PHICO: PERSONALISED HUMAN-AI COOPERATIVE CLASSIFICATION USING AUGMENTED NOISY LABELS AND MODEL PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The nuanced differences in human behavior and the complex dynamics of human-AI interactions pose significant challenges in optimizing human-AI cooperation. Existing approaches tend to oversimplify the problem and rely on a single global behavior model, which overlooks individual variability, leading to sub-optimal solutions. To bridge this gap, we introduce PHICO, a novel framework for human-AI cooperative classification that initially identifies a set of representative annotator profiles characterized by unique noisy label patterns. These patterns are then augmented to train personalised AI cooperative models, each tailored to an annotator profile. When these models are paired with human inputs that exhibit similar noise patterns from a corresponding profile, they consistently achieve a joint classification accuracy that exceeds those achieved by either AI or humans alone. We theoretically prove the convergence of PHICO, ensuring the reliability of the framework. To evaluate PHICO, we introduce novel measures for assessing human-AI cooperative classification and empirically demonstrate its generalisability and performance across diverse datasets including CIFAR-10N, CIFAR-10H, Fashion-MNIST-H, AgNews, and Chaoyang histopathology. PHICO is both a model-agnostic and effective solution for improving human-AI cooperation.

1 INTRODUCTION

Determining the optimal human-AI cooperation mechanism is challenging (Dafoe et al., 2021). Humans bring experience and contextual insights but are prone to biases; machine learning models excel in specific tasks but lack contextual understanding and complex reasoning (Holstein & Aleven, 2021). Many human-AI joint decision making strategies were proposed, e.g., learning to defer (Raghu et al., 2019; Madras et al., 2018; Mozannar et al., 2023), learning to complement (Wilder et al., 2021), human-in-the-loop (Wu et al., 2022), and algorithm-in-the-loop (Green & Chen, 2019), seeking to blend the best of human and AI for optimal decision-making.

We argue that effective human-AI joint decision-making depends on personalising machine learning (ML) models to the individual's behaviour patterns. While recent works have shown promising progress in incorporating human behaviours through behaviour models (Vodrahalli et al., 2022) or confusion matrices (Kerrigan et al., 2021), they rely on single global matrix and could not account for the varied biases and preferences among annotators (Kocielnik et al., 2019; Wang et al., 2021).

Indeed, learning individual behavior patterns is challenging, as each person's data usually represents only a small portion of the total, making it insufficient to train personalised AI models Johnson et al. (2021). Beyond the scarcity of individual data, evaluating the effectiveness of various human-AI cooperation frameworks also poses difficulties. Traditional metrics such as accuracy fail to capture whether the ML model's alteration to human inputs improve or degrade performance, further complicating the assessment of cooperation effectiveness Shneiderman (2022).

This paper addresses these research gaps with PHICO, a framework designed for personalised human-AI cooperative classification to achieve optimal performance (Figure 1). More specifically, given a training dataset with noisy labels from multiple annotators, PHICO first identifies a set of annotator profiles, each characterized by distinct noisy labeling patterns. PHICO then augments these identified noisy label patterns to train personalised AI cooperative model, each optimized to

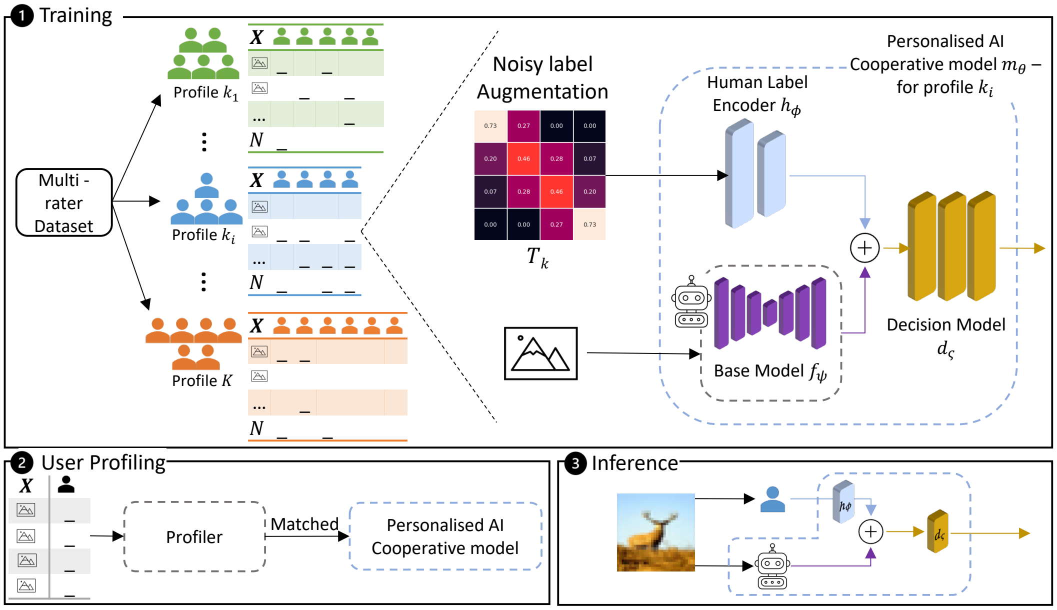

Figure 1: Training and inference of PHICO.

effectively interact with its corresponding annotator profile. During testing, a new user undergoes a *user profiling* process, after which a suitable personalised AI cooperative model is selected for personalised human-AI cooperative classification.

We present both a theoretical proof of convergence and an empirical evaluation of PHICO, and introduce a novel assessment measure, *alteration rate*, which quantifies how the model positively or negatively alters labels from human and AI sources. Our empirical studies include both simulated and real multi-rater datasets across various modalities (images and texts) and domains (daily objects, news, and medical), including CIFAR-10N, CIFAR-10H, Fashion-MNIST-H, AgNews, and Chaoyang histopathology. The results show that PHICO is a model-agnostic human-AI cooperation framework outperforming both AI and human decisions alone, as well as state-of-the-art human-AI cooperation methods across various classification tasks. To summarise, our contributions are:

- The first human-AI cooperation framework that combines noisy label learning methods and personalised AI cooperative model.
- A new cooperative classification assessment measure, *alteration rate*, to quantify how the model positively or negatively alters labels from human and AI sources.
- A theoretical proof of convergence and empirical results demonstrating state-of-the-art performance across diverse datasets, including CIFAR-10N, CIFAR-10H, Fashion-MNIST-H, AgNews, and Chaoyang histopathology.

PHICO is model-agnostic and can be trained effectively with noisy labels from multiple raters without ground truth labels, making it a valuable and practical contribution to the ML community.

2 Related Work

The conventional belief that automation lessens human control is under revision (Parasuraman et al., 2000; Committee, 2014), as the uncertainties of automation often demand more human involvement, leading to new human-AI collaboration strategies (Strauch, 2018). With AI models exceeding human accuracy in certain tasks, three new human-AI collaboration paradigms have emerged:

Learning-to-assist approaches aim to support human decision-making with AI model predictions (Straitouri et al., 2023). These approaches are commonly seen in critical domains, such as law (Liu et al., 2021) and medicine (Levy et al., 2021), where humans make the final decision. Considerable work has been done to improve model explainability and transparency. (Tjoa & Guan, 2021).

Learning-to-defer methods allow AI models to autonomously manage confident cases and defer decisions to humans when confidence is low (Madras et al., 2018; Mozannar et al., 2023; Alves et al., 2023). These approaches focus on the optimization of a utility function that takes into account the accuracy of the AI model, the preference of the human decision maker, and the cost of deferring decisions. For example, Raghu et al. (2019) used an ensemble of AI models to predict the risk of patient death, and then defers decisions to a human expert for patients with the highest risk.

Learning-to-complement models are optimized to leverage the strengths from both human and AI model to improve decision-making. For example, Steyvers et al. (2022) proposed a Bayesian framework for modeling human-AI complementarity. Kerrigan et al. (2021) used a calibrated confusion matrix to combine human and model predictions in a way that minimizes the expected loss. Wilder et al. (2021) consider the uncertainty from AI models and humans to jointly train a model that allocates tasks to the AI model or the human to maximize the overall accuracy.

PHICO falls into the category of learning-to-complement and aims to utilise complementary strengths of both humans and AI. Unlike other approaches that rely on a single behavior model or a global confusion matrix for the entire dataset, PHICO takes a step further by identifying biases among annotators and personalizing the human-AI cooperation to account for these unique biases.

2.1 Evaluating Human-AI Cooperation

Human-AI complementarity is defined by Dellermann et al. (2021) as leveraging the unique capabilities of both humans and AI to achieve better results than each one could achieve alone. However, assessing the interaction between humans and AI is complicated, and numerous benchmarks have been suggested in existing literature. In the context of learning-to-assist or learning-to-complement, traditional measures such as *accuracy*, *precision*, and *recall* are commonly used. For learning-to-defer, measures such as *coverage* are proposed to evaluate the proportion of the data that is processed by the model alone (Raghu et al., 2019). When dealing with noisy labels, additional measurements such as *label precision*, *label recall*, and *correction error* are also used (Song et al., 2022a). As PHICO presents a new paradigm that combines decisions from humans and AI, we introduce new assessment measures to understand whether combination leads to better decisions.

2.2 Learning from Noisy-label (LNL) and Multi-rater Learning (MRL)

PHICO draws insights from the LNL and MRL community. LNL aims to design algorithms that are robust to the presence of noisy training labels. Recent advancements include DivideMix (Li et al., 2020) with its semi-supervised approach, ELR (Liu et al., 2020) exploring early learning phenomena, C2D (Zheltonozhskii et al., 2022) tackling the warm-up obstacle, and UNICON (Karim et al., 2022) with a unified supervised and unsupervised learning to handle noisy labels effectively. MRL trains models using noisy labels from multiple annotators per sample, which can mitigate the identifiability problem under certain conditions (Liu et al., 2023). Key developments include MRNet (Ji et al., 2021), which addresses multi-rater disagreement, and Crowdlab (Goh et al., 2023), designed to be model-agnostic. Despite improvements from LNL and MRL, an accuracy gap persists compared to training with clean labels. This has led to our personalized human-AI joint decision-making paradigm, which incorporates inputs from both humans and AI to make decisions.

3 Methodology

PHICO is a model-agnostic human-AI cooperation framework designed to enhance the performance of human-AI joint decision making. In the following sub-sections, we first define the dataset notations in Section 3.1, explain the training process and convergence proof in Section 3.2, and outline the profiling and inference stages in Section 3.3. Section 3.4 presents our proposed metrics for assessing personalised human-AI cooperation.

3.1 DATASET NOTATION

Let a multi-rater training set for a multi-class classification task be $\tilde{\mathcal{D}} = \{(\mathbf{x}_i, \{\tilde{\mathbf{y}}_{i,j}\}_{j\in\mathcal{A}})\}_{i=1}^N$, where $\mathbf{x}_i \in \mathcal{X}$ is a data sample, $\tilde{\mathbf{y}}_{i,j} \in \mathcal{Y} \subset \{0,1\}^C$ is a one-hot vector for the C-class classification, representing the noisy-label provided by annotator $j \in \mathcal{A}$. We assume that each data sample has a latent clean label denoted by $\mathbf{y}_i \in \mathcal{Y}$, annotators' label noise is class-dependent (or asymmetric) (Song et al., 2022b), and a consensus labelled training set denoted by $\bar{\mathcal{D}} = \{(\mathbf{x}_i, \bar{\mathbf{y}}_i)\}_{i=1}^N$. Note that a key challenge in most human-AI cooperation approaches is their dependence on ground truth labels, which are often hard to obtain. PHICO tackles this problem by using consensus labels, generated through methods like majority voting or expectation maximization (Sinha et al., 2018; Ji et al., 2021; Warfield et al., 2004), eliminating the need for ground truth. In our case, we utilize Crowdlab (Goh et al., 2023) for its simplicity and superior performance in estimating consensus labels. We provide more details about estimating consensus labels in Appendix A.

3.2 TRAINING OF PERSONALISED HUMAN-AI COOPERATIVE MODEL

Figure 1 shows the three steps for training PHICO: 1) identifying annotator profiles with distinct noisy-label patterns, 2) augmenting noisy labels for each profile, and 3) training personalized AI cooperative models using the augmented noisy labels. We explain each step below.

