INTERPRETABLE CLIENT CONTRIBUTION EVALUATION IN FEDERATED LEARNING VIA PARTIAL INFORMATION DECOMPOSITION

Anonymous authors
Paper under double-blind review

ABSTRACT

In federated learning (FL), multiple clients collaboratively train a shared model without sharing their raw data. Understanding how much each client contributes to the global model is central to fair incentive design, principled client selection, and the detection of unreliable or harmful participants. Yet client contribution is typically summarised by a single scalar utility, which conflates fundamentally different sources of value: information that is unique to a client, information that is duplicated elsewhere in the federation, and information that emerges only when several clients are combined. We propose **FedPID**, an information-theoretic framework for client contribution evaluation based on Partial Information Decomposition (PID), characterising each client's value in terms of these three components rather than a single number. FedPID builds class-conditioned PIDs for each client from predictions on a shared server-side validation set, producing round-wise profiles of unique, redundant, and synergistic information that can be inspected directly as a diagnostic, used to compare clients, or converted into contribution-guided aggregation weights. Across heterogeneous client settings on CIFAR-10, Fed-ISIC, and ImageNet-100, FedPID recovers expected contribution orderings from first-round profiles, identifies specialist clients that other scalar methods undervalue, and separates useful clients from noisy ones. As one downstream use, PID-derived aggregation weights improve performance in noisy-client settings compared with baselines. FedPID thus reframes client contribution as an informational composition rather than a single opaque score, providing both an interpretable diagnostic and a control signal for aggregation.

1 INTRODUCTION

Federated learning (FL) enables multiple clients to train a shared model without centralising their raw data (McMahan et al., 2017). This paradigm is attractive in settings where data are naturally decentralised or sensitive. In any such collaboration, a basic question quickly arises: how much does each participant actually contribute to the shared model? Answering this question is fundamental: the lack of robust contribution evaluation methods is itself a barrier to the widespread adoption of FL, especially in high-stakes domains such as healthcare and finance (Jiang et al., 2023). In practice, contribution evaluation underpins incentive design and fair compensation (Ding et al., 2020; Li et al., 2023; Yu et al., 2020a; Shi et al., 2024), client selection and scheduling (Ding et al., 2022; Cho et al., 2022; Huang et al., 2023), contribution-aware aggregation in domain applications (Liu et al., 2022b), and the detection of harmful or free-riding participants whose updates should not dominate the global model (Fraboni et al., 2021; Lin et al., 2019; Wan et al., 2022).

Assessing contribution in FL is difficult because a client's value depends on what the rest of the federation provides. It may appear unimportant when its information is largely duplicated elsewhere or be underrated when its value only emerges when combined with others. Capturing this interdependence is therefore central to contribution in FL (Liu et al., 2022a; Song et al., 2019).

Most existing contribution-estimation methods collapse each client's contribution to a single scalar (Liu et al., 2022a; Wang et al., 2019; Yan et al., 2021; Zhao et al., 2021). Shapley value (Ghorbani & Zou, 2019; Jia et al., 2019; Shapley, 1953; Brânzei, 2010; Wang et al., 2020), leave-one-out scores

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(Wang et al., 2019), gradient-based contribution proxies (Jiang et al., 2023), and attention-style contribution weights (Yan et al., 2021) all provide useful summaries, but do not distinguish unique, redundant, and synergistic information. As a result, they may not only hide the structure of client contribution but also misestimate the client's total value. This limitation is especially important under heterogeneous federated settings, where unique, overlapping, complementary, and noisy client updates coexist, and where clients holding distinctive but minority signal can be systematically undervalued by scalar contribution metrics (Huang et al., 2023; Liu et al., 2024).

In this paper, we introduce an information-theoretic view of federated contribution evaluation based on Partial Information Decomposition (PID) (Bertschinger et al., 2014; Williams & Beer, 2010). PID decomposes predictive information into unique, redundant, and synergistic components, making it a natural tool for characterising how clients contribute to a global learning objective. Building on this decomposition, we propose FedPID, a framework that computes per-client PID profiles from server-side evaluations, either at selected rounds or accumulated across training. The resulting profiles can be inspected directly as a contribution diagnostic, used to compare clients on interpretable axes, or, as one downstream application, mapped to aggregation weights that steer the global update. In all cases, contribution measurement is kept inside the training loop and exposed as an interpretable profile rather than being reduced to a purely post-hoc scalar score.

Our contributions are as follows:

- We formulate client contribution evaluation in FL through Partial Information Decomposition, yielding per-client profiles that support better assessment of total client value through unique, redundant, and synergistic information rather than a single scalar score.
- We show that the decomposition can also be inspected at the level of individual class outcomes, revealing how unique, redundant, and synergistic information are distributed for each class rather than only on average across classes.
- As one downstream application, we map PID profiles to aggregation weights, allowing the same decomposition to act as both a diagnostic and a control signal during training.
- Across heterogeneous FL scenarios on CIFAR-10, Fed-ISIC, and ImageNet-100, we show that PID profiles recover expected contribution orderings from the first round, surface specialist clients that scalar baselines undervalue, and downweight noisy clients more decisively than competing methods.

2 Related Work

2.1 Client Contribution Evaluation in Federated Learning

Evaluating client contribution in FL has attracted sustained interest across incentive design (Lyu et al., 2020; Yu et al., 2020a;b; Zhan et al., 2020), auditing (Xue et al., 2021; Koh & Liang, 2017), and robust participation (Fraboni et al., 2021; Han & Zhang, 2020; Lin et al., 2019; Wan et al., 2022). The dominant line of work adopts the Shapley value from cooperative game theory and treats each client as a player whose contribution is its average marginal utility across coalitions. This offers a principled fairness interpretation, but exact computation becomes impractical as the number of clients grows. A large body of follow-up work targets the resulting computational cost, approximating Shapley-style procedures through gradient-based coalition reconstruction (Wei et al., 2020), reconstruction with guided truncation (Liu et al., 2022a), estimation from logged client updates (Wang et al., 2022), lightweight proxy estimators (Guo et al., 2025; Tang et al., 2025), and single-round approximations (Chen et al., 2023). A parallel line revises scoring rules for data heterogeneity (You et al., 2021; Guo et al., 2024; Liu et al., 2024), partial participation (Ghosh et al., 2025), and robust client assessment from local updates in Byzantine settings (Zhang et al., 2024).

Other approaches replace coalition reasoning with simpler proxies. Leave-one-out methods (Wang et al., 2019) estimate contribution through performance degradation after removing a client. Gradient-similarity methods (Jiang et al., 2023) compare a client's update against a reference direction, while attention-style methods (Yan et al., 2021) and reputation-aware schemes (Lyu et al., 2020; Zhang et al., 2021) derive client-specific weights from observed contribution or reliability signals. These estimators are cheaper and easier to integrate into large training pipelines, but, like Shapley-style methods, they ultimately collapse each client's contribution to a single score.

Some methods use scalar contribution estimates for adaptive aggregation or personalisation. ShapFed (Tastan et al., 2024) estimates overall and class-conditioned contributions and converts them into fairer aggregation and personalisation weights. Likewise, FedCon (Gao et al., 2025) couples contribution estimation with adaptive aggregation. FedTrans (Yang et al., 2024) estimates client utility from model parameters and performance on auxiliary data to guide robust selection, while Oort (Lai et al., 2021) uses data utility and device speed to guide participant selection. Our method, FedPID, also produces aggregation weights, but from an explicit decomposition into unique, redundant, and synergistic components rather than a single scalar utility, serving as a contribution diagnostic in its own right.

