.

5.3. Discussion on SHAP Value Consistency

To further analyze the differences in SHAPC across various CL methods and to validate the metrics proposed in this paper, we formulate the below discussion in terms of two questions:

Do the traditional and proposed metrics exhibit identical trend? As depicted in Fig. 3, four evaluation metrics are illustrated through a bar graph, and it is worth noting that BFP's accuracy A is comparable to that of iCaRL. However, the difference in SHAPC-Mean between the two methods is quite pronounced, with iCaRL significantly outperforming BFP, as indicated by the red box. This demonstrates that the evaluation metrics introduced in this paper contribute a novel aspect to the assessment of continual learning methods, implying that when traditional metrics, such as accuracy, are comparable between two methods, the one with superior decision attribution stability ought to be favored.

Does the distribution width of SHAPC values vary across tasks? As illustrated in Fig. 4, we employ a box

Threshold	10%		30%		50%	
Method	SHAPC-Mean(↑)	SHAPC-Var(↓)	SHAPC-Mean(↑)	SHAPC-Var(↓)	SHAPC-Mean(↑)	SHAPC-Var(↓)
LwF[20]	18.69	23.08	28.46	10.85	41.55	6.03
iCaRL[25]	**34.94**	**16.76**	**43.26**	**9.52**	**54.47**	**5.80**
BFP[12]	31.01	18.21	39.80	9.80	51.57	5.89

Table 2. SHAPC-Mean and SHAPC-Var for three CL methods on Split CIFAR-100 with different threshold settings

plot to depict the changes in the distribution of SHAPC during the continual learning process. It can be observed that iCaRL consistently maintains a relatively small interquartile range (IQR), indicating that its SHAPC on task 1 exhibits a narrower distribution throughout the continual learning process. In contrast, SI consistently exhibits a relatively wider distribution. This suggests that, for different CL methods, the relative width of SHAPC value distribution is consistently close throughout the entire continual learning process. Furthermore, the trend of this relative width aligns with the SHAPC-Var results. This demonstrates that SHAPC-Var, by averaging $\Lambda_{\tau,t}$ across sequential tasks, is a reasonable and effective metric for measuring the variations of SHAPC across different samples for the given tasks.

Does SHAPC exhibit sensitivity to the setting of the hyperparameter (specifically, the threshold of the SHAP value)? Table 2 showcase the results of evaluating three continual learning methods using the metrics proposed in this study across various SHAP value thresholds (*top* 10%, *top* 30%, *top* 50%). The results indicate that SHAPC-Mean and SHAPC-Var of the three methods vary with changes in the threshold. Neverthelss, we observe that the relative magnitude of the three methods on the proposed metrics remain consistent, with iCaRL consistently outperforming BFP and LwF. This demonstrates that our metrics can capture differences between methods across different threshold settings, indicating that they are not sensitive to the threshold setting. Furthermore, it is noted that the discrepancy between the three methods in terms of SHAPC-Mean is most pronounced when the threshold is set to 30%. Therefore, we adopt a threshold of 30% for computing the metrics presented in this paper.

6. Conclusions

To evaluate the decision attribution stability of CL methods, this work proposes a new metric—SHAP value consistency (SHAPC) in sequential decision-making. The new metric is based on decision attribution theory, *i.e.* Shapley value theory. Specifically, we calculate the SHAP value for each feature element to quantify decision behavior for model prediction and use SHAP value consistency in sequential decisions to measure the decision attribution stability of continual learners. The higher SHAP value consistency and lower variation across samples indicate better decision attribution stability. Extensive experimental results demonstrate the necessity and validity of the proposed metrics.

7. Acknowledgement

This work is supported by the National Key R&D Program of China (2021ZD0112001) and the National Natural Science Foundation of China (No. 62171111).

References

[1] Matteo Boschini, Lorenzo Bonicelli, Pietro Buzzega, Angelo Porrello, and Simone Calderara. Class-incremental continual learning into the extended der-verse. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:5497–5512, 2022. 1, 5, 6

[2] Leo Breiman. *Classification and regression trees*. Routledge, 2017. 3

[3] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 1, 5, 6

[4] Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co2l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9516–9525, 2021. 1, 5, 6

[5] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision*, pages 532–547, 2018. 1, 2, 5

[6] Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *European Conference on Computer Vision*, 2018. 5

[7] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *International Conference on Learning Representations*, 2018. 1, 5, 6

[8] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021. 1

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image

database. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[10] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *stat*, 1050:2, 2017. 2

[11] Gabriel Erion, Joseph D Janizek, Pascal Sturmfels, Scott M Lundberg, and Su-In Lee. Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature machine intelligence*, 3(7):620–631, 2021. 3

[12] Qiao Gu, Dongsub Shim, and Florian Shkurti. Preserving linear separability in continual learning by backward feature projection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 24286–24295, 2023. 5, 6, 7

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[14] Gwladys Kelodjou, Laurence Rozé, Véronique Masson, Luis Galárraga, Romaric Gaudel, Maurice Tchuente, and Alexandre Termier. Shaping up shap: Enhancing stability through layer-wise neighbor selection. *arXiv preprint arXiv:2312.12115*, 2023. 3

[15] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 5

[16] Abhishek Kumar and Hal Daumé III. Learning task grouping and overlap in multi-task learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1723–1730, 2012. 5

[17] Matthias De Lange, Gido M van de Ven, and Tinne Tuytelaars. Continual evaluation for lifelong learning: Identifying the stability gap. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[18] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *International Conference on Machine Learning*, pages 3925–3934, 2019. 1

[19] Xuhong Li, Haoyi Xiong, Xingjian Li, Xuanyu Wu, Xiao Zhang, Ji Liu, Jiang Bian, and Dejing Dou. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. *Knowledge and Information Systems*, 64 (12):3197–3234, 2022. 1

[20] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 1, 5, 6, 7

[21] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 1, 2

[22] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 1, 3

[23] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, pages 109–165. 1989. 1

[24] Dushyant Rao, Francesco Visin, Andrei Rusu, Razvan Pascanu, Yee Whye Teh, and Raia Hadsell. Continual unsupervised representation learning. *Advances in neural information processing systems*, 32, 2019. 1

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 5, 6, 7

[26] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016. 3

[27] LS Shapley. 17. a value for n-person games. In *Contributions to the Theory of Games (AM-28), Volume II*, pages 307–318. Princeton University Press, 2016. 3

[28] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328, 2017. 3

[29] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013. 3

[30] Kush R Varshney, Prashant Khanduri, Pranay Sharma, Shan Zhang, and Pramod K Varshney. Why interpretability in machine learning? an answer using distributed detection and data fusion theory. *arXiv preprint arXiv:1806.09710*, 2018. 1

[31] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*, 2023. 1

[32] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995, 2017. 1, 5, 6

[33] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018. 3

[34] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 3