Identifying annotator profiles: To identify a set of representative profiles, each with a distinct noisy label pattern, we first arrange the label sets from all annotators in a uniform format as equation 1. We take each annotator $j \in \mathcal{A}$ and each class $c \in \{1, ..., C\}$ to build the set of sample labels that have consensus label c, with $\mathcal{S}_j^{(c)} = \{\tilde{\mathbf{y}}_{i,j} | (\mathbf{x}_i, \tilde{\mathbf{y}}_{i,j}) \in \tilde{\mathcal{D}}, c = \arg\max_{\tilde{c}\in\{1,...,C\}} \bar{\mathbf{y}}_i(\tilde{c})\}$. We can then build the $L \times C$ vector,

$$\mathbf{s}_j = [l_1^{(1)}, ..., l_L^{(1)}, ..., l_1^{(C)}, ..., l_L^{(C)}] \tag{1}$$

for annotator $j \in \mathcal{A}$ by randomly selecting L data samples for each class, where $l_l^{(c)} = \arg\max_{\tilde{c}\in\{1,...,C\}} \tilde{\mathbf{y}}_{i,j}(\tilde{c})$ with $\tilde{\mathbf{y}}_{i,j} \in \mathcal{S}_j^{(c)}$ representing one of the noisy labels from $\mathcal{S}_j^{(c)}$. Each $\mathbf{s}_j$ may be different, but class order is preserved for all annotators. This process is repeated for all annotators to form the set $\mathcal{L} = \{\mathbf{s}_j\}_{j\in\mathcal{A}}$. We identify representative annotator profiles within $\mathcal{L}$ based on distinct noisy label patterns (Dehariya et al., 2010), using Fuzzy K-Means for its robustness in handling noisy data (Xu et al., 2016) with the optimal K determined by the silhouette score, which measures clustering quality (Appendix B). Each annotator is then assigned a profile.

Noisy-label augmentation: After identifying a set of K profiles, the original training set $\tilde{\mathcal{D}}$ is divided into K subsets $\tilde{\mathcal{D}}_k \subset \tilde{\mathcal{D}}$, each containing the users allocated to profile $k \in \{1, ..., K\}$. Since the data is divided, some subsets may be missing samples from the original set, as users may not have annotated all samples in $\mathcal{D}$. To address this, we propose a noisy label augmentation process that generates extra labels for each profile, enabling the training of K models. This label augmentation is obtained by sampling from the estimated profile-specific label transition matrix, mapping the consensus label to the noisy label. This approach captures the label biases in each profile, allowing the classifier to be trained to effectively handle these biases.

Assuming profile k from annotator subset $\mathcal{A}_k \subset \mathcal{A}$, k's label transition matrix $\mathbf{T}_k \in [0,1]^{C\times C}$ is:

$$\mathbf{T}_k(c,:) = \frac{1}{|\mathcal{A}_k|} \sum_{\tilde{\mathbf{y}}_i \in \left\{\mathcal{S}_j^{(c)}\right\}_{j\in\mathcal{A}_k}} \tilde{\mathbf{y}}_i, \tag{2}$$

where $\left\{\mathcal{S}_j^{(c)}\right\}_{j\in\mathcal{A}_k}$ denotes the set of labels defined above (from samples with consensus label c, for all users in $\mathcal{A}_k$). Note that each element of the transition matrix for profile k from equation 2 denotes the probability that a user in profile k flips from the consensus label $\bar{Y} = c$ to the noisy label $\tilde{Y} = n$, as in $\mathbf{T}_k(c,n) = p(\tilde{Y} = n | \bar{Y} = c, R = k)$, where R is the random variable for the user profile. For each data point $\mathbf{x}_i$ in $\tilde{\mathcal{D}}_k$, we take its consensus label c from $\bar{\mathcal{D}}$ and the profile k's transition matrix $\mathbf{T}_k$ from equation 2 to generate G labels by sampling $\{\hat{\mathbf{y}}_g\}_{g=1}^G \sim p(\tilde{Y}|\bar{Y} = c, R = k)$, which represents the categorical distribution in row c of the transition matrix $\mathbf{T}_k$. The new noisy-label augmented training set for each profile k is denoted by $\hat{\mathcal{D}}_k = \{(\mathbf{x}, \{\hat{\mathbf{y}}_g\}_{g=1}^G) | (\mathbf{x}, \{\tilde{\mathbf{y}}_j\}_{j=1}^{A_k}) \in \tilde{\mathcal{D}}_k, \{\hat{\mathbf{y}}_g\}_{g=1}^G \sim p(\tilde{Y}|\bar{Y} = c, R = k)\}$.

Training personalised human-AI cooperative model: With the annotator profiles and their augmented noisy labels, we can now formulate the training of the personalised AI cooperative model. The proposed model (top-right of Figure 1) has three components: 1) a base model that transforms input data into a logit with $f_{\psi_k} : \mathcal{X} \rightarrow \mathbb{R}^C$; 2) a human label encoder that takes the one-hot user provided noisy label and transforms it into a logit with $h_{\phi_k} : \mathcal{Y} \rightarrow \mathbb{R}^C$; and 3) a decision model that takes the model's and human's logits to output a categorical distribution with $d_{\zeta_k} : \mathbb{R}^C \times \mathbb{R}^C \rightarrow \Delta^{C-1}$. The base model $f_{\psi_k}(.)$ learns the features of the data, the human label encoder model $h_{\phi_k}(.)$ aims to discover the label biases of user profile k, and $d_{\zeta_k}(.)$ aims to model the joint label noise distribution between the base model and human label encoder to make $m_{\theta_k}(\mathbf{x}, \hat{\mathbf{y}})$ robust to label noise. The whole model $m_{\theta_k} : \mathcal{X} \times \mathcal{Y} \rightarrow \Delta^{C-1}$ is defined as:

$$m_{\theta_k}(\mathbf{x}, \hat{\mathbf{y}}) = d_{\zeta_k}(f_{\psi_k}(\mathbf{x}) \oplus h_{\phi_k}(\hat{\mathbf{y}})), \tag{3}$$

where $\theta_k = \{\psi_k, \phi_k, \zeta_k\}$, and $\oplus$ represents the concatenation operator. The base model $f_{\psi_k}(.)$ could use a different architecture, provided it is trained on $\bar{\mathcal{D}}$. Similarly, $h_{\phi_k}(.)$ and $d_{\zeta_k}(.)$ can be of different architectures; we configured them as a two-layer and three-layer multi-layer perceptron, respectively, with ReLU activations. The model in equation 3 is trained as:

$$\begin{aligned} \{\theta_k^*\}_{k=1}^K = \arg \min_{\{\theta_k\}_{k=1}^K} \frac{1}{K \times |\hat{\mathcal{D}}_k| \times G} \times \sum_{k=1}^K \sum_{\left(\mathbf{x}_i, \{\hat{\mathbf{y}}_{i,g}\}_{g=1}^G\right) \in \hat{\mathcal{D}}_k} & \ell\left(\bar{\mathbf{y}}_i, m_{\theta_k}(\mathbf{x}_i, \hat{\mathbf{y}}_{i,g})\right) + \\ & \lambda \times \ell\left(\hat{\mathbf{y}}_{i,g}, (\mathbf{T}_k)^\top \times m_{\theta_k}(\mathbf{x}_i, \hat{\mathbf{y}}_{i,g})\right), \end{aligned} \tag{4}$$

where $\bar{\mathbf{y}}_i$ is the consensus label from $\bar{\mathcal{D}}$, $\ell(.)$ is the cross-entropy loss, $\lambda \in [0, \infty]$ is a hyper-parameter, and the second loss term is motivated by the forward correction procedure proposed by Patrini et al. (2017), transforming the clean label prediction from $m_{\theta_k}(.)$ into the noisy ones in $\hat{\mathcal{D}}_k$.

Theoretical proof of PHICO convergence: In the Appendix D, we prove the convergence of the key steps PHICO, namely, the Fuzzy K-Means clustering used to identify annotator profiles, the training of the personalized human-AI cooperative models, and the integration of these two steps.

3.3 User Profiling for Personalisation

Once the models are trained, PHICO achieves personalisation during the testing by first matching the new user to one of the learned personalised AI cooperative models, after which they perform human-AI cooperative classification. The matching process, which we name *user profiling*, has two steps: 1) classifying the testing user into one of the K profiles, to enable the matching of the user to its personalized classifier $m_{\theta_k}(.)$ and 2) setting an entry condition based on a comparison between the accuracy of the testing user and the base model $f_{\psi_k}(.)$.

The classifier used in the first step is trained with samples that consist of randomly collected labels of M training samples for each of the C classes (estimated from the consensus labels), from users belonging to each of the K profiles. This forms multiple vectors of size $M \times C$, which have the structure defined in equation 1, where each of those vectors is labelled with the user's profile. We then train a one-versus-all (OVA) support vector machine (SVM) K-class classifier.

To classify a testing user into one of the K profiles, we first ask the user to label each image in a validation set, $\mathcal{V} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{M \times C}$, which contains images not used in the training or testing sets. Using these labels, we build an $M \times C$ vector, which is then processed by the OVA SVM classifier to determine the user's profile.

In the second step, we compare the base model and testing user accuracies on the validation set $\mathcal{V}$. The model $m_{\theta_k}(.)$ is used only if the base model $f_{\psi_k}(.)$ performs better (Steyvers et al., 2022). $m_{\theta_k}(.)$ is evaluated on the test set $\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i)\}$ with no overlap with training or validation images.

3.4 New Metrics for Personalised Human-AI Cooperative Classification

Our new evaluation criteria assesses the impact of the model's label alterations on user performance. We first define the positive and negative alteration measures:

$$\text{Positive Alteration} : A_{+} = \frac{1}{|\mathcal{T}| \times |\mathcal{A}|} \sum_{i=1,j=1}^{|\mathcal{T}|,|\mathcal{A}|} \frac{\ddot{\mathbf{y}}_{i,j} = \bar{\mathbf{y}}_i}{\tilde{\mathbf{y}}_{i,j} \neq \bar{\mathbf{y}}_i}$$
$$\text{Negative Alteration} : A_{-} = \frac{1}{|\mathcal{T}| \times |\mathcal{A}|} \sum_{i=1,j=1}^{|\mathcal{T}|,|\mathcal{A}|} \frac{\ddot{\mathbf{y}}_{i,j} \neq \bar{\mathbf{y}}_i}{\mathbf{y}_{i,j} = \bar{\mathbf{y}}_i} \tag{5}$$

$$\text{Positive Alteration Rate} : R_{A_{+}} = \frac{A_{+}}{A_{+} + A_{-}}$$
$$\text{Negative Alteration Rate} : R_{A_{-}} = \frac{A_{-}}{A_{+} + A_{-}} \tag{6}$$

where $\ddot{\mathbf{y}}_j = \mathsf{OneHot}(m_{\theta_k}(\mathbf{x}, \tilde{\mathbf{y}}_j))$, with the function $\mathsf{OneHot} : \Delta^{C-1} \to \mathcal{Y}$ returning a one-hot label representing the class with the largest prediction from the model $m_{\theta_k}(.)$. In equation 5, A_+ quantifies the effectiveness of the model to correct users' labels, where the user provided incorrect labels. In contrast, A_-, in equation 5, measures the proportion where the user had a correct label that was subsequently misclassified by the model.

Aligning with that, R_{A_+} and R_{A_-} in equation 6 measure positive and negative alteration rates, respectively. Hence, an effective model should have high R_{A_+}, low R_{A_-}, and a high post-alteration accuracy, i.e. the accuracy after the label alteration by the personalised AI cooperative model.

4 Experiments

4.1 Datasets

CIFAR-10 includes 50,000 training, 200 validation, and 9,800 testing class-balanced color images, each sized 32×32, with 10 classes. **CIFAR-10N** extends CIFAR-10's training set via crowd-sourced labeling to 747 annotators, with each image having three labels from different annotators. **CIFAR-10H** expands CIFAR-10's testing set via crowd-sourcing to 2571 annotators, resulting in an average of 51 labels per image. **Fashion-MNIST-H** extends Xiao et al.'s Fashion-MNIST's testing set to multiple annotations from 885 annotators, averaging 66 labels per image. We use the crowd-sourced testing set as the training set, with 200 images from the original training set allocated for validation and the remainder for testing. **AgNews** is a text classification dataset with 120,000 training, 200 validation, and 7,400 testing news articles across 4 classes. Lastly, **Chaoyang** is a pathological dataset with 4021 training, 80 validation, and 2059 testing images, each having three expert labels in the training set. More details about datasets can be found in Appendix C.1.

Setup on datasets with simulated annotators: On CIFAR-10, a pairwise flipping experiment is conducted where 8 out of 10 classes have clean labels, but in two classes, 80% of samples have labels flipped. Three user profiles are simulated by flipping labels between classes airplane↔bird, another profile that flips horse↔deer, and the other flips truck↔automobile. This results in 15 unique users (5 for each profile) for training and testing. For AgNews, pairwise flipping occurs on two out of four classes, with 80% of samples flipped. Three user profiles are simulated, one that flips between classes business↔science/technology, another that flips world↔sports, and the third that flips sports↔business. resulting in 15 unique users (with 5 for each profile) for training and testing. Both datasets use $\tilde{\mathcal{D}}$ for training OVA SVM with automatically chosen K profiles based on silhouette score in equation 8. ResNet-18 He et al. (2016) and Bert-Base-Uncased Devlin et al. (2018) models are used as $f_{\psi_k}(.)$ in training $m_{\theta_k}(.)$ in equation 3 for each profile k with CIFAR-10 and AgNews respectively. More details on the simulation setup is in Appendix C.