Existing methods share the same structural limitation: they quantify total usefulness, but not its informational composition. They do not reveal whether a client's value comes from distinctive information, information duplicated elsewhere in the federation, or information that only becomes useful jointly with other clients. This distinction is particularly important under non-IID heterogeneity, where specialist clients can be undervalued by scalar metrics (Huang et al., 2023; Liu et al., 2024).

2.2 Partial Information Decomposition in Machine Learning

Partial Information Decomposition refines mutual-information analysis in multi-source settings by separating predictive information into unique, redundant, and synergistic components (Bertschinger et al., 2014; Williams & Beer, 2010; Griffith & Koch, 2014; Harder et al., 2013; Lyu et al., 2024; Makkeh et al., 2021). Because redundancy can be defined in different ways, PID is best viewed as a general framework rather than a single unique decomposition. Recent machine learning work has used PID to analyse knowledge distillation (Dissanayake et al., 2024), interpretability in diffusion models (Dewan et al., 2024; Kong et al., 2024), and fairness trade-offs in FL (where Hamman & Dutta (2024) characterised tensions between local and global fairness rather than evaluating client contribution). In parallel, methodological advances by Kleinman et al. (2021) have made PID-related quantities more tractable in higher-dimensional settings.

To our knowledge, PID has not been used as the organising framework for client contribution evaluation in FL. Yet, it constitutes a natural fit for the gap left by scalar valuation methods: PID can expose redundancy among similar clients, uniqueness in clients carrying distinctive signal, and synergy when value emerges only jointly. This decomposition is the conceptual basis of FedPID.

3 Preliminaries

3.1 Federated Learning Setup

We consider a federated classification problem with K clients. Client k holds a private local dataset $\mathcal{D}_k = \{(x_{k,i}, y_{k,i})\}_{i=1}^{n_k}$ of size n_k, and $\theta \in \mathbb{R}^d$ denotes the model parameters. Training proceeds in rounds. At round t, the server selects a subset of clients $\mathcal{S}^{(t)} \subseteq \{1, \dots, K\}$, broadcasts the current global model $\theta^{(t)}$, and receives locally updated parameters $\theta_k^{(t+1)}$ from each participating client $k \in \mathcal{S}^{(t)}$. Standard dataset-size-weighted aggregation (McMahan et al., 2017) takes the form

$$\theta^{(t+1)} = \sum_{k \in \mathcal{S}^{(t)}} \frac{n_k}{\sum_{\ell \in \mathcal{S}^{(t)}} n_\ell} \theta_k^{(t+1)}. \tag{1}$$

This rule depends only on sample counts and therefore does not distinguish between clients that contribute distinctive predictive information and clients whose signal is largely duplicated or unreliable. To evaluate client contributions, the server additionally maintains a shared validation set $\mathcal{D}_{\text{val}} = \{(x_j, y_j)\}_{j=1}^{N_{\text{val}}}$. This follows the common server-validation protocol used by validation-based FL aggregation methods (Tang et al., 2025; Wen et al., 2025); the holdout set is used for contribution evaluation, is never shared with clients, and is disjoint from the final test set. After round t, the server evaluates each uploaded client model on $\mathcal{D}_{\text{val}}$ and records its predictive probabilities. For client $k \in \mathcal{S}^{(t)}$, sample x_j, and class $c \in \{0, \dots, C-1\}$, we write

$$p_{k,j,c}^{(t)} := \frac{\exp(f_c(x_j; \theta_k^{(t+1)}))}{\sum_{c'=0}^{C-1} \exp(f_{c'}(x_j; \theta_k^{(t+1)}))}, \qquad \mathbf{p}_{k,j}^{(t)} := \left[p_{k,j,0}^{(t)}, \dots, p_{k,j,C-1}^{(t)}\right], \tag{2}$$

where $f_c(x_j; \theta_k^{(t+1)})$ denotes the logit for class c. When needed, the corresponding hard prediction is $\hat{y}_{k,j}^{(t)} := \arg\max_c p_{k,j,c}^{(t)}$.

3.2 Partial Information Decomposition

For a target S and two sources R_1 and R_2, PID decomposes the joint information as

$$I(S; R_1, R_2) = \mathrm{Unq}(S; R_1) + \mathrm{Unq}(S; R_2) + \mathrm{Rdn}(S; R_1, R_2) + \mathrm{Syn}(S; R_1, R_2). \tag{3}$$

Here, $\mathrm{Unq}(S; R_1)$ and $\mathrm{Unq}(S; R_2)$ denote information about the target available only from one source, $\mathrm{Rdn}(S; R_1, R_2)$ denotes information shared by both sources, and $\mathrm{Syn}(S; R_1, R_2)$ denotes information available only when the sources are considered jointly. For multiple sources, the PID lattice grows combinatorially and includes terms for information redundant among some sources, unique to others, or synergistic within particular source collections. For interpretability and computational efficiency, we use the two-source Williams–Beer PID with the $I_{\min}$ redundancy function (Williams & Beer, 2010); Appendix A provides additional PID definitions, class-wise diagnostics, and estimator comparisons supporting this choice. In the method below, the target is derived from the ground-truth label, and the two sources are constructed from the predictive output of one client and a tuple-valued representation of the remaining clients.

4 Method

To build a client contribution profile, FedPID constructs a class-conditioned two-source PID estimator over predictions on the shared validation set. For each evaluated client k, the first source is the predictive state of client k, and the second source is an ordered tuple-valued representation of the remaining participating clients. To preserve the structure of soft predictions while retaining a discrete PID estimator, we discretise softmax probabilities into bins. We estimate PID separately for each class before aggregating across classes. The resulting per-client profiles are the primary output of FedPID: they can be inspected directly as a contribution diagnostic or, as a downstream use case, mapped to contribution-guided aggregation weights that determine the global update.

4.1 Class-Conditioned Soft-Probability Tuple PID

Let $\mathcal{S}^{(t)}$ denote the participating clients and let $\mathcal{D}_{\mathrm{val}} = \{(x_j, y_j)\}_{j=1}^{N_{\mathrm{val}}}$ be the shared server-side validation set at round t. Under this two-source reduction, before preprocessing, the PID tuple for evaluating client k on sample x_j is $\left(\mathbf{p}_{k,j}^{(t)},\ \left(\mathbf{p}_{\ell,j}^{(t)}\right)_{\ell \in \mathcal{S}^{(t)} \setminus \{k\}},\ y_j\right)$, which contains the current client (first source), the tuple of the remaining clients (second source), and the true label (target). The second source contains $(|\mathcal{S}^{(t)}| - 1)C$ continuous probability coordinates per validation sample, which makes reliable empirical distribution estimation difficult on a finite validation set. FedPID therefore preprocesses the predictive states before PID estimation.

Let $M \geq 1$ denote the number of interior bin edges. For chosen edges $\eta_1 < \cdots < \eta_M$ in $(0, 1)$, define $\eta_0 = 0$, $\eta_{M+1} = 1$, and $\mathcal{B} := (\eta_0, \eta_1, \ldots, \eta_{M+1})$, yielding $M + 1$ probability bins. The binning map $b_{\mathcal{B}} : [0, 1] \to \{0, \ldots, M\}$ assigns each probability to the index of its bin. We apply this map coordinate-wise to the softmax outputs. For client k, sample x_j, and class c, the binned predictive coordinate is $z_{k,j,c}^{(t)} := b_{\mathcal{B}}\left(p_{k,j,c}^{(t)}\right)$. This preprocessing changes each softmax coordinate from a continuous value to a finite symbol in $\{0, \ldots, M\}$. This discretisation controls a sample-efficiency trade-off: coarser bins give denser empirical distributions but remove confidence detail, while finer bins preserve more predictive structure at the cost of sparser joint distributions. We therefore treat the bin count as an estimator choice and evaluate its sensitivity in Appendix D.5.

For each class $c \in \{0, \ldots, C - 1\}$, FedPID measures class-conditioned information by processing the true validation label into the one-vs-rest target $y_j^{(c)} := \mathbf{1}[y_j = c]$. This aligns the target with the binned source coordinate: $z_{k,j,c}^{(t)}$ represents client k's confidence in class c, so the corresponding target is the binary event that the true label is class c. The reduction is statistically useful as well, since a multiclass target makes PID estimation noisier on a finite validation set. The class-conditioned PID then pairs this target with the binned source coordinate $z_{k,j,c}^{(t)}$, so that source and target both describe the one-vs-rest question: client k's confidence about class c predicting whether the true label is class c. Equivalently, we write $\mathbf{z}_{-k,j,c}^{(t)}$ for the ordered tuple containing $z_{\ell,j,c}^{(t)}$ for all other participating clients $\ell \in \mathcal{S}^{(t)} \setminus \{k\}$. Thus the empirical PID for class c is estimated from $\left(z_{k,j,c}^{(t)}, \mathbf{z}_{-k,j,c}^{(t)}, y_j^{(c)}\right)$.

For client k and class c, we estimate the empirical joint distribution

$$\hat{P}^{(t)}_{k,c}(a, \mathbf{b}, u) = \frac{1}{N_{\text{val}}} \sum_{j=1}^{N_{\text{val}}} \mathbf{1}\Big[z^{(t)}_{k,j,c} = a,\, \mathbf{z}^{(t)}_{-k,j,c} = \mathbf{b},\, y^{(c)}_j = u\Big]. \tag{4}$$

Let $Z^{(c)}_k$, $Z^{(c)}_{-k}$, and $Y^{(c)}$ denote the discrete random variables induced by this empirical distribution. Applying the two-source PID yields

$$I\Big(Y^{(c)}; Z^{(c)}_k, Z^{(c)}_{-k}\Big) = \mathrm{Unq}^{(t)}_{k,c} + \mathrm{Unq}^{(t)}_{-k,c} + \mathrm{Rdn}^{(t)}_{k,c} + \mathrm{Syn}^{(t)}_{k,c}, \tag{5}$$

where $\mathrm{Unq}^{(t)}_{k,c}$ is the information about class c uniquely provided by client k, $\mathrm{Unq}^{(t)}_{-k,c}$ is the information uniquely provided by the tuple-valued source, $\mathrm{Rdn}^{(t)}_{k,c}$ is redundant information, and $\mathrm{Syn}^{(t)}_{k,c}$ is synergistic information. We retain $\mathrm{Unq}^{(t)}_{k,c}$, $\mathrm{Rdn}^{(t)}_{k,c}$, and $\mathrm{Syn}^{(t)}_{k,c}$ as the client-centred quantities, omitting only $\mathrm{Unq}^{(t)}_{-k,c}$ because it describes information attributable exclusively to the peer set.

For each client k, the two-source construction evaluates its informational contribution relative to the participating peers collectively. The ordered tuple preserves their joint binned predictive state, with synergy measured between the client being evaluated and the peer set rather than localised to arbitrary peer subsets. This choice distinguishes what the client contributes uniquely, shares with its peers, and provides only jointly with them. In the Williams–Beer lattice, two, three, four, and five sources yield 4, 18, 166, and 7,579 PID atoms, respectively (Williams & Beer, 2010). Three sources already include mixed terms such as $\{1\}\{23\}$, information available from source 1 alone and also from sources 2 and 3 jointly, and $\{12\}\{13\}\{23\}$, redundancy shared among pairwise source groups (see Appendix A.1). These atoms do not map directly to a single client's contribution. Using them for client scoring therefore requires additional choices about which atoms to retain, discard, allocate or group, and how to weight them. With our three retained bivariate components, fixing the unique-information coefficient to one leaves only the redundancy and synergy coefficients to tune for contribution-guided aggregation (Section 4.3).

Replacing the binned class probabilities with argmax labels and omitting one-vs-rest target binarisation reduces the construction to a hard-label tuple PID. This discards the within-class confidence structure that distinguishes clients of similar accuracy; Appendix A.5 illustrates how using hard labels alone fails to localise specialist information.

4.2 From Class-Conditioned PID to Whole-Client Profiles

The class-conditioned decompositions above yield one PID per class. To obtain a single round-wise profile for client k, we aggregate these class-level terms using the empirical class prior on the shared server-side validation set $\pi_c := \frac{1}{N_{\text{val}}} \sum_{j=1}^{N_{\text{val}}} \mathbf{1}[y_j = c]$. We then define the whole-client PID tuple,

$$\big(\mathrm{Unq}^{(t)}_k, \mathrm{Unq}^{(t)}_{-k}, \mathrm{Rdn}^{(t)}_k, \mathrm{Syn}^{(t)}_k\big) = \sum_{c=0}^{C-1} \pi_c \big(\mathrm{Unq}^{(t)}_{k,c}, \mathrm{Unq}^{(t)}_{-k,c}, \mathrm{Rdn}^{(t)}_{k,c}, \mathrm{Syn}^{(t)}_{k,c}\big). \tag{6}$$

The resulting client-centred profile $\phi^{(t)}_k := (\mathrm{Unq}^{(t)}_k, \mathrm{Rdn}^{(t)}_k, \mathrm{Syn}^{(t)}_k)$ summarises client k's unique, redundant, and synergistic information at round t. We exclude $\mathrm{Unq}^{(t)}_{-k}$ because it is attributable exclusively to the peer set. The summary $\phi^{(t)}_k$ can be compared across clients and across rounds, while the class-conditioned structure can still be accessed for inspection.

4.3 From PID Profiles to Contribution-Guided Aggregation

The PID profile assigns each client an interpretable, multi-axis description of what it contributes to the target. A natural downstream use is to weight the contribution of each client during aggregation: if a client carries informative predictive signal not already available from the rest of the federation, its update should weigh more heavily in the global model than that of a client whose signal is redundant, noisy, or absent. To convert the per-client profile into a scalar that can drive such contribution-guided aggregation, we define the round-wise client score $s^{(t)}_k = \alpha \mathrm{Unq}^{(t)}_k + \beta \mathrm{Rdn}^{(t)}_k + \gamma \mathrm{Syn}^{(t)}_k$, where α, β, and γ control the relative importance of unique, redundant, and synergistic information.

Because client profiles may change over time, FedPID maintains an accumulated score $A_k^{(t)}$ for each client across PID measurement rounds. On a PID measurement round, FedPID updates $A_k^{(t)} \leftarrow A_k^{(t-1)} + s_k^{(t)}$ for each participating client $k \in \mathcal{S}^{(t)}$; non-participating clients keep their previous score. At aggregation, the server normalises $A_k^{(t)}$ over the clients participating in that round ($\mathcal{S}^{(t)}$) to obtain weights $w_k^{(t)}$, which are used to average the uploaded client parameters (see more in Appendix C). Thus, FedPID can be applied under partial participation: at each round, the tuple source, client scores, and aggregation weights are all computed using only the current participant set.

4.4 Practical Protocol

PID need not be evaluated at every communication round: between measurements, the server reuses the latest accumulated scores for aggregation. This reduces computation while keeping the diagnostic profile $\phi_k^{(t)}$ separate from the aggregation weight $w_k^{(t)}$. We use three schedules: FedPID$_F$ computes the PID scores at the first round and reuses them for all subsequent contribution-guided aggregation, FedPID$_L$ reports the last-round PID profile for diagnostic comparison, and FedPID$_R$ recomputes PID every round and accumulates the resulting scores. In order to compare with scalar contribution baselines, we combine the PID terms using fixed $(\alpha, \beta, \gamma) = (1, 0.15, 0.15)$ to favour unique information. For aggregation, we select (α, β, γ) by validation performance in Appendix D.5.