Setup on datasets with real annotators: for CIFAR-10N training, we conduct two experiments. In the first experiment, the labels from 747 annotators form $\tilde{\mathcal{D}}$. Of these, 155 annotators who labeled at least 20 images per class are selected, split into 79 training users and 80 testing users. The training users' labels are used to build K profiles where K is automatically chosen based on the silhouette score in equation 8, and train the OVA SVM classifier. During testing, noisy-label transition matrices are estimated using annotator labels and consensus labels for each testing user, resulting in 80 noisy test sets. In the second CIFAR-10N experiment, CIFAR-10H is used as the testing set without modification. Noise transition matrices are estimated and used to simulate noisy annotations for each testing user, resulting in unique noisy test sets for all 2571 users. For Fashion-MNIST-H, labels from 885 annotators form $\tilde{\mathcal{D}}$. 366 annotators who labeled at least 20 images per class are selected, split into 183 training and 183 testing users. Similar to CIFAR-10N, noisy-label transition matrices are estimated for testing users, producing 183 noisy testing sets. Chaoyang dataset has three annotators per image, forming $\tilde{\mathcal{D}}$. Training users are used to build K profiles and train an OVA SVM classifier. During testing, noisy-label transition matrices are estimated, resulting in three noisy test sets. Details on experiment setup, data preparation, and implementation are in Appendix C.

Backbone models and training details: Our experiments use various backbone models to showcase robustness, including ViT-Large-16, DenseNet-121, and ResNet-50. Data augmentation policy by Cubuk et al. (2019) was adopted for CIFAR-10 and Cubuk et al. (2020) for Fashion-MNIST datasets, while Chaoyang is limited to random horizontal and vertical flips. Pre-trained models are employed for their robustness to noisy labels (Jiang et al., 2020). We use Adam and NAdam optimizers to train $f_{\psi_k}(.)$ and $m_{\theta_k}(.)$. Implementation is in PyTorch, running on an NVIDIA RTX 4090 GPU.

4.2 Results

Table 1 displays the post-alteration accuracy, provided by PHICO, with respect to the original accuracy of users, followed by Table 2 that shows positive and negative alteration as computed in equation 5 and alteration rates from equation 6 for K selected from the silhouette score in equation 8. The shaded rows in Table 1 contrast testing users who met the entry condition (see second step in Section 3.3), against all testing users in the unshaded rows (note: for the CIFAR10 simulation, the two sets are the same since all users met the condition). Note that Table 2 shows results for profiled users from the shaded rows of Table 1.

Table 1: Number of users who improved (I), maintained (M) or did not improve (NI) and Initial accuracy vs accuracy after alterations. (Un)shaded rows: users who (do not)meet entry condition.

Dataset	K (Silhouette score)	Users	I	M	NI	Original Accuracy	Post-alt. acc.
With simulated annotators							
CIFAR10	3 (0.55)	15	15	0	0	0.8400	0.8788
		15	15	0	0	0.8400	0.8788
AgNews	3 (0.57)	15	15	0	0	0.5998	0.9802
		15	15	0	0	0.5998	0.9802
With real annotators							
CIFAR10-N	2 (0.01)	80	80	0	0	0.8365	0.9891
		80	80	0	0	0.8365	0.9891
CIFAR10-H	2 (0.01)	2571	2566	1	4	0.9487	0.9930
		2022	2022	0	0	0.9399	0.9926
Fashion-MNIST-H	2 (0.09)	183	183	0	0	0.6723	0.8785
		182	182	0	0	0.6625	0.8779
Chaoyang	3 (0.99)	3	3	0	0	0.9027	0.9466
		2	2	0	0	0.8582	0.9237

Table 2: Positive and negative alterations and rates from on-boarded users of Table 1.

Dataset	K (Silhouette score)	Positive and Negative alterations		Positive and Negative alteration rates	
		A_+	A_-	R_{A_+}	R_{A_-}
With simulated annotators					
CIFAR10	3 (0.55)	0.9437	0.1336	0.8759	0.1240
AgNews	3 (0.57)	0.9748	0.0162	0.9836	0.0164
With real annotators					
CIFAR10-N	2 (0.01)	0.9541	0.0040	0.9958	0.0042
CIFAR10-H	2 (0.01)	0.9389	0.0041	0.9956	0.0044
Fashion-MNIST-H	2 (0.09)	0.7581	0.0731	0.9121	0.0879
Chaoyang	3 (0.99)	0.7377	0.0453	0.9422	0.0578

Results of datasets with simulated annotators: The first and second rows of Table 1 detail the number of testing users that improved (I), maintained (M), or did not improve (NI) with PHICO in the CIFAR-10 and AgNews simulations. The accompanying comparison between original and post-alteration accuracy is reported in the last two columns. Note that in Table 1, all 15 users improved, with the average accuracy after alteration surpassing the original accuracy in both datasets. In Table 2, a large A_+ contrasts with a low A_-, emphasizing a high proportion of R_{A_+} and a low proportion of $R_{A\_}$. Notably, the noise matrices estimated for $K = 3$ in figures 9 and 10 closely resemble those used to simulate 15 users in figures 3 and 2, which confirms the estimated $K = 3$ in Tables 1 and 2.

Results of datasets with real annotators: According to Table 1, all users who were profiled and met entry condition in every experiment, improved their accuracy with PHICO. Even considering all users, the method tends to improve the performance of the majority. Table 1 shows that the accuracy after alterations for profiled users in CIFAR-10N, CIFAR-10H, Fashion-MNIST-H and Chaoyang increase by approximately 18%, 5%, 30%, 7%, respectively. Table 2 shows that PHICO has high positive alteration rates for profiled users compared to negative alteration rates.

Appendix E presents standard deviation and 95% confidence values for post-alteration accuracy at automatically selected K for all datasets, showing a significant improvement in user accuracy in all datasets. Additionally, Table 11 in Appendix F highlights effective joint decision-making, even when both human and base model are incorrect, showcasing the capacity to learn joint biases. A simple attempt to model interpretability is discussed in Appendix G using CIFAR-10 simulation.

Comparison with related methods: In Table 3, we compare our results with the following competing methods proposed in literature: Raghu et al. (2019) which defers to humans when the classifier's error probability is high, Madras et al. (2018) blending human and AI insights, Okati et al. (2021)

Table 3: Comparison of PHICO against proposed methods in literature.

Method	CIFAR-10N	CIFAR-10H	FashionM-H	Chaoyang
	Trained with Ground Truth			
Madras et al. (2018)	0.8307	0.8120	0.6002	0.5835
Raghu et al. (2019)	0.9703	0.9709	0.8005	0.8626
Mozannar & Sontag (2020)	0.9489	0.9669	0.7295	0.7059
Okati et al. (2021)	0.9402	0.9439	0.7040	0.7648
Verma & Nalisnick (2022)	0.9588	0.9741	0.7938	0.8448
Mozannar et al. (2023)	0.9479	0.9757	0.7753	0.8724
	Trained without Ground Truth			
Madras et al. (2018)	0.8605	0.8838	0.5998	0.5951
Raghu et al. (2019)	0.9668	0.9688	0.7834	0.8621
Mozannar & Sontag (2020)	0.9254	0.9688	0.7491	0.6774
Okati et al. (2021)	0.8811	0.9002	0.7522	0.7195
Verma & Nalisnick (2022)	0.9450	0.9711	0.6090	0.8668
Mozannar et al. (2023)	0.9446	0.9682	0.7515	0.8668
Ours	**0.9891**	**0.9926**	**0.8778**	**0.9237**

Table 4: Comparing PHICO to LNL and MRL methods with asymmetric label noise 10%, 30%, 40% on CIFAR-10, referencing accuracy from Karim et al.; Zheltonozhskii et al.

Method	Noise Rate		
	10%	30%	40%
	LNL methods		
CE	0.888	0.817	0.761
JPL Kim et al. (2021)	0.942	0.925	0.907
Dmix Li et al. (2020)	0.938	0.925	0.917
ELR Liu et al. (2020)	0.954	0.947	0.930
MOIT Ortego et al. (2021)	0.942	0.941	0.932
C2D Zheltonozhskii et al. (2022)	-	-	0.937
UNICON Karim et al. (2022)	0.953	0.948	0.941
	MRL methods		
Fast-DS Sinha et al. (2018)	0.9847	0.9836	0.9811
CrowdLab Goh et al. (2023)	0.9878	0.9874	0.9818
Ours	**0.9978**	**0.9959**	**0.9927**

refining the classifier to outperform humans and using a post-hoc rejector to decide who is more likely to err on individual case and Mozannar & Sontag (2020), Verma & Nalisnick (2022), Mozannar et al. (2023) which propose surrogate loss functions to better align the optimisation with deferral goals. The comparison involves training models *with* and *without* ground truth, assessed by accuracy against test set ground truth annotations (see Table 3). When trained without ground truth, the training set consensus $\bar{\mathbf{y}}$ is used. Remarkably, our models trained *without* ground truth outperform those trained *with* ground truth.

Table 4 shows a comparison between PHICO and LNL and MRL methods on CIFAR-10, following Karim et al. (2022) using a Vit-Base-16 backbone pre-trained on ImageNet-21K. In this experiment, we simulate six users, each introducing a 10% asymmetric noise in three class pairs (Airplane↔Bird, Truck↔Automobile, and Horse↔Deer). Subsequently, we trained and evaluated PHICO with $K = 3$. The same experiment was repeated for 30% and 40% noise rates. This comparison uses the cross entropy (CE) baseline and the following LNL methods: DMix (Li et al., 2020) based on semi-supervised learning, ELR (Liu et al., 2020) exploring a regularised loss, C2D (Zheltonozhskii et al., 2022) addressing the warm-up obstacle, JPL (Kim et al., 2021) exploring negative learning, MOIT (Ortego et al., 2021) combining contrastive and semi-supervised learning, and UNICON (Karim et al., 2022) providing a unified framework for supervised and unsupervised learning. We also include the following MRL methods in the comparison: Goh et al. (2023) exploring a majority voting followed by ensemble method to reach consensus, and Sinha et al. (2018) introducing a rapid vote aggregation method for consensus labelling based on expectation maximization.

5 Ablation Studies

We report the results and main conclusions of the ablation study, where details can be found in the cited appendices. We study the effect of noisy label augmentation in Table 5 (details explained in Appendix K), which evaluates post alteration accuracy against augmentation times $G \in \{0, 1, 3, 5\}$, where results show a large accuracy increase from $G = 0$ to $G = 1$ and a steady improvement for $G > 1$. Next, we evaluate different backbone models, including DenseNet-121, ResNet-50 and ViT/B-16. Results in Table 8 show consistent improvement across all backbones while remaining agnostic to the backbone model. Additionally, comparison to related methods confirms our superior performance across different backbone models in Table 7 (see details in Appendix J). Table 9 performs an ablation study by varying asymmetric noise rates (40%, 60%, 80%, 90%) on CIFAR-10 simulations (details in Appendix I), showcasing the robustness of our approach with accuracy above 86% in all noise rates. Table 6 (and Appendix H) shows the variation in post-alteration accuracy for higher $K \in 1, 2, 3, 6, 10$ with CIFAR-10N. Increasing K from 1 to 3 improves accuracy, but it declines for $K > 3$ due to fewer users per profile. Additional experiments on positive and negative alterations around the optimal K are shown in Tables 16 and 17. Appendix L explores the effect of λ in the loss function equation 4, with $\lambda = 0.1$ yielding the best accuracy.

Table 5: Performance on CIFAR-10N [K=2] as a function of the noisy label augmentation hyper-parameter G.

G	Post-alt. acc.	A+	A-	RA+	RA-
0	0.6148	0.4113	0.3015	0.5770	0.4229
1	0.9889	0.9530	0.0040	0.9958	0.0042
3	0.9891	0.9541	0.0040	0.9958	0.0042
5	0.9892	0.9522	0.0035	0.9963	0.0037

Table 6: Performance on CIFAR-10N as a function of the number of clusters K.

K	Post-alt. acc.	A+	A-	RA+	RA-
K=1	0.9878	0.9528	0.0055	0.9943	0.0057
K=2	0.9891	0.9541	0.0040	0.9958	0.0042
K=3	0.9892	0.9542	0.0040	0.9958	0.0042
K=6	0.9877	0.9438	0.0037	0.9961	0.0039
K=10	0.9728	0.9135	0.0038	0.9959	0.0041

Table 7: Comparison between HAICO-CN and competing methods in the literature with different base models using CIFAR-10N.