5 Experiments

We evaluate FedPID along three axes: whether PID profiles characterise client value in interpretable terms, whether the resulting scalar scores agree with or improve upon standard contribution baselines, and whether PID-guided aggregation improves performance across heterogeneous client settings.

5.1 Experimental Setup

Datasets and Implementation Details. Experiments use CIFAR-10 (Krizhevsky, 2009), Fed-ISIC (Ogier du Terrail et al., 2022), and ImageNet-100 (Deng et al., 2009). CIFAR-10 and Fed-ISIC are reported in the main text; ImageNet-100 is used only for first-round diagnostics in Appendix D, alongside additional CIFAR-10 contribution results and scalability/robustness analyses. Implementation details are provided in Appendix C. The main-text experiments use full client participation to isolate contribution-evaluation behaviour. Appendix D.3.1 extends evaluation to larger client pools. We reserve a clean 30% class-stratified split of the official test set as $\mathcal{D}_{\text{val}}$ for contribution evaluation and hyperparameter selection, and use the remaining samples for final evaluation. Validation-set stress tests are reported in Appendix D.3.2. For contribution evaluation, unless otherwise stated, we set $M = 3$ interior bin edges, yielding four probability bins, and use $(\alpha, \beta, \gamma) = (1, 0.15, 0.15)$ from Section 4.4. Bin-count and coefficient tuning for aggregation are reported in Appendix D.5.

Dataset Design. On CIFAR-10, we consider IID, quantity-skew, noisy-client, and non-IID settings. For noisy-client experiments, client 1 remains clean and clients $k = 2, \ldots, 5$ receive progressively stronger Gaussian feature noise, label flips, or both (see Appendix C for exact corruption levels). The non-IID setting consists of a specialist client 1 (receiving only samples from classes 0-1) and four other clients with overlapping support (sharing classes 2–9). This tests whether contribution methods recognise unique class support rather than only global utility. We also evaluate free-rider (a client not performing local training) and data-inflation (a client reusing its local samples) settings. Full details for all the above are in Appendix B. On Fed-ISIC, we evaluate the clean benchmark and two targeted corruption variants (Gaussian feature noise or random label flips) affecting only the largest centre.

Compared Methods. For contribution evaluation, we compare FedPID with exact Shapley value (SV), leave-one-out (LOO) (Wang et al., 2019), GTG-Shapley (GTG) (Liu et al., 2022a), and gradient-cosine similarity (GCS: cosine similarity between each client update and the aggregated update), reporting the FedPID$_F$ and FedPID$_L$ variants. For aggregation, we compare FedPID$_F$ and FedPID$_R$ against FedAvg (McMahan et al., 2017), FedProx (Li et al., 2020), Trimmed Mean (Yin et al., 2018), Coordinate-wise Median (Yin et al., 2018), Krum (Blanchard et al., 2017), and validation-based weighting using accuracy or loss on $\mathcal{D}_{\text{val}}$. Balanced-accuracy weighting is reported separately for class-imbalanced validation in Appendix D.3.3.

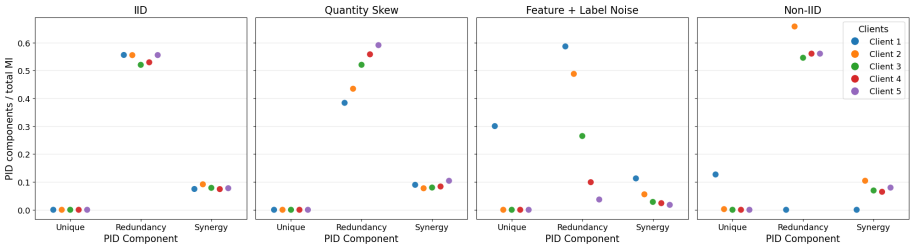

Figure 1: Normalised PID components at the first round across settings on CIFAR-10.

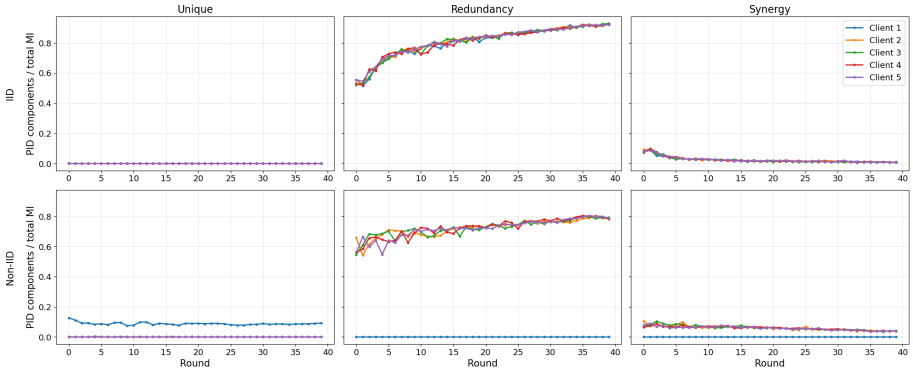

Figure 2: Normalised PID components across both IID and non-IID settings on CIFAR-10.

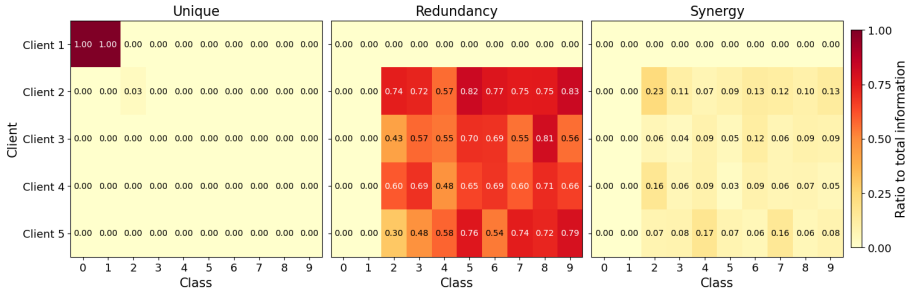

Figure 3: Class-wise normalised PID components for non-IID setting on CIFAR-10 at the first round.

5.2 PID AS A DIAGNOSTIC FOR FL

PID Profiles. Figure 1 shows each client's normalised first-round PID components across four CIFAR-10 settings, while Figure 2 tracks how the same components evolve across all rounds in IID and non-IID settings. These profiles are computed directly from the PID decomposition and therefore do not depend on the aggregation coefficients. Differences between clients' PID profiles are already visible in the first round under heterogeneity. Thus, FedPID$_F$ is the natural choice for early contribution evaluation before aggregation smooths client differences, while FedPID$_L$ is more appropriate for a post-training diagnostic of the final client models.

Under IID settings, all clients exhibit nearly identical profiles: unique information remains zero, while redundancy increases and synergy decreases over training. Under quantity skew, the decomposition distinguishes clients based on their redundancy component, reflecting the larger shared predictive signal of clients with more data. Under feature-and-label noise, PID separates clients by data quality: the clean client 1 displays substantial useful information, whereas progressively corrupted clients contribute less. The clean client is also identified as the only one with unique information. Under the non-IID setting, client 1 is the only client containing classes 0 and 1, so it has predictive information not shared with the other clients. Accordingly, FedPID attributes all its contribution to the unique

Table 1: Client contribution scores across settings on CIFAR-10. Expected contribution patterns are near-uniform in IID, decreasing for quantity skew and feature-and-label noise; in the non-IID setting, C1 is the specialist client. Error bars for FedPID are reported in Table 6.

	IID					Quantity Skew					Feature + Label Noise					Non-IID				
Method	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5
SV	0.163	0.167	0.164	0.168	0.166	0.184	0.180	0.169	0.159	0.143	0.351	0.162	0.022	-0.067	-0.168	-0.005	0.153	0.149	0.146	0.145
LOO	0.004	0.004	0.003	0.011	0.004	0.015	0.014	0.008	0.008	0.005	0.187	0.058	0.006	-0.049	-0.168	-0.031	-0.004	-0.005	-0.012	-0.005
GTG	0.127	0.158	0.129	0.188	0.174	0.173	0.190	0.187	0.139	0.099	0.437	0.266	0.103	-0.149	-0.575	-0.676	0.248	0.235	0.223	0.233
GCS	0.199	0.200	0.200	0.199	0.201	0.327	0.258	0.195	0.135	0.085	0.172	0.206	0.211	0.213	0.199	0.119	0.219	0.221	0.225	0.216
FedPID$_F$	0.197	0.203	0.200	0.195	0.205	0.236	0.216	0.198	0.184	0.166	0.736	0.141	0.079	0.033	0.011	0.233	0.225	0.178	0.163	0.200
FedPID$_L$	0.199	0.201	0.201	0.200	0.200	0.202	0.202	0.200	0.199	0.197	0.652	0.191	0.101	0.037	0.019	0.159	0.211	0.210	0.210	0.209

Table 2: Client contribution scores across settings on Fed-ISIC. Error bars are reported in Table 7.

	Clean						Feature Noise						Label Noise					
Method	C1	C2	C3	C4	C5	C6	C1	C2	C3	C4	C5	C6	C1	C2	C3	C4	C5	C6
SV	0.353	0.058	0.062	0.076	0.008	0.012	-0.127	0.063	0.058	0.096	0.024	0.014	0.084	0.055	0.093	0.101	0.027	0.006
LOO	0.263	0.069	-0.012	0.037	-0.007	-0.022	-0.187	0.0423	-0.092	0.082	0.077	0.013	0.071	0.001	0.046	0.081	0.055	-0.034
GTG	0.426	-0.020	0.016	0.039	-0.028	0.012	-0.104	0.060	0.110	0.073	0.002	0.004	0.014	0.052	0.148	0.105	-0.032	0.018
GCS	0.644	0.093	0.146	0.087	0.016	0.014	0.678	0.084	0.134	0.078	0.014	0.010	0.688	0.089	0.119	0.073	0.014	0.015
FedPID$_F$	0.258	0.140	0.218	0.184	0.108	0.093	0.046	0.154	0.288	0.234	0.122	0.156	0.183	0.096	0.259	0.221	0.116	0.124
FedPID$_L$	0.257	0.123	0.172	0.175	0.122	0.151	0.069	0.149	0.247	0.266	0.134	0.136	0.150	0.156	0.235	0.204	0.110	0.145

component, and Figure 2 shows that this component persists across all rounds rather than disappearing later in training. Thus, FedPID correctly identifies client 1 as a specialist.

To localise the unique contribution from client 1, Figure 3 reports class-wise normalised PID components for the non-IID setting at the first round: each cell is the fraction of class-conditioned joint mutual information assigned to a PID component for one client and class. The unique information of client 1 concentrates on classes 0 and 1, exactly where its local data are exclusive; for the remaining classes, contribution is mainly redundant or synergistic across clients 2–5. The redundant and synergistic terms for clients 2–5 vary in this first-round diagnostic due to finite validation data and a single round of training; Appendix D.1 shows a more stable last-round profile.

Contribution-Score Results. Tables 1 and 2 report client contribution scores, with FedPID profiles collapsed using the fixed weights described in Section 5.1. Because the methods use different scales, we compare client orderings within each method. In the IID setting on CIFAR-10, exact Shapley (SV), GCS, and FedPID are close to uniform, while LOO remains small for all clients and GTG-Shapley shows mild variation. Under quantity skew, SV, GCS, and FedPID scores measured at the first round recover the data-size ordering. Under feature-and-label noise, FedPID assigns the largest contribution to the clean client and sharply downweights progressively corrupted clients; in contrast, GCS remains nearly uniform and even ranks corrupted clients above the clean client. The most notable comparison is for the non-IID setting. Client 1 uniquely holds classes 0 and 1 and is therefore the most informationally distinctive participant in the federation, yet exact Shapley assigns it near-zero value, LOO and GTG-Shapley assign it negative scores, and GCS ranks it lowest. The cause is structural: scalar contribution methods score clients by marginal performance impact or alignment with the global update, both of which can obscure specialist predictive information. Instead FedPID scores measured at the first round attribute the value of client 1 to its unique-information component and thus correctly reveal its contribution. Further contribution experiments with five-client Dirichlet partitions ($\alpha = 0.5$) are reported in Appendix D.2.1.

FedPID also detects clients whose apparent participation does not reflect their predictive information: free-rider clients receive practically null scores ($\sim$0.000) and data-inflation clients receive scores roughly 5x lower than genuine clients, because PID measures predictive information about the target (Appendix D.2). On Fed-ISIC (Table 2), FedPID assigns to C1 the largest score in the clean setting, consistent with it being the largest and most class-diverse centre. When C1 is corrupted, FedPID reduces its score and shifts contribution toward other clean centres, whereas GCS continues to rank C1 highest under both corruptions. Appendix D.3 reports 10-client CIFAR-10 runtime and contribution results: FedPID is faster than exact Shapley (6,299s vs 1,708,123s) and GTG-Shapley (11,369s), while retaining decomposition information. Experiments with $K = 20$ and 50 further support FedPID's use beyond fully participating cross-silo FL (Appendix D.3.1).

Table 3: Test accuracy (%) of PID-based aggregation and baseline aggregation rules on CIFAR-10 with a 30% class-balanced server validation set. Values are reported as mean ± standard deviation over three seeds.

Method	IID	Quantity Skew	Non-IID	Feature + Label Noise
FedAvg	92.29 ± 0.12	92.46 ± 0.41	69.26 ± 0.55	53.40 ± 1.88
FedProx	90.90 ± 0.10	90.62 ± 0.16	69.27 ± 0.45	62.65 ± 0.40
Trimmed Mean	91.98 ± 0.24	92.07 ± 0.26	70.98 ± 0.17	54.36 ± 0.64
Coordinate-wise Median	92.38 ± 0.34	92.40 ± 0.14	70.89 ± 0.20	57.15 ± 1.41
Validation-accuracy Weighting	92.26 ± 0.18	92.68 ± 0.08	71.17 ± 0.04	75.01 ± 0.53
Validation-loss Weighting	$\mathbf{92.45 \pm 0.19}$	92.58 ± 0.30	70.88 ± 0.34	76.47 ± 0.95
Krum	90.10 ± 0.18	83.91 ± 0.78	68.89 ± 0.88	46.80 ± 1.51
FedPID$_F$	92.15 ± 0.22	$\mathbf{92.83 \pm 0.39}$	71.04 ± 0.14	$\mathbf{84.75 \pm 0.52}$
FedPID$_R$	92.30 ± 0.27	92.72 ± 0.15	$\mathbf{71.46 \pm 0.11}$	81.99 ± 0.36

5.3 PID-BASED AGGREGATION

We evaluate whether PID profiles can serve as a control signal for aggregation. We select FedPID's aggregation coefficients using validation performance, without consulting final-test accuracy. We compare against standard baselines and two validation-based weighting methods, which use the same $\mathcal{D}_{\text{val}}$ as FedPID. These validation-based methods assess each client model in isolation, whereas FedPID measures how the client's predictive state relates to the target jointly with the rest of the federation. For aggregation, FedPID$_F$ is the cheaper default when client quality is stable or corrupted from the start, while FedPID$_R$ is preferable when the weighting signal is expected to evolve across rounds.