Method	ResNet50	DenseNet121	ViTB16
	With Ground Truth		
Madras et al. (2018)	0.8508	0.8412	0.8307
Raghu et al. (2019)	0.8707	0.8281	0.9703
Mozannar & Sontag (2020)	0.8514	0.8502	0.9489
Okati et al. (2021)	0.8103	0.8021	0.9402
Verma & Nalisnick (2022)	0.7008	0.6332	0.9588
Mozannar et al. (2023)	0.7822	0.8496	0.9479
	Without Ground Truth		
Madras et al. (2018)	0.8427	0.8474	0.8605
Raghu et al. (2019)	0.8316	0.8788	0.9668
Mozannar & Sontag (2020)	0.7030	0.8489	0.9254
Okati et al. (2021)	0.8003	0.7055	0.8811
Verma & Nalisnick (2022)	0.6241	0.5932	0.9450
Mozannar et al. (2023)	0.6588	0.8470	0.9446
Ours	**0.9677**	**0.9686**	**0.9891**

Table 8: Ablation with CIFAR-10N using different backbone models as the base model $f_{\psi_k}(.)$.

Backbone Model	Original Accuracy	Post-alt. acc.	A+	A-	RA+	RA-
ResNet-50	0.8461	0.9677	0.8623	0.0131	0.9849	0.0150
DenseNet-121	0.8464	0.9686	0.8535	0.0105	0.9878	0.0122
Vit/B-16	0.8365	0.9891	0.9541	0.0040	0.9958	0.0042

Table 9: Performance on CIFAR-10 as a function of noise rate

Asymmetric Noise Rate	Original Accuracy	Post alt. acc. (K=3)
40%	0.9198	0.9923
60%	0.8800	0.9678
80%	0.8400	0.8788
90%	0.8202	0.8684

6 Discussion

An intriguing aspect of PHICO is its capability to correct errors even when both humans and AI models make mistakes. Sec. 4.2 and Appendix F suggest it happens from the personalised AI cooperative model that associates noisy labelling patterns of raters and the AI model to the correct label. A necessary condition for this to happen is to prove that $P(C|\neg A, \neg B) > 0$, where A represents the event that the base model provides a correct prediction, B denotes the event that the human provides a correct label, and C is the event that our joint decision model produces a correct classification. Assuming that the base model and humans can make mistakes, and that events A and B are independent (and also independent given C), we trivially have: $P(C|\neg A, \neg B) = \frac{P(\neg A, \neg B|C).P(C)}{P(\neg A, \neg B)} = \frac{(1-P(A|C)).(1-P(B|C)).P(C)}{(1-P(A))(1-P(B))} > 0$ because $0 < P(B|C), P(A|C), P(A), P(B), P(C) < 1$.

Future work for PHICO includes addressing the complexity of human-AI cooperation, where interactions may change human behavior over time. While PHICO currently doesn't account for this dynamic, it could be adapted by regularly updating user's assigned profile to reflect evolving interactions and noisy patterns. Additionally, we will aim to create a more efficient few-shot profiling process and extend PHICO for multi-label classification, building on insights from Li et al. (2022); Kye et al. (2022). Enhancing privacy in learned profiles through local differential privacy Yang et al. (2022) is also a key direction for future work.

7 Conclusions

We introduced PHICO, a novel human-AI cooperation framework that integrates noisy label learning methods with personalized AI cooperative models. Through both a theoretical convergence proof and an empirical evaluation across diverse datasets, including CIFAR-10N, CIFAR-10H, Fashion-MNIST-H, AgNews, and Chaoyang histopathology, we demonstrated the robustness and effectiveness of PHICO. We also proposed a new measure, the alteration rate, to quantify the impact of PHICO on label modifications from both human and AI sources. With its model-agnostic design and the ability to manage multi-rater datasets without ground truth labels, PHICO offers an effective solution to human-AI cooperation tasks.

REFERENCES

Jean V Alves, Diogo Leitão, Sérgio Jesus, Marco OP Sampaio, Pedro Saleiro, Mário AT Figueiredo, and Pedro Bizarro. Fifar: A fraud detection dataset for learning to defer. *arXiv preprint arXiv:2312.13218*, 2023.

G. Anandalingam and T. L. Friesz. Hierarchical optimization: An introduction. *Annals of Operations Research*, 34(1):1–11, 1992. ISSN 1572-9338. doi: 10.1007/BF02098169. URL https://doi.org/10.1007/BF02098169.

Kenneth George Binmore. *Mathematical Analysis: a straightforward approach*. Cambridge University Press, 1982.

Léon Bottou and Yoshua Bengio. Convergence properties of the k-means algorithms. In G. Tesauro, D. Touretzky, and T. Leen (eds.), *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1994. URL https://proceedings.neurips.cc/paper_files/paper/1994/file/a1140a3d0df1c81e24ae954d935e8926-Paper.pdf.

Jerome Bracken and James T McGill. Mathematical programs with optimization problems in the constraints. *Operations research*, 21(1):37–44, 1973.

On-Road Automated Driving (ORAD) Committee. *Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems*. SAE International, 2014.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 113–123, 2019.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 702–703, 2020.

Allan Dafoe, Yoram Bachrach, Gillian Hadfield, Eric Horvitz, Kate Larson, and Thore Graepel. Cooperative ai: machines must learn to find common ground. *Nature*, 593(7857):33–36, 2021.

Vinod Kumar Dehariya, Shailendra Kumar Shrivastava, and RC Jain. Clustering of image data set using k-means and fuzzy k-means algorithms. In *2010 International conference on computational intelligence and communication networks*, pp. 386–391. IEEE, 2010.

Dominik Dellermann, Adrian Calma, Nikolaus Lipusch, Thorsten Weber, Sascha Weigel, and Philipp Ebel. The future of human-ai collaboration: a taxonomy of design knowledge for hybrid intelligence systems. *arXiv preprint arXiv:2105.03354*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Hui Wen Goh, Ulyana Tkachenko, and Jonas Mueller. Crowdlab: Supervised learning to infer consensus labels and quality scores for data with multiple annotators, 2023.

Ben Green and Yiling Chen. The principles and limits of algorithm-in-the-loop decision making. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW), nov 2019. doi: 10.1145/3359152. URL https://doi.org/10.1145/3359152.

Richard J Hathaway and James C Bezdek. Local convergence of the fuzzy c-means algorithms. *Pattern recognition*, 19(6):477–480, 1986.

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*, pp. 770–778, 2016.

Kenneth Holstein and Vincent Aleven. Designing for human-ai complementarity in k-12 education, 2021.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Takashi Ishida, Ikko Yamane, Nontawat Charoenphakdee, Gang Niu, and Masashi Sugiyama. Is the performance of my deep network too good to be true? a direct approach to estimating the bayes error in binary classification, 2023.

Wei Ji, Shuang Yu, Junde Wu, Kai Ma, Cheng Bian, Qi Bi, Jingjing Li, Hanruo Liu, Li Cheng, and Yefeng Zheng. Learning calibrated medical image segmentation via multi-rater agreement modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12341–12351, 2021.

Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. Beyond synthetic noise: Deep learning on controlled noisy labels. In *International conference on machine learning*, pp. 4804–4815. PMLR, 2020.

Kevin B Johnson, Wei-Qi Wei, Dilhan Weeraratne, Mark E Frisse, Karl Misulis, Kyu Rhee, Juan Zhao, and Jane L Snowdon. Precision medicine, ai, and the future of personalized health care. *Clinical and translational science*, 14(1):86–93, 2021.

Nazmul Karim, Mamshad Nayeem Rizve, Nazanin Rahnavard, Ajmal Mian, and Mubarak Shah. Unicon: Combating label noise through uniform selection and contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9676–9686, 2022.

Gavin Kerrigan, Padhraic Smyth, and Mark Steyvers. Combining human predictions with model probabilities via confusion matrices and calibration. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 4421–4434. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper_files/paper/2021/file/234b941e88b755b7a72a1c1dd5022f30-Paper.pdf`.

Youngdong Kim, Juseung Yun, Hyounguk Shon, and Junmo Kim. Joint negative and positive learning for noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9442–9451, 2021.

Rafal Kocielnik, Saleema Amershi, and Paul N Bennett. Will you accept an imperfect AI? exploring designs for adjusting end-user expectations of AI systems. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, number Paper 411 in CHI '19, pp. 1–14, New York, NY, USA, May 2019. Association for Computing Machinery.

Seong Min Kye, Kwanghee Choi, Joonyoung Yi, and Buru Chang. Learning with noisy labels by efficient transition matrix estimation to combat label miscorrection. In *European Conference on Computer Vision*, pp. 717–738. Springer, 2022.

Ariel Levy, Monica Agrawal, Arvind Satyanarayan, and David Sontag. Assessing the impact of automated suggestions on decision making: Domain experts mediate model errors but take less initiative. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi: 10.1145/3411764.3445522. URL `https://doi.org/10.1145/3411764.3445522`.

Junnan Li, Richard Socher, and Steven C. H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning, 2020.

Shikun Li, Xiaobo Xia, Hansong Zhang, Yibing Zhan, Shiming Ge, and Tongliang Liu. Estimating noise transition matrix with label correlations for noisy multi-label learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 24184–24198. Curran Associates, Inc.,

2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/98f8c89ae042c512e6c87e0e0c2a0f98-Paper-Conference.pdf.

Han Liu, Vivian Lai, and Chenhao Tan. Understanding the effect of out-of-distribution examples and interactive explanations on human-ai decision making. *Proc. ACM Hum.-Comput. Interact.*, 5 (CSCW2), oct 2021. doi: 10.1145/3479552. URL https://doi.org/10.1145/3479552.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Yang Liu, Hao Cheng, and Kun Zhang. Identifiability of label noise transition matrix. In *International Conference on Machine Learning*, pp. 21475–21496. PMLR, 2023.

David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 6150–6160, Red Hook, NY, USA, 2018. Curran Associates Inc.

Mehrdad Mahdavi, Lijun Zhang, and Rong Jin. Mixed optimization for smooth functions. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/file/f73b76ce8949fe29bf2a537cfa420e8f-Paper.pdf.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In Hal Daumé Iii and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7076–7087. PMLR, 2020.

Hussein Mozannar, Hunter Lang, Dennis Wei, Prasanna Sattigeri, Subhro Das, and David Sontag. Who should predict? exact algorithms for learning to defer to humans. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent (eds.), *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 10520–10545. PMLR, 25–27 Apr 2023. URL https://proceedings.mlr.press/v206/mozannar23a.html.

Nastaran Okati, Abir De, and Manuel Rodriguez. Differentiable learning under triage. *Advances in Neural Information Processing Systems*, 34:9140–9151, 2021.

Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6606–6615, 2021.

R Parasuraman, T B Sheridan, and C D Wickens. A model for types and levels of human interaction with automation. *IEEE Trans. Syst. Man Cybern. A Syst. Hum.*, 30(3):286–297, May 2000.

Vivak Patel, Shushu Zhang, and Bowen Tian. Global convergence and stability of stochastic gradient descent. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 36014–36025. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/ea05e4fc0299c27648c9985266abad47-Paper-Conference.pdf.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Computing Surveys (CSUR)*, 54(4):1–34, 2021.

Ben Shneiderman. *Human-centered AI.* Oxford University Press, 2022.

Vaibhav B Sinha, Sukrut Rao, and Vineeth N Balasubramanian. Fast dawid-skene: A fast vote aggregation scheme for sentiment classification, 2018.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, PP, March 2022a.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022b.

Mark Steyvers, Heliodoro Tejeda, Gavin Kerrigan, and Padhraic Smyth. Bayesian modeling of human-AI complementarity. *Proceedings of the National Academy of Sciences of the United States of America*, 119(11):e2111547119, March 2022.

Eleni Straitouri, Lequn Wang, Nastaran Okati, and Manuel Gomez Rodriguez. Improving expert predictions with prediction sets. *arXiv preprint arXiv:2201.12006*, 2023.

Barry Strauch. Ironies of automation: Still unresolved after all these years. *IEEE Transactions on Human-Machine Systems*, 48(5):419–433, 2018. doi: 10.1109/THMS.2017.2732506.

Cheng Tang and Claire Monteleoni. Convergence rate of stochastic k-means. In Aarti Singh and Jerry Zhu (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1495–1503. PMLR, 20–22 Apr 2017. URL `https://proceedings.mlr.press/v54/tang17b.html`.

Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE Transactions on Neural Networks and Learning Systems*, 32(11):4793–4813, 2021. doi: 10.1109/TNNLS.2020.3027314.

Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 22184–22202. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/verma22c.html`.

Kailas Vodrahalli, Tobias Gerstenberg, and James Y Zou. Uncalibrated models can improve human-ai collaboration. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 4004–4016. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/1968ea7d985aa377e3a610b05fc79be0-Paper-Conference.pdf`.

Dakuo Wang, Liuping Wang, Zhan Zhang, Ding Wang, Haiyi Zhu, Yvonne Gao, Xiangmin Fan, and Feng Tian. "brilliant AI doctor" in rural clinics: Challenges in AI-powered clinical decision support system deployment. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, number Article 697 in CHI '21, pp. 1–18, New York, NY, USA, May 2021. Association for Computing Machinery.

Simon K Warfield, Kelly H Zou, and William M Wells. Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation. *IEEE transactions on medical imaging*, 23(7):903–921, 2004.

Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021. ISBN 9780999241165.

Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. A survey of human-in-the-loop for machine learning. *Future Gener. Comput. Syst.*, 135(C):364–381, oct 2022. ISSN 0167-739X. doi: 10.1016/j.future.2022.05.014. URL `https://doi.org/10.1016/j.future.2022.05.014`.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Jinglin Xu, Junwei Han, Kai Xiong, and Feiping Nie. Robust and sparse fuzzy k-means clustering. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pp. 2224–2230. AAAI Press, 2016. ISBN 9781577357704.

Mengmeng Yang, Ivan Tjuawinata, and Kwok-Yan Lam. K-means clustering with local dx-privacy for privacy-preserving data analysis. *IEEE Transactions on Information Forensics and Security*, 17:2524–2537, 2022. doi: 10.1109/TIFS.2022.3189532.

Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1657–1667, 2022.

APPENDIX

APPENDIX: TABLE OF CONTENTS

A CONSENSUS LABEL ESTIMATION

Many multi-rater input datasets lack ground truth labels. To address this, PHICO is built to function effectively without relying on them. During training, we use Crowdlab (Goh et al., 2023) to estimate a consensus label $\bar{\mathbf{y}}_i$, which approximates the true clean label $\mathbf{y}_i$. Crowdlab works in two steps. In the first step, it estimates a consensus by majority vote $\bar{\mathbf{y}}_i'$ per training sample. In the second step, it trains a classifier using the initial consensus and obtains predicted class probabilities for each training example. After that, Crowdlab uses these predicted probabilities along with the original annotations from raters to estimate a better consensus, creating the following ensemble,

$$\bar{\mathbf{y}}_i = \mathbf{w}_\gamma \times f_\gamma(\mathbf{x}_i) + \mathbf{w}_1 \times \tilde{\mathbf{y}}_{i,1} + ... + \mathbf{w}_{|\mathcal{A}|} \times \tilde{\mathbf{y}}_{i,|\mathcal{A}|}, \tag{7}$$

where $f_\gamma : \mathcal{X} \to \Delta^{C-1}$ is a classifier trained with the majority vote label $\bar{\mathbf{y}}_i'$ to output a categorical distribution for C classes, and the weights $\mathbf{w}_\gamma, \mathbf{w}_1, ..., \mathbf{w}_{|\mathcal{A}|}$ are assigned according to an estimate of how trustworthy the model is, compared to each individual annotator. The outcome of Crowdlab is a consensus labelled training set denoted by $\bar{\mathcal{D}} = \{(\mathbf{x}_i, \bar{\mathbf{y}}_i)\}_{i=1}^N$. Note that the consensus label is necessary only when the clean label $\mathbf{y}_i$ is latent. If such clean label is observed, then Crowdlab is no longer needed, and PHICO can be trained with $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$.

B DECIDING THE OPTIMAL NUMBER OF PROFILES

We determine the optimal number of profiles K with the silhouette score defined by,

$$S_k = \frac{1}{|A|} \sum_{j \in \mathcal{A}} \frac{b(\mathbf{s}_j) - a(\mathbf{s}_j)}{\max\{a(\mathbf{s}_j), b(\mathbf{s}_j)\}}, \tag{8}$$

where $a(\mathbf{s}_j)$ denotes the sample's intra-profile distance (i.e., the average L2 distance to all other points in the same profile), $b(\mathbf{s}_j)$ represents the inter-profile distance (i.e., the lowest average L2 distance to all points in any other profile). The mean silhouette score for K profiles is defined by $S(K) = \frac{1}{K} \sum_{k=1}^{K} S_k$. The optimal number of profiles for the dataset is identified by selecting K that yields the highest silhouette score.

C Experimental Setup

C.1 Datasets

CIFAR-10 comprises 50,000 training, 200 validation, and 9,800 testing class-balanced color images, each sized 32×32, and has 10 classes. **CIFAR-10N** extends the training set of CIFAR-10 by crowd-sourcing its labelling to 747 annotators, where each image has three labels produced by different annotators. The majority of annotators provided 200 labels. **CIFAR-10H** extends the CIFAR-10 testing set by crowd-sourcing it to 2571 annotators, each contributing with 210 labels. The resulting label set contains an average of 51 labels per image. Fashion-MNIST (Xiao et al., 2017) comprises 60,000 training samples, and 10,000 testing samples with class-balanced images (belonging to one of 10 classes) of size 28×28 pixels. Fashion-MNIST-H (Ishida et al., 2023) extends the Fashion-MNIST's testing set of 10,000 images by crowd-sourcing them to 885 annotators. The resulting label set contains an average of 66 labels per image. We train the model using **Fashion-MNIST-H**'s annotations on Fashion-MNIST's test set, utilizing its 10,000 test images for training and splitting the original training set into 200 validation and 59,800 test images. **AgNews** is a text classification dataset comprising 120,000 training, 200 validation and 7,400 testing class-balanced news articles categorized into 4 classes. Lastly, **Chaoyang** is a pathological dataset featuring four classes of images, having a training set of 4021 images, a validation set with 80 images, and a testing set of 2059 images. Notably, each image in the training set is labeled by three experts, resulting in three labels per image, and the testing set presents a single consensus label.

C.2 Dataset with simulated annotators

The simulation experiments on CIFAR-10 consists of a pairwise flipping experiment, where 8 out of 10 classes have 100% of clean labels, but in two classes, 80% of samples have labels flipped to the incorrect class. We simulate three profiles of users, one that flips 80% of the samples between classes airplane$\leftrightarrow$bird, another profile that flips horse$\leftrightarrow$deer, and the other profile that flips truck$\leftrightarrow$automobile. For each profile, we simulate five training and five testing users, producing a total of $5 \times 3 = 15$ unique users for training and another 15 users for testing. The training images, together with the 15 labels/image by the training users, will form $\tilde{\mathcal{D}}$ and it is used to build K profiles and train the OVA SVM, where K is automatically chosen based on the silhouette score in equation 8. The model for each profile k, $m_{\theta_k}(.)$ in equation 3, uses a ResNet-18 He et al. (2016) as $f_{\psi_k}(.)$ Figure 3 shows the noise matrices used for simulating CIFAR-10 users.

For AgNews, we adopted a pairwise flipping on two out of four classes, where 80% of samples are flipped to the incorrect class while the remaining 2 classes have 100% clean labels. We simulate three profiles of users, one that flips between classes business$\leftrightarrow$science/technology, another profile that flips world$\leftrightarrow$sports, and the third profile that flips sports$\leftrightarrow$business. Five training and five testing users are simulated for each profile producing a total of 15 unique users for training and another 15 for testing. The training articles together with 15 training labels/article make up $\tilde{\mathcal{D}}$ which is used to make K profiles and train OVA SVM, where K is automatically chosen based on the silhouette score in equation 8. A Bert-Base-Uncased Devlin et al. (2018) model is used as $f_{\psi_k}(.)$ when training $m_{\theta_k}(.)$ in equation 3 for each profile k. The figure 2 shows the noise matrices used for simulating AgNews users.

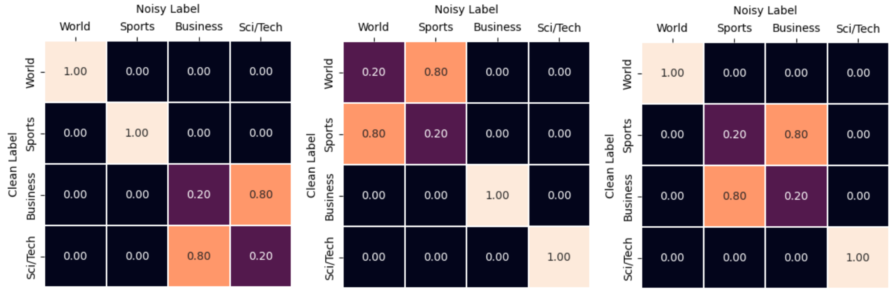

Figure 2: Noise matrices used for simulating users with AgNews.

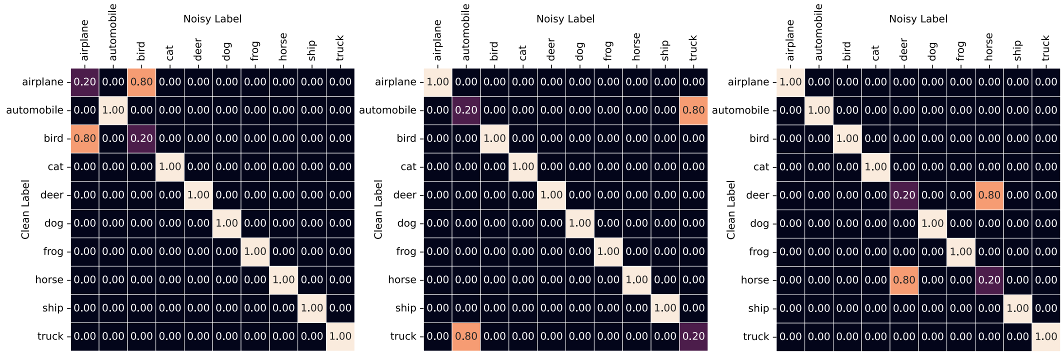

Figure 3: Noise matrices used for simulating users with CIFAR-10.

C.3 Dataset with real annotators

When training with CIFAR-10N, we present two experiments. For the first experiment, the labels from 747 annotators form $\tilde{\mathcal{D}}$. Out of them, 155 were identified for having annotated at least 20 images per class, and they were split in half, taking 79 as training users and 80 as testing users. The training users' labels are used to build the K profiles and train the OVA SVM classifier, where K is automatically chosen based on the silhouette score in equation 8. During testing, a testing user's noisy-label transition matrix is estimated using the annotator's labels and consensus labels. This matrix is used to simulate noisy annotations from that testing user. Therefore, 80 noisy test sets are produced, with each representing the biases that each user possesses. The model for each profile k, denoted by $m_{\theta_k}(.)$, uses ViT-Base-16 (Dosovitskiy et al., 2020) as the backbone for $f_{\psi_k}(.)$.

For the second CIFAR-10N experiment, we use CIFAR-10H as the testing set, where the labels from testing users were used without any modification for user profiling. The same labels were used to estimate a noise transition matrix and simulate their own test set. For all 2571 users, their own test test was simulated with own biases. The models trained for CIFAR-10N were used for this experiment.

For the Fashion-MNIST-H experiment, the labels from all 885 annotators are taken to form the $\tilde{\mathcal{D}}$. Then, 366 out of 885 users are chosen since they have annotated at least 20 images per class and are split in half to have 183 users for training and 183 for testing. The training users' labels are used to build the K profiles and train the OVA SVM classifier, where K is automatically chosen based on the silhouette score in equation 8. During testing, the testing user's noisy-label transition matrix is estimated using the annotator's labels and consensus labels. This matrix is used to simulate noisy annotations from that testing user. Therefore, 183 noisy testing sets are produced, with each representing the biases that each user possesses. The model for each profile k, represented by $m_{\theta_k}(.)$ uses DenseNet-121 (Huang et al., 2017) for $f_{\psi_k}(.)$.

Chaoyang has three annotators per image, which form the $\tilde{\mathcal{D}}$. Training users are used to make K profiles, and train an OVA SVM, where K is automatically chosen based on the silhouette score in equation 8. For each profile k, a model $m_{\theta_k}(.)$ is trained with a ViT-Large-16 as the backbone for $f_{\psi_k}(.)$. During testing, user's noisy-label transition matrix is estimated using the annotator's labels and consensus labels. This matrix is used to simulate noisy annotations from that user, resulting three noisy test sets.

Our method retain annotators' noisy label patterns, but it's important to note that Fashion-MNIST-H and Chaoyang test sets are simulated and might not completely mimic real annotator inputs. In contrast, CIFAR-10N and CIFAR-10H, with human labels for CIFAR-10's training and testing sets, offer a more realistic setup with crowd-sourced labels in both phases, better reflecting real-world conditions.

In our experiments, we use various backbone models to showcase our model's robustness. An ablation study in Appendix J details the switch from ViT-Base-16 (Dosovitskiy et al., 2020) to DenseNet-121 (Huang et al., 2017) and Resnet-50 He et al. (2016) on CIFAR-10N.