Table 3 shows similar test accuracies across most methods under IID. Under quantity skew, validation-accuracy weighting, FedPID$_F$, and FedPID$_R$ improve slightly over FedAvg, with FedPID$_F$ achieving the best mean accuracy. Under non-IID, FedPID$_R$ obtains the highest mean accuracy, followed by validation-accuracy weighting. The largest separation occurs under feature-and-label noise. Validation-loss weighting reaches 76.47%, compared with 75.01% for validation-accuracy weighting. FedPID$_F$ reaches 84.75%, exceeding the strongest validation-based baseline by 8.3 percentage points and FedAvg (53.40%) by 31 points. This gap suggests that PID captures useful predictive information beyond standalone validation quality. The advantage of FedPID$_F$ is consistent with first-round scores preserving a sharper estimate of intrinsic client quality before noisy clients benefit from the federation. Appendix D.3.3 reports further comparisons with smaller and class-imbalanced validation sets.

6 LIMITATIONS

FedPID evaluates each client relative to its participating peers collectively, without resolving interactions among arbitrary client subsets. It requires labelled server-side validation data, although contribution evaluation remains robust to smaller sets, class imbalance and random label corruption in the tested validation set settings (Appendix D.3.2). Discretisation trades confidence detail for denser empirical counts, but peer-tuple sparsity remains a finite-sample limitation as the active set grows. Informational contribution does not imply a corresponding gain from aggregation, whose scalarisation coefficients are setting-dependent. PID scoring adds validation and estimation cost; FedPID$_F$ reduces this by using first-round scores. Experiments with larger client pools support use beyond fully participating cross-silo FL, but scalability to much larger client pools remains untested.

7 CONCLUSION

We introduced FedPID, a PID-based framework that decomposes client contribution into unique, redundant, and synergistic predictive information relative to the federation. Its class-conditioned soft-probability tuples make these diagnostics practical within training. Across heterogeneous settings, FedPID reveals informative client ordering from the first round, localises specialist value through class-wise unique information, and exposes contribution structure that can also be converted into a scalar score for contribution-guided weighting of client updates, providing finer-grained diagnostics than scalar utilities and interpretable scores for contribution-guided aggregation. Richer multi-source and continuous PID estimators remain natural future directions.

AI USE STATEMENT

The research ideas, method design, equations and experimental results are the authors' own original work. We used generative AI to assist with grammar correction and language polishing and formatting. The authors are responsible for the final content.

ETHICS STATEMENT

This work studies client contribution evaluation using CIFAR-10, Fed-ISIC and ImageNet-100. FedPID scores measure predictive information relative to the chosen validation data and participating clients. Applications that use these scores to allocate rewards or select clients should consider whether the validation data represent the populations affected by those decisions.

REPRODUCIBILITY STATEMENT

Section 5.1 describes the experimental protocol and datasets. Appendix B specifies the client partitions and corruption procedures, while Appendix C gives the data splits, model architectures, training schedules, baseline settings and PID implementation details. Appendix D.5 reports the bin-count and coefficient studies. The PID definitions and outcome-specific derivation are provided in Appendix A.

REFERENCES

Nils Bertschinger, Johannes Rauh, Eckehard Olbrich, Jürgen Jost, and Nihat Ay. Quantifying unique information. *Entropy*, 16(4):2161–2183, April 2014. ISSN 1099-4300. doi: 10.3390/e16042161. URL http://arxiv.org/abs/1311.2852. arXiv:1311.2852 [cs].

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017.

Simina Brânzei. State of the art: Solution concepts for coalitional games. *Brain. Broad Research in Artificial Intelligence and Neuroscience*, 1(2):89–101, 04 2010.

Yi-Chung Chen, Hsi-Wen Chen, Shun-Gui Wang, and Ming-syan Chen. Space: Single-round participant amalgamation for contribution evaluation in federated learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 6422–6441. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/14a812fa4b6bf244d055e37a7cd2f557-Paper-Conference.pdf.

Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Towards understanding biased client selection in federated learning. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 10351–10375. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/jee-cho22a.html.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Shaurya Dewan, Rushikesh Zawar, Prakanshul Saxena, Yingshan Chang, Andrew Luo, and Yonatan Bisk. Diffusion PID: Interpreting diffusion via partial information decomposition. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 2045–2079. Curran Associates, Inc., 2024. doi: 10.52202/079017-0066. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/03d113a060c0ac93a5859517a0f07271-Paper-Conference.pdf.

Guanyu Ding, Zhengxiong Li, Yusen Wu, Xiaokun Yang, Mehrdad Aliasgari, and Hailu Xu. Towards an efficient client selection system for federated learning. In Kejiang Ye and Liang-Jie Zhang (eds.), *Cloud Computing – CLOUD 2022*, pp. 13–21, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-23498-9. doi: 10.1007/978-3-031-23498-9_2.

Ningning Ding, Zhixuan Fang, and Jianwei Huang. Incentive mechanism design for federated learning with multi-dimensional private information. In *2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*, pp. 33–40, 2020. URL https://opendl.ifip-tc6.org/db/conf/wiopt/wiopt2020/index.html.

Pasan Dissanayake, Faisal Hamman, Barproda Halder, Ilia Sucholutsky, Qiuyi Zhang, and Sanghamitra Dutta. Quantifying knowledge distillation using partial information decomposition. *ArXiv*, abs/2411.07483, 2024. URL https://api.semanticscholar.org/CorpusID:273969804.

Yann Fraboni, Richard Vidal, and Marco Lorenzi. Free-rider attacks on model aggregation in federated learning. In Arindam Banerjee and Kenji Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 1846–1854. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/fraboni21a.html.

Wenyu Gao, Gaochao Xu, and Xianqiu Meng. FedCon: Scalable and Efficient Federated Learning via Contribution-Based Aggregation. *Electronics*, 14(5):1024, March 2025. ISSN 2079-9292. doi: 10.3390/electronics14051024. URL https://www.mdpi.com/2079-9292/14/5/1024.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning*, pp. 2242–2251, 2019.

Bishwamittra Ghosh, Debabrota Basu, Huazhu Fu, Yuan Wang, Renuga Kanagavelu, Jin Peng Jiang, Yong Liu, Rick Siow Mong Goh, and Qingsong Wei. History-Aware and Dynamic Client Contribution in Federated Learning. In *28th European Conference on Artificial Intelligence (ECAI)*, pp. 1374–1381, Bologna, Italy, October 2025. doi: 10.3233/FAIA250955. URL https://doi.org/10.3233/FAIA250955.

Virgil Griffith and Christof Koch. Quantifying synergistic mutual information. In Mikhail Prokopenko (ed.), *Guided Self-Organization: Inception*, pp. 159–190. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. ISBN 978-3-642-53734-9. doi: 10.1007/978-3-642-53734-9_6. URL https://doi.org/10.1007/978-3-642-53734-9_6.

Jingjing Guo, Jiaxing Li, Zhiquan Liu, Yupeng Xiong, Yong Ma, Athanasios V. Vasilakos, Xinghua Li, and Jianfeng Ma. LCEFL: A lightweight contribution evaluation approach for federated learning. *IEEE Transactions on Mobile Computing*, 24(7):6643–6657, 2025. doi: 10.1109/TMC.2025.3545140.