In our CIFAR experiments, we adopted the data augmentation policy introduced by Cubuk et al. (2019). Also, for Fashion-MNIST, alongside random horizontal and vertical flips, we integrated auto augmentations as proposed by Cubuk et al. (2020). For the Chaoyang dataset, data augmentation was limited to random resized crops of dimensions 224×224. For the AgNews dataset, the title and description were concatenated and truncated to maximum length of 64 tokens. We rely on pre-trained models for f_{ψ_k} because of their robustness to noisy labels (Jiang et al., 2020) (e.g., ViT models were pre-trained on ImageNet-21K, while ResNet-18 and DenseNet-121 models were pre-trained on ImageNet-1K. Bert model and Bert tokenizer are trained on a large corpora of articles in self-supervised fashion). Adam optimizer was employed for training $f_{\psi_k}(.)$ with consensus $\bar{\mathcal{D}}$, where NAdam was used for training $m_{\theta_k}(.)$ on $\hat{\mathcal{D}}$, each utilizing their respective default learning rates. Implementations were done in PyTorch and executed on an NVIDIA GeForce RTX 4090 GPU.

D Theoretical proof of convergence of PHICO

D.1 Convergence of fuzzy k-means

Each annotator $j \in \mathcal{A}$ is represented by a set of labels that this user has given to instances of the training set. Assuming that the training set has N instances belonging to one of C classes and each instance has a label $y \in \{0, 1, 2, ..., C-1\} = \mathcal{C}$, then, v_j is an N dimensional array of integers denoted by $v \in \mathcal{C}^N$ representing user j's annotations.

We assume an additive label noise process defined by $\tilde{y} = y + \epsilon$, where $\epsilon \in \mathbb{Z}$ denotes an integer number generator. For example, if $y = 0$ and $\epsilon = 1$, then $\tilde{y} = 1$. Similarly an N-dimensional vector j is affected by the same process – for instance, if we have $v_j = [0, 1, 2]$ and ϵ is $[1, 0, -2]$, this forms the user j's noisy vector $\mathbf{s}_j = [1, 1, 0] \in \mathcal{C}^N$.

Let $\{\mathbf{s}_j\}_{j \in \mathcal{A}}$ form the noisy labels from the users in $\mathcal{A}$. The clustering of users with K means can be written as an optimisation process using the following cost function

$$f(K, \{\mathcal{L}_r\}_{r=1}^K, \{\mathbf{c}_r\}_{r=1}^K) := \sum_{r=1}^{K} \sum_{\mathbf{s}_j \in \mathcal{L}_r} ||\mathbf{s}_j - \mathbf{c}_r||^2, \tag{9}$$

where K denotes the number of cluster centroids, $\mathcal{L}_r \subset \{\mathbf{s}_j\}_{j \in \mathcal{A}}$, contains users assigned to centroid $\mathbf{c}_r$. When K is fixed, minimal cost can be achieved by choosing the clustering that assigns each $\mathbf{s}_j$ to the closest centroid following Bottou & Bengio (1994) and Tang & Monteleoni (2017), as in

$$f(K) := \min_{\{\mathcal{L}_r\}_{r=1}^K, \{\mathbf{c}_r\}_{r=1}^K} f(K, \{\mathcal{L}_r\}_{r=1}^K, \{\mathbf{c}_r\}_{r=1}^K) = \min_{\{\mathcal{L}_r\}_{r=1}^K} \sum_{r=1}^{K} \sum_{\mathbf{s}_j \in \mathcal{L}_r} ||\mathbf{s}_j - \mathbf{c}_r||^2. \tag{10}$$

Bottou & Bengio (1994) and Tang & Monteleoni (2017) present evidence that clustering converges under fixed cluster numbers (as in equation 10 in Tang & Monteleoni (2017), despite being NP-hard in general (equation 9 in Tang & Monteleoni (2017)).

The fuzzy K-means is an extension of the classic K-means clustering algorithm, shown above, where each data point has a degree of belonging to each cluster, rather than a binary membership as in traditional K-means. More specifically, in fuzzy K-means, we minimise the following cost function,

$$f(K) := \min_{\{\mathbf{u}_{j,r}\}_{j \in \mathcal{A}, r=1..K}, \{\mathbf{c}_r\}_{r=1}^K} \sum_{r=1}^{K} \sum_{j \in \mathcal{A}} \mathbf{u}_{j,r}^b \times ||\mathbf{s}_j - \mathbf{c}_r||^2, \tag{11}$$

where $b > 1$ is the fuzziness parameter, and $\mathbf{u}_{j,r}$ is the membership degree of $\mathbf{s}_j$ to cluster $\mathbf{c}_r$ with the constraint that $\sum_{r=1}^K \mathbf{u}_{j,r} = 1$. Hathaway & Bezdek (1986) presents the convergence proof of the Fuzzy K-means algorithm, showing that the iterative update rules for the membership matrix and cluster centers lead to the decrease of the objective function and establish conditions for convergence to a local minimum.

D.2 Convergence of the model m_θ

The three component model architecture is optimised towards the objective function 4, which is,

$$\mathcal{L}\left(\{\theta_k^*\}_{k=1}^K\right) = \arg\min_{\{\theta_k\}_{k=1}^K} \frac{1}{K \times |\hat{\mathcal{D}}_k| \times G} \times \sum_{k=1}^{K} \sum_{\left(\mathbf{x}_i, \{\hat{\mathbf{y}}_{i,g}\}_{g=1}^G\right) \in \hat{\mathcal{D}}_k} \ell\left(\bar{\mathbf{y}}_i, m_{\theta_k}(\mathbf{x}_i, \hat{\mathbf{y}}_{i,g})\right) + \lambda \times \ell\left(\hat{\mathbf{y}}_{i,g}, (\mathbf{T}_k)^\top \times m_{\theta_k}(\mathbf{x}_i, \hat{\mathbf{y}}_{i,g})\right),$$

we aim to find $\{\theta_k\}_{k=1}^K$ that minimizes $\mathcal{L}$. Hence, the objective function is a sum of $K \times 2$ cross-entropy losses.

Facts

1. The objective function is differentiable as it is a sum of $K \times 2$ differentiable functions.
2. Smoothness: Given the function $\mathcal{L}$ is differentiable, its gradient $\nabla\mathcal{L}$ is Lipschitz continuous with constant L. This means for any θ and θ'(Patel et al., 2022),

$$\|\nabla\mathcal{L}(\theta) - \nabla\mathcal{L}(\theta')\| \leq L\|\theta - \theta'\|.$$

Gradient Descent Algorithm

The update rule for gradient descent is: $\theta_k^{(t+1)} = \theta_k^{(t)} - \alpha\nabla\mathcal{L}(\theta_k^{(t)})$, where α is the learning rate.

Convergence Proof

Step 1: Descent Lemma For a smooth function with Lipschitz continuous gradient, the following inequality holds (Patel et al., 2022; Mahdavi et al., 2013):

$$\mathcal{L}(\theta_k^{(t+1)}) \leq \mathcal{L}(\theta_k^{(t)}) + \nabla\mathcal{L}(\theta_k^{(t)})^T(\theta_k^{(t+1)} - \theta_k^{(t)}) + \frac{L}{2}\|\theta_k^{(t+1)} - \theta_k^{(t)}\|^2.$$

Substitute the gradient descent update rule into this inequality:

$$\theta_k^{(t+1)} = \theta_k^{(t)} - \alpha\nabla\mathcal{L}(\theta_k^{(t)}),$$
$$\theta_k^{(t+1)} - \theta_k^{(t)} = -\alpha\nabla\mathcal{L}(\theta_k^{(t)}),$$
$$\|\theta_k^{(t+1)} - \theta_k^{(t)}\|^2 = \alpha^2\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

Thus,

$$\mathcal{L}(\theta_k^{(t+1)}) \leq \mathcal{L}(\theta_k^{(t)}) - \alpha\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2 + \frac{L\alpha^2}{2}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

Step 2: Simplifying and rearranging the inequality, we have:

$$\mathcal{L}(\theta_k^{(t+1)}) \leq \mathcal{L}(\theta_k^{(t)}) - \left(\alpha - \frac{L\alpha^2}{2}\right)\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

To ensure that the coefficient of $\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2$ is positive, choose α such that $0 < \alpha < \frac{2}{L}$. A common choice is $\alpha = \frac{1}{L}$:

$$\mathcal{L}(\theta_k^{(t+1)}) \leq \mathcal{L}(\theta_k^{(t)}) - \frac{1}{2L}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

Step 3: Summing the Inequalities over $t = 0, 1, \ldots, T-1$:

$$\sum_{t=0}^{T-1}\left(\mathcal{L}(\theta_k^{(t)}) - \mathcal{L}(\theta_k^{(t+1)})\right) \geq \frac{1}{2L}\sum_{t=0}^{T-1}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

Since $\mathcal{L}(\theta_k^{(t)})$ is non-increasing,

$$\mathcal{L}(\theta_k^{(0)}) - \mathcal{L}(\theta_k^{(T)}) \geq \frac{1}{2L}\sum_{t=0}^{T-1}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2.$$

Step 4: Convergence of the Gradient Norm. By dividing both sides by T:

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2 \leq \frac{2L(\mathcal{L}(\theta_k^{(0)})-\mathcal{L}(\theta_k^{(T)}))}{T}.$$

As $t \to \infty$, $\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla\mathcal{L}(\theta_k^{(t)})\|^2 \to 0$, which implies that

$$\|\nabla\mathcal{L}(\theta_k^{(t)})\| \to 0 \quad \text{as} \quad t \to \infty.$$

This means that the gradient of $\mathcal{L}(.)$ converges to zero as $t \to \infty$. Hence, given that the function $\mathcal{L}(.)$ is smooth and its gradient is Lipschitz continuous, the gradient descent algorithm consists of a sequence of iterates $\{\theta_k^{(t)}\}$ that converges to a stationary point of the objective function $\mathcal{L}$.

Linear combination of convergent functions is also convergent (Binmore, 1982).

D.3 Convergence of the training process

An overall p-level hierarchical optimization converges, under sufficient conditions such as sequential decision making, dependence of subsequent level's problem on previous level's problem, non-empty solution sets of levels and existence of optimal solutions for each level (Anandalingam & Friesz, 1992; Bracken & McGill, 1973; Ren et al., 2021). Accordingly, we can structure PHICO's two step training process as a bi-level (p=2) optimization problem, where the first level involves choosing best profiles K followed by a model training process on each profile $K = \{1, .., K\}$.

Let,

- $f(K, \{\mathbf{u}_{j,r}\}_{j\in\mathcal{A}, r=1..K}, \{\mathbf{c}_r\}_{r=1}^{K}) = \sum_{r=1}^{K}\sum_{j\in\mathcal{A}}\mathbf{u}_{j,r}^{b} \times ||\mathbf{s}_j - \mathbf{c}_r||^2$ is the objective function for fuzzy-k means clustering (from eq. 11)
- $\mathcal{L}\left(\{\theta_k^*\}_{k=1}^{K}\right)$ is the objective function for the model training.

Bi-Level Problem Formulation

Our optimisation consists of a bi-level optimisation problem that first finds the set of annotator noise profiles using Fuzzy K-Means, which is used to constrain the optimisation of the objective function 4 given the result from the Fuzzy K-Means, as follows:

$$\text{minimize}_{\{\theta_k\}_{i=1}^{K^*}} \mathcal{L}\left(\{\theta_k\}_{i=1}^{|K^*|}\right)$$
$$\text{subject to } K^*, \{\mathbf{u}_{j,r}^*\}_{j\in\mathcal{A}, r=1..K^*} = \arg\min_{K, \{\mathbf{u}_{j,r}\}_{j\in\mathcal{A}, r=1..K}, \{\mathbf{c}_r\}_{r=1}^{K}} f(K, \{\mathbf{u}_{j,r}\}_{j\in\mathcal{A}, r=1..K}, \{\mathbf{c}_r\}_{r=1}^{K}).$$

Convergence

Upper level convergence: Given the optimal number of profiles K^* from the lower level, the deep learning model's parameters $\{\theta_k\}_{i=1}^{K^*}$ are optimized using gradient descent. This optimization converges as shown in the appendix D.2.

Lower level convergence: The fuzzy K-means algorithm converges, as shown in the appendix D.1.

Overall convergence: Since lower level provides a stable constraint to the upper level, and both problems converge individually, the overall hierarchical optimization problem converges under stated assumptions for each sub-problem (Anandalingam & Friesz, 1992; Bracken & McGill, 1973).

E Statistical confidence of results

Table 10 shows the standard deviation and 95% confidence interval of post-alteration accuracy for real-annotator experiments, under the optimal K from the silhouette score. Results show that PHICO significantly improve users compared to their original accuracy.

Table 10: Standard deviation and confidence interval of experiments with real annotators

Dataset	K	Mean accuracy after alterations	Standard deviation (±)	95% confidence interval
CIFAR10-N	2	0.98913	0.00104	(0.98890, 0.98937)
CIFAR10-H	2	0.99260	0.00240	(0.99250, 0.99271)
Fashion-MNIST-H	2	0.87786	0.00837	(0.87661, 0.87913)
Chaoyang	3	0.92374	0.00388	(0.87438, 0.97312)

F Distribution of decisions made by Human, AI and Human-AI cooperation

Table 11 shows how decisions from human, base model and joint decisions are distributed at each experiment conducted in Section 4. This proportions are computed using the testing set. Decision of human, or the AI model $f_{\psi_k}(.)$, or the cooperation $m_{\theta_k}(.)$ are divided into correct (✓), if their label is equal to the target, or wrong (✗), otherwise. According to Table 11, in all experiments, the majority of correct joint decisions are resulted following both correct human and AI counterparts. On the contrary, the smallest proportion of incorrect joint decisions are made when both individual parties are correct. Further, the results reflect the tendency of joint decision being correct when at least one member of the Human-AI team is correct, as anticipated in a cooperative setting. An interesting observation is that we can also see cases where the cooperative decision is correct even when both individual counterparts are wrong. We believe this showcases the capacity of our approach to learn the joint biases posed by individual parties and intervene in cases where both are weak.

Table 11: Proportion that each combination of Human, AI, or Cooperation is correct (✓) or incorrect (✗). Columns sum to 1 to indicate all possible combinations.

Human	AI $f_{\psi_k}(.)$	Cooperation $m_{\theta_k}(.)$	CIFAR10-N %	CIFAR10-H %	Fashion-Mnist-H %	Chaoyang %
✗	✓	✓	5.15	5.59	4.47	3.35
✓	✗	✓	0.65	2.26	15.05	1.82
✓	✓	✓	93.79	91.35	72.13	92.16
✗	✗	✓	0.05	0.05	4.29	0.13
✗	✓	✗	0.13	0.19	0.33	0.49
✓	✗	✗	0.11	0.39	1.38	1.29
✓	✓	✗	0.00	0.00	0.20	0.00
✗	✗	✗	0.12	0.17	2.17	0.76

G Model Interpretability

This section suggests a way to interpret our model's decision via visualising profile examples and use a interpretable decision model. Figures 4, 5 and 6 illustrates profiles from CIFAR-10 simulation, Fashion-MNIST-H and Chaoyang experiments. Those profile noise visualisations are complemented with sample images where label noise was found and positively altered by the model.

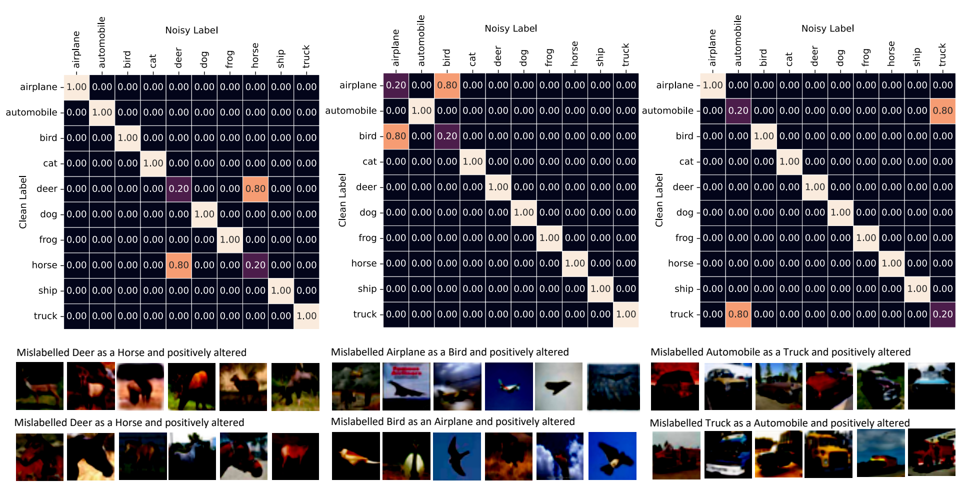

Figure 4: Noise matrices when K=3 in CIFAR-10 simulation experiment

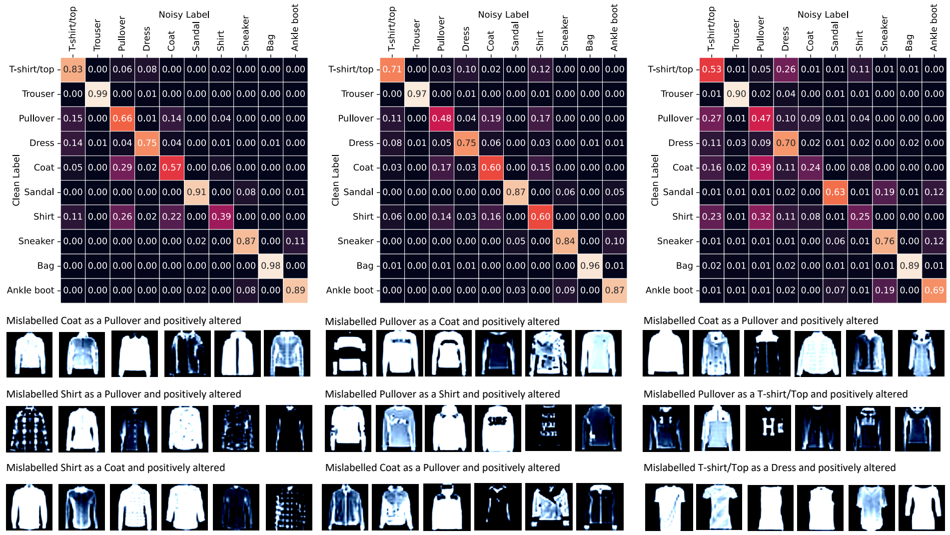

Figure 5: Noise matrices when K=3 in Fashion-MNIST-H experiment

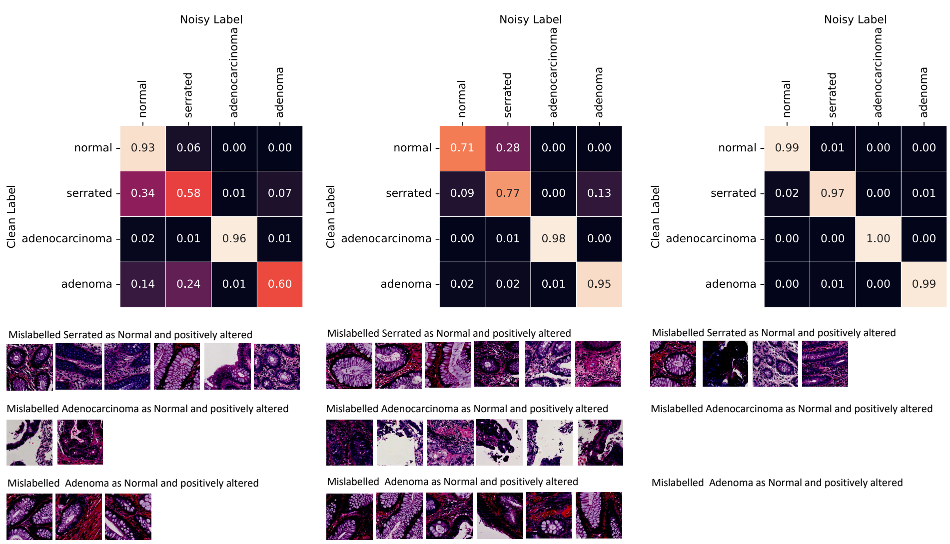

Figure 6: Noise matrices when K=3 in Chaoyang experiment

We also conducted an experiment by replacing the decision model in PHICO with a decision tree model to enable interpretability. The decision tree was trained by concatenating the output logits from base model and human embedding for the training set as in the Section 3.2.

Experiment was done for K=3 in simulation experiment with CIFAR-10 and trained decision trees are plot in the figures 7 and 8. It can be seen the decision tree uses the base model's output features (with the prefix 'b_') as a decision factor when there is user noise present in a specific class. Otherwise the tree relies on human input features with the prefix 'u_'.

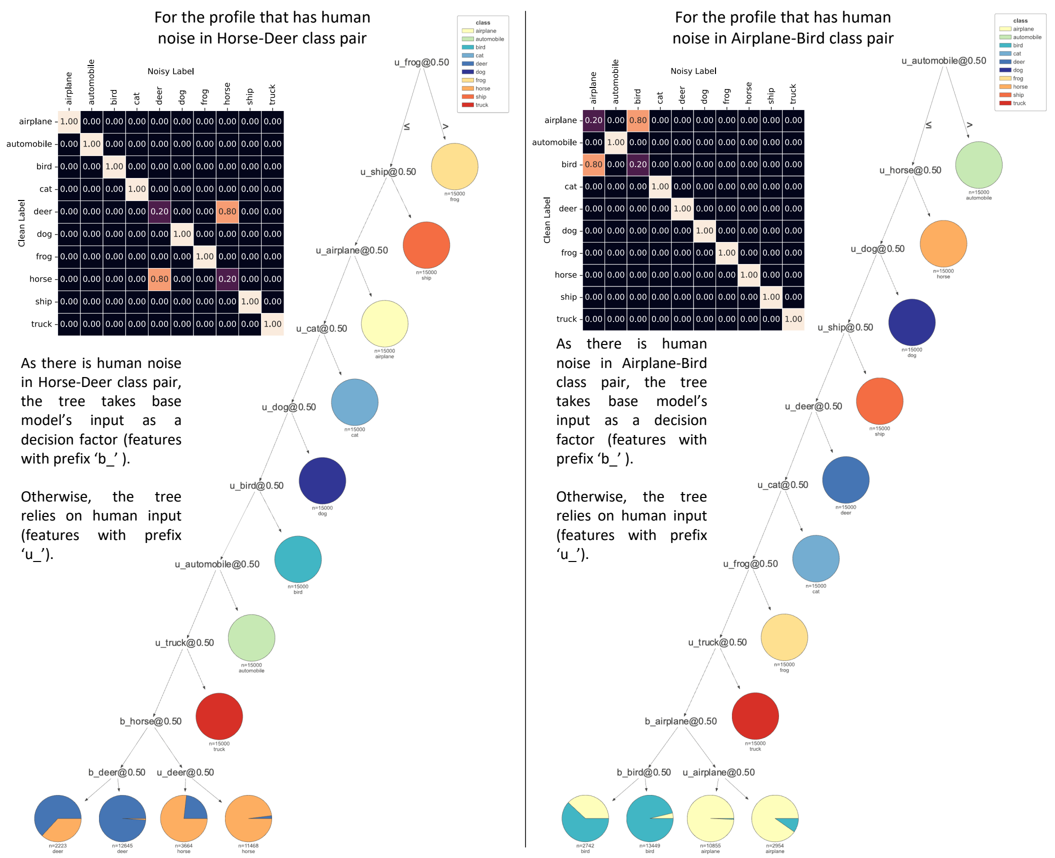

Figure 7: Decision tree behaviour when it is trained on profile with human noise in Horse-Deer class pair (left) and Airplane-Bird class pair (right).

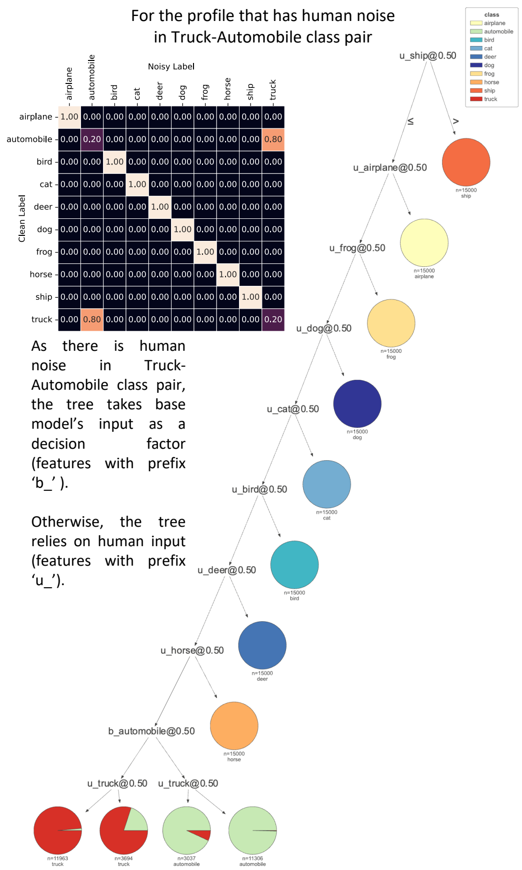

Figure 8: Decision tree behaviour when it is trained on profile with human noise in Truck-Automobile class pair.

Table 12: Post alteration accuracy variation in terms of λ that weights the second term of the loss in equation 4 (with CIFAR-10N).