Peng Guo, Yanqing Yang, Wei Guo, and Yanping Shen. A Fair Contribution Measurement Method for Federated Learning. *Sensors*, 24(15):4967, July 2024. ISSN 1424-8220. doi: 10.3390/s24154967. URL https://www.mdpi.com/1424-8220/24/15/4967.

Faisal Hamman and Sanghamitra Dutta. Demystifying local & global fairness trade-offs in federated learning using partial information decomposition. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=SBj2Qdhgew.

Yufei Han and Xiangliang Zhang. Robust federated learning via collaborative machine teaching. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):4075–4082, Apr. 2020. doi: 10.1609/aaai.v34i04.5826. URL https://ojs.aaai.org/index.php/AAAI/article/view/5826.

Malte Harder, Christoph Salge, and Daniel Polani. Bivariate measure of redundant information. *Physical Review E*, 87(1):012130, 2013. ISSN 1539-3755. doi: 10.1103/PhysRevE.87.012130.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '16, pp. 770–778. IEEE, June 2016. doi: 10.1109/CVPR.2016.90. URL http://ieeexplore.ieee.org/document/7780459.

Jiyue Huang, Chi Hong, Yang Liu, Lydia Y. Chen, and Stefanie Roos. Maverick matters: Client contribution and selection in federated learning. In Hisashi Kashima, Tsuyoshi Ide, and Wen-Chih Peng (eds.), *Advances in Knowledge Discovery and Data Mining*, pp. 269–282, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-33377-4.

Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J. Spanos. Towards efficient data valuation based on the shapley value. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1167–1176. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/jia19a.html.

Meirui Jiang, Holger R. Roth, Wenqi Li, Dong Yang, Can Zhao, Vishwesh Nath, Daguang Xu, Qi Dou, and Ziyue Xu. Fair federated medical image segmentation via client contribution estimation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16302–16311, 2023. doi: 10.1109/CVPR52729.2023.01564.

Michael Kleinman, Alessandro Achille, Stefano Soatto, and Jonathan C. Kao. Redundant information neural estimation. *Entropy*, 23(7):922, 2021. ISSN 1099-4300. doi: 10.3390/e23070922. URL https://www.mdpi.com/1099-4300/23/7/922.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/koh17a.html.

Xianghao Kong, Ollie Liu, Han Li, Dani Yogatama, and Greg Ver Steeg. Interpretable diffusion via information decomposition. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=X6tNkN6ate.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Fan Lai, Xiangfeng Zhu, Harsha V. Madhyastha, and Mosharaf Chowdhury. Oort: Efficient federated learning via guided participant selection. In *15th USENIX Symposium on Operating Systems Design and Implementation (OSDI 21)*, pp. 19–35. USENIX Association, July 2021. ISBN 978-1-939133-22-9. URL https://www.usenix.org/conference/osdi21/presentation/lai.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

Ying Li, Xingwei Wang, Rongfei Zeng, Mingzhou Yang, Kexin Li, Min Huang, and Schahram Dustdar. Varf: An incentive mechanism of cross-silo federated learning in mec. *IEEE Internet of Things Journal*, 10(17): 15115–15132, 2023. doi: 10.1109/JIOT.2023.3264611.

Jierui Lin, Min Du, and Jian Liu. Free-riders in federated learning: Attacks and defenses, 2019. URL https://arxiv.org/abs/1911.12560.

Yiqi Liu, Shan Chang, Ye Liu, Bo Li, and Cong Wang. Fairfed: Improving fairness and efficiency of contribution evaluation in federated learning via cooperative shapley value. In *IEEE INFOCOM 2024 - IEEE Conference on Computer Communications*, pp. 621–630, 2024. doi: 10.1109/INFOCOM52122.2024.10621438.

Zelei Liu, Yuanyuan Chen, Han Yu, Yang Liu, and Lizhen Cui. GTG-Shapley: Efficient and Accurate Participant Contribution Evaluation in Federated Learning. *ACM Transactions on Intelligent Systems and Technology*, 13(4):60:1–60:21, August 2022a. ISSN 2157-6904, 2157-6912. doi: 10.1145/3501811. URL https://dl.acm.org/doi/10.1145/3501811.

Zelei Liu, Yuanyuan Chen, Yansong Zhao, Han Yu, Yang Liu, Renyi Bao, Jinpeng Jiang, Zaiqing Nie, Qian Xu, and Qiang Yang. Contribution-aware federated learning for smart healthcare. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(11):12396–12404, 2022b. doi: 10.1609/aaai.v36i11.21505.

Aobo Lyu, Andrew Clark, and Netanel Raviv. Explicit formula for partial information decomposition. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 2329–2334, 2024. doi: 10.1109/ISIT57864.2024.10619369.

Lingjuan Lyu, Xinyi Xu, Qian Wang, and Han Yu. Collaborative fairness in federated learning. In *Federated Learning: Privacy and Incentive*, pp. 189–204. Springer International Publishing, Cham, 2020.

Abdullah Makkeh, Aaron J. Gutknecht, and Michael Wibral. Introducing a differentiable measure of pointwise shared information. *Phys. Rev. E*, 103(3):032149, Mar 2021. doi: 10.1103/PhysRevE.103.032149. URL https://link.aps.org/doi/10.1103/PhysRevE.103.032149.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. 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. 1273–1282. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Jean Ogier du Terrail, Samy-Safwan Ayed, Edwige Cyffers, Felix Grimberg, Chaoyang He, Regis Loeb, Paul Mangold, Tanguy Marchand, Othmane Marfoq, Erum Mushtaq, Boris Muzellec, Constantin Philippenko, Santiago Silva, Maria Teleńczuk, Shadi Albarqouni, Salman Avestimehr, Aurélien Bellet, Aymeric Dieuleveut, Martin Jaggi, Sai Praneeth Karimireddy, Marco Lorenzi, Giovanni Neglia, Marc Tommasi, and Mathieu Andreux. Flamby: Datasets and benchmarks for cross-silo federated learning in realistic healthcare settings. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 5315–5334. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/232eee8ef411a0a316efa298d7be3c2b-Paper-Datasets_and_Benchmarks.pdf.

Lloyd S. Shapley. A value for n-person games. In Harold W. Kuhn and Albert W. Tucker (eds.), *Contributions to the Theory of Games II*, pp. 307–317. Princeton University Press, Princeton, NJ, 1953. doi: 10.1515/9781400881970-018.

Zhuan Shi, Lan Zhang, Zhenyu Yao, Lingjuan Lyu, Cen Chen, Li Wang, Junhao Wang, and Xiang-Yang Li. FedFAIM: A Model Performance-Based Fair Incentive Mechanism for Federated Learning. *IEEE Transactions on Big Data*, 10(6):1038–1050, December 2024. ISSN 2332-7790, 2372-2096. doi: 10.1109/TBDATA.2022.3183614. URL https://ieeexplore.ieee.org/document/9797864/.

Tianshu Song, Yongxin Tong, and Shuyue Wei. Profit Allocation for Federated Learning. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2577–2586, Los Angeles, CA, USA, December 2019. IEEE. ISBN 978-1-7281-0858-2. doi: 10.1109/BigData47090.2019.9006327. URL https://ieeexplore.ieee.org/document/9006327/.

R. P. Stanley. *Enumerative Combinatorics*, volume 1. Cambridge University Press, 1997.

Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6105–6114. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/tan19a.html.