Backbone model	$\lambda = 0$	$\lambda = 0.01$	$\lambda = 0.1$	$\lambda = 1$	$\lambda = 10$
ResNet-50	0.9295	0.9437	0.9677	0.9399	0.9291
DenseNet-121	0.9364	0.9501	0.9686	0.9373	0.9306
ViT-B/16	0.9821	0.9815	0.9891	0.9759	0.9695

Table 13: Silhouette score variation as a function of K for experiments

K	Sillhoutte score				
	CIFAR-10	AgNews	CIFAR-10N	F-MNIST-H	Chaoyang
2	0.3475	0.4489	0.0103	0.0909	0.6606
3	0.5519	0.5759	0.0077	0.0909	0.9999
4	0.3705	0.3692	0.0035	0.0909	-
5	0.1868	0.1835	-1.0635	0.0406	-
6	0.0057	0.0002	0.0043	0.0021	-
7	0.0064	0.0008	-0.0076	0.0909	-
8	0.0019	0.0016	-0.0033	0.0196	-
9	0.0047	0.0011	-0.0155	0.0909	-
10	0.0028	3.669E-05	-0.0072	0.0196	-

H Performance as a Function of K

H.1 Results of Datasets with simulated annotators

The first and second rows of Table 14 detail the number of testing users that improved (I), maintained (M), or did not improve (NI) with PHICO in the CIFAR-10 and AgNews simulations. Notice how the number of I users increases and NI users decreases in CIFAR-10, showcasing the best personalisation when $K = 3$, which has the highest silhouette score of 0.55195 (silhouette scores in Table 13). At $K = 3$, Table 14 shows that all 15 users improved with CIFAR-10, and Table 15 displays that the average accuracy after alteration is larger than the user's original accuracy. Similarly, AgNews reports its highest post alteration accuracy at $K = 3$ when silhouette score reaches max 0.57586. Also, as K decreases, the post alteration accuracy decreases slightly as a result of the lower number of improved users. Similarly, the simulation results in Table 16 highlights the increase of A_+ when reaching optimal K, accompanied by a decline in negative alterations A_-. Additionally, Table 17 shows an increasing alteration rate with K, reflecting the larger proportion of positive alterations and smaller proportion of negative alterations when reaching optimal $K = 3$ with both simulation datasets. The figures 9 and 10 showcase the estimated noise matrices for $K \in \{1, 2, 3\}$ from CIFAR-10 and AgNews test users. Note that $K = 3$ in those figures, closely resembles the noise matrices used to simulate the users in figures 3 and 2.

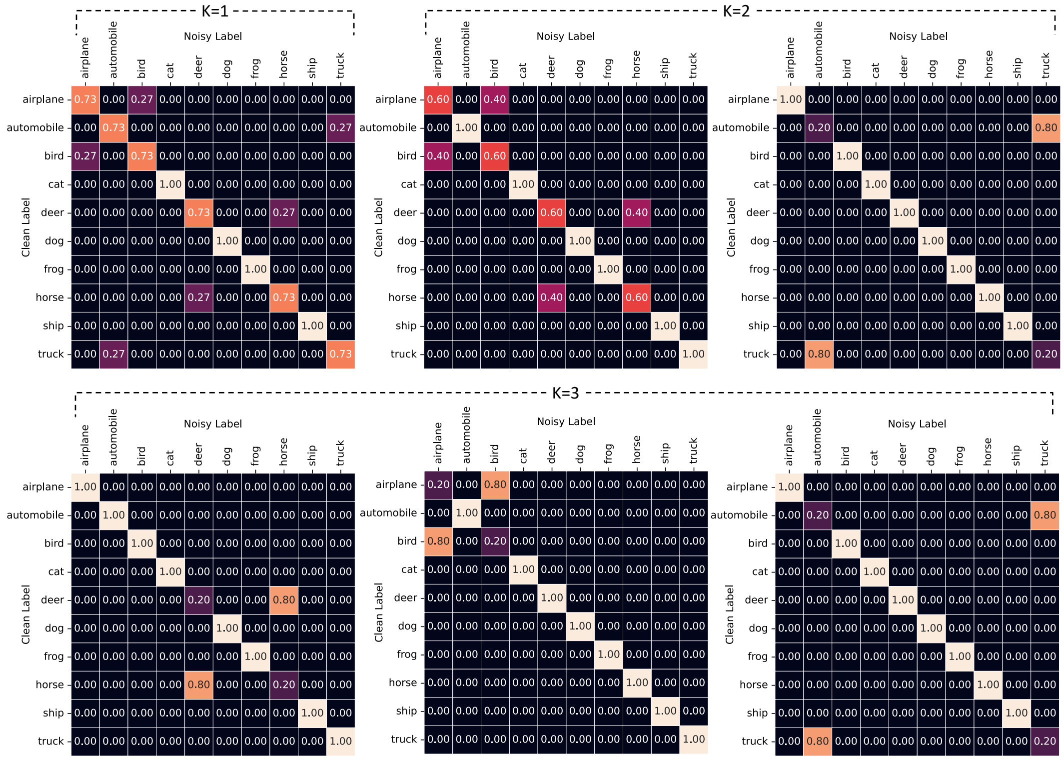

Figure 9: Estimated noise matrices for each profile when $K \in \{1, 2, 3\}$ from the simulation with CIFAR-10.

Figure 10: Estimated noise matrices for each profile when $K \in \{1, 2, 3\}$ from the simulation with AgNews.

H.2 Results of Datasets with Real Annotators

According to Table 14, all profiled users in every experiment have improved their accuracy with PHICO. Even considering all users, the method tends to improve the performance of the majority of users. Similarly to the simulated case, the number of improved users increases for the optimal K of the respective dataset (silhouette scores in Table 13). Table 15 shows that the accuracy after alterations for the profiled users in CIFAR-10N, CIFAR-10H, Fashion-MNIST-H and Chaoyang increase by least 18%, 5%, 30%, 6%, respectively. Table 16 shows that negative alterations for profiled users tend to decrease as $K > 1$. On CIFAR-10N and Fashion-MNIST-H positive alterations increase with K, but CIFAR-10H and Chaoyang show the opposite trend. Nevertheless, the accuracy for all datasets increases as a function of K, as shown in Table 15 because of the declining negative alterations. Table 17 shows that PHICO has increasing positive alteration rates compared to decreasing negative alteration rates as a function of K.

Table 14: Number of users who improved (I), maintained (M) and did not improve (NI).

Dataset	Users	K=1			K=2			K=3		
		I	S	NI	I	S	NI	I	S	NI
With simulated annotators										
CIFAR10	15	5	0	10	9	0	6	15	0	0
	15	5	0	10	9	0	6	15	0	0
AgNews	15	15	0	0	15	0	0	15	0	0
	15	15	0	0	15	0	0	15	0	0
With real annotators										
CIFAR10-N	80	80	0	0	80	0	0	80	0	0
	80	80	0	0	80	0	0	80	0	0
CIFAR10-H	2571	2548	0	23	2566	1	4	2567	1	3
	2022	2022	0	0	2022	0	0	2022	0	0
Fashion-MNIST-H	183	182	0	1	183	0	0	183	0	0
	182	182	0	0	182	0	0	182	0	0
Chaoyang	3	2	0	1	2	0	1	3	0	0
	2	2	0	0	2	0	0	2	0	0

Table 15: Initial accuracy vs the accuracy after alterations.

Dataset	Original Accuracy	Accuracy after Alterations		
		K=1	K=2	K=3
With simulated annotators				
CIFAR10	0.84001	0.83478	0.84500	0.87875
	0.84001	0.83478	0.84500	0.87875
AgNews	0.59976	0.93695	0.94974	0.98020
	0.59976	0.93695	0.94974	0.98020
With real annotators				
CIFAR10-N	0.83648	0.98775	0.98913	0.98915
	0.83648	0.98775	0.98913	0.98915
CIFAR10-H	0.94873	0.99184	0.99304	0.99318
	0.93999	0.99143	0.99260	0.99277
Fashion-MNIST-H	0.67226	0.86483	0.87849	0.87693
	0.66249	0.86432	0.87786	0.87636
Chaoyang	0.90270	0.91937	0.94123	0.94657
	0.85818	0.91500	0.92714	0.92374

Table 16: Alterations around optimal K

Dataset	K=1		K=2		K=3	
	A_+	A_-	A_+	A_-	A_+	A_-
With simulated annotators						
CIFAR10	0.8147	0.1614	0.8378	0.1536	0.9437	0.1336
AgNews	0.9028	0.0403	0.9357	0.0409	0.9748	0.0162
With real annotators						
CIFAR10-N	0.9528	0.0055	0.9541	0.0040	0.9542	0.0040
CIFAR10-H	0.9419	0.0055	0.9388	0.0041	0.9419	0.0041
FashionM-H	0.7352	0.0814	0.7581	0.0731	0.7544	0.074
Chaoyang	0.7943	0.0648	0.6862	0.0328	0.7377	0.0453

Table 17: Alteration rates around optimal K

Dataset	K=1		K=2		K=3	
	R_{A_+}	R_{A_-}	R_{A_+}	R_{A_-}	R_{A_+}	R_{A_-}
With simulated annotators						
CIFAR10	0.8347	0.1653	0.8451	0.1549	0.8759	0.1240
AgNews	0.9573	0.0427	0.9581	0.0419	0.9836	0.0164
With real annotators						
CIFAR10-N	0.9943	0.0057	0.9958	0.0042	0.9958	0.0042
CIFAR10-H	0.9942	0.0058	0.9956	0.0044	0.9956	0.0044
FashionM-H	0.9003	0.0997	0.9121	0.0879	0.9103	0.0897
Chaoyang	0.9246	0.0754	0.9543	0.0457	0.9422	0.0578

The effect of having values of K that are larger than its optimal and having more profiles was studied by extending the experiment done with CIFAR-10N dataset with VIT/B-16 base model. The results in Table 6 indicate that from $K = 1$ to $K = 3$, the accuracy increases and, for $K > 3$, it starts to decrease. Even though all testing users had their accuracy improved in all experiments, their accuracy gain has been slightly impacted by K. This demonstrates that having larger Ks beyond optimal silhouette score does not guarantee the best accuracy gain. Possibly, as K increases, the number of users per profile during training decreases, meaning that the augmented noisy labels may over personalise to the users' biases which may lead to a less generalisable model for testing users.

I Performance as a function of noise rate

The robustness of the approach for different noise rates was studied by extending the simulation with CIFAR-10 to different noise rates. The obtained results are reported in Table 9. An ImageNet pre-trained ResNet-18 was used as the backbone for the base model. The same simulation data preparation explained in Section 4.1 was followed here.

J THE ABLATION WITH DIFFERENT BACKBONE MODELS AS BASE MODEL

This experiment tests different backbones as the base model on CIFAR-10N dataset. The CIFAR-10N experiment follows the one described in Section 4.1 with a VIT/B-16 as the base model $f_{\psi_k}(.)$, and DenseNet-121 and Resnet-50 as $f_{\psi_k}(.)$.

The results in Table 8 showcases that different base models improve users in different degrees as accuracy after alterations is different among them. Yet, it consistently surpasses the original accuracy of users and all the profiled users were improved irrespectively of the base model.

It is important to emphasise that as the $f_{\psi_k}(.)$ changes, the consensus estimation in Section A changes. Following that, the number of users chosen for labelling at least 20 images from each class varies. This also changes the number of users in the test set and the recorded original accuracy in Table 8. To be specific, the experiments with ResNet-50 and DenseNet-121 were conducted respectively with 155 and 157 users identified for labelling 20 images per class. In the experiment with ResNet-50, 77 were in the training set and 78 were in the testing set. In the case with DenseNet-121, it was 78 and 79 in training and testing sets, respectively. The recorded results and user distribution for the experiment with ViT/B-16 are same as in the main paper.

Further, we extend the comparative analysis in Section 4.2 and use the two backbones with methods from literature to examine the performance. From the results in Table 7, our approach consistently outperforms the methods in literature.

K PERFORMANCE AS A FUNCTION OF NOISY LABEL AUGMENTATION G

The effect of the number of times G that noisy labels were augmented in profile $\hat{\mathcal{D}}_k$ is explored by extending the CIFAR-10N experiment with VIT/B-16. The results in Table 5 shows that larger G promotes a slight increase in the users' post alteration accuracy. Note that K was fixed at 2 for this experiment.

L TESTING λ IN THE LOSS FUNCTION

Here, we study how the second term in the loss function in equation 4 affects the post alteration accuracy. We conduct a range of experiments with $\lambda \in \{0, 0.01, 0.1, 1, 10\}$. Using CIFAR-10N dataset, three sets of experiments were conducted using ResNet-50, DenseNet-121 and Bit/B-16 as base models. Even though all users were improved in every experiment, the results in Table 12 show how post alteration accuracy vary with λ. It is clear that the highest post alteration accuracy is centered around $\lambda = 0.1$ for all 3 base models.