Guojun Tang, Jiayu Zhou, Mohammad Mamun, and Steve Drew. Lightweight and Robust Federated Data Valuation. In *2025 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 2320–2329, Los Alamitos, CA, USA, November 2025. IEEE Computer Society. doi: 10.1109/ICDMW69685.2025.00282. URL https://doi.ieeecomputersociety.org/10.1109/ICDMW69685.2025.00282.

Nurbek Tastan, Samar Fares, Toluwani Aremu, Samuel Horváth, and Karthik Nandakumar. Redefining contributions: Shapley-driven federated learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, pp. 5009–5017, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/554. URL https://doi.org/10.24963/ijcai.2024/554.

Wei Wan, Shengshan Hu, Jianrong Lu, Leo Yu Zhang, Hai Jin, and Yuanyuan He. Shielding federated learning: Robust aggregation with adaptive client selection. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, IJCAI-2022, pp. 753–760. International Joint Conferences on Artificial Intelligence Organization, July 2022. doi: 10.24963/ijcai.2022/106. URL http://dx.doi.org/10.24963/ijcai.2022/106.

Guan Wang, Charlie Xiaoqian Dang, and Ziye Zhou. Measure Contribution of Participants in Federated Learning, September 2019. URL http://arxiv.org/abs/1909.08525. arXiv:1909.08525 [cs].

Junhao Wang, Lan Zhang, Anran Li, Xuanke You, and Haoran Cheng. Efficient Participant Contribution Evaluation for Horizontal and Vertical Federated Learning. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 911–923, Kuala Lumpur, Malaysia, May 2022. IEEE. ISBN 978-1-6654-0883-7. doi: 10.1109/ICDE53745.2022.00073. URL https://ieeexplore.ieee.org/document/9835159/.

Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. A principled approach to data valuation for federated learning. In Qiang Yang, Lixin Fan, and Han Yu (eds.), *Federated Learning: Privacy and Incentive*, pp. 153–167. Springer International Publishing, Cham, 2020. ISBN 978-3-030-63076-8. doi: 10.1007/978-3-030-63076-8_11. URL https://doi.org/10.1007/978-3-030-63076-8_11.

Shuyue Wei, Yongxin Tong, Zimu Zhou, and Tianshu Song. Efficient and fair data valuation for horizontal federated learning. In Qiang Yang, Lixin Fan, and Han Yu (eds.), *Federated Learning: Privacy and Incentive*, pp. 139–152. Springer International Publishing, Cham, 2020. ISBN 978-3-030-63076-8. doi: 10.1007/978-3-030-63076-8_10. URL https://doi.org/10.1007/978-3-030-63076-8_10.

Jun Wen, Xiusheng Li, Xin Ye, Xiaoli Li, and Hang Mao. A highly generalized federated learning algorithm for brain tumor segmentation. *Scientific Reports*, 15(1):21053, 2025. doi: 10.1038/s41598-025-05297-2.

Paul L. Williams and Randall D. Beer. Nonnegative Decomposition of Multivariate Information, April 2010. URL http://arxiv.org/abs/1004.2515. arXiv:1004.2515 [cs].

Yihao Xue, Chaoyue Niu, Zhenzhe Zheng, Shaojie Tang, Chengfei Lyu, Fan Wu, and Guihai Chen. Toward understanding the influence of individual clients in federated learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):10560–10567, 2021. doi: 10.1609/aaai.v35i12.17263. URL https://ojs.aaai.org/index.php/AAAI/article/view/17263.

Bingjie Yan, Boyi Liu, Lujia Wang, Yize Zhou, Zhixuan Liang, Ming Liu, and Cheng-Zhong Xu. FedCM: A Real-time Contribution Measurement Method for Participants in Federated Learning. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Shenzhen, China, July 2021. IEEE. ISBN 978-1-6654-3900-8. doi: 10.1109/IJCNN52387.2021.9534451. URL https://ieeexplore.ieee.org/document/9534451/.

Mingkun Yang, Ran Zhu, Qing Wang, and Jie Yang. FedTrans: Client-transparent utility estimation for robust federated learning. In *International Conference on Learning Representations*, pp. 42668–42692, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/bb309cc1fbdb88ea755bd7cee4b310ec-Paper-Conference.pdf.

Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pp. 5650–5659. Pmlr, 2018.

Zhaoyang You, Xinya Wu, Kexuan Chen, Xinyi Liu, and Chao Wu. Evaluate the Contribution of Multiple Participants in Federated Learning. In Christine Strauss, Gabriele Kotsis, A Min Tjoa, and Ismail Khalil (eds.), *Database and Expert Systems Applications*, volume 12924, pp. 189–194. Springer International Publishing, Cham, 2021. ISBN 978-3-030-86474-3 978-3-030-86475-0. doi: 10.1007/978-3-030-86475-0_19. URL https://link.springer.com/10.1007/978-3-030-86475-0_19. Series Title: Lecture Notes in Computer Science.

Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. A Fairness-aware Incentive Scheme for Federated Learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 393–399, New York NY USA, February 2020a. ACM. ISBN 978-1-4503-7110-0. doi: 10.1145/3375627.3375840. URL https://dl.acm.org/doi/10.1145/3375627.3375840.

Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. A Sustainable Incentive Scheme for Federated Learning. *IEEE Intelligent Systems*, 35(4):58–69, July 2020b. ISSN 1541-1672, 1941-1294. doi: 10.1109/MIS.2020.2987774. URL https://ieeexplore.ieee.org/document/9069185/.

Yufeng Zhan, Peng Li, Zhihao Qu, Deze Zeng, and Song Guo. A Learning-Based Incentive Mechanism for Federated Learning. *IEEE Internet of Things Journal*, 7(7):6360–6368, July 2020. ISSN 2327-4662, 2372-2541. doi: 10.1109/JIOT.2020.2967772. URL https://ieeexplore.ieee.org/document/8963610/.

Jingwen Zhang, Yuezhou Wu, and Rong Pan. Incentive Mechanism for Horizontal Federated Learning Based on Reputation and Reverse Auction. In *Proceedings of the Web Conference 2021*, pp. 947–956, Ljubljana Slovenia, April 2021. ACM. ISBN 978-1-4503-8312-7. doi: 10.1145/3442381.3449888. URL https://dl.acm.org/doi/10.1145/3442381.3449888.

Meiying Zhang, Huan Zhao, Sheldon Ebron, and Kan Yang. Towards fair, robust and efficient client contribution evaluation in federated learning. In *GLOBECOM 2024 - 2024 IEEE Global Communications Conference*, pp. 1635–1640, 2024. doi: 10.1109/GLOBECOM52923.2024.10901033.

Jie Zhao, Xinghua Zhu, Jianzong Wang, and Jing Xiao. Efficient Client Contribution Evaluation for Horizontal Federated Learning. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3060–3064, Toronto, ON, Canada, June 2021. IEEE. ISBN 978-1-7281-7605-5. doi: 10.1109/ICASSP39728.2021.9413377. URL https://ieeexplore.ieee.org/document/9413377/.

A Additional PID Definition and Diagnostics

This appendix provides additional detail on Partial Information Decomposition, including the Williams–Beer PID, illustrative PID examples, an outcome-specific PID decomposition for class-wise diagnostics, the estimator-selection experiment used for FedPID, and the diagnostic result using hard-label PID.

A.1 Definition of Williams–Beer PID

Here we closely follow the presentation of the PID framework as developed in (Williams & Beer, 2010). In Appendix A.4 we give the outcome-specific decomposition used for class-wise hard-label diagnostics. Let S be a discrete target and let $R = \{R_1, \ldots, R_n\}$ be a set of source variables. Williams and Beer define redundancy over collections of nonempty source subsets. For a source subset $A \subseteq R$, the mutual information can be written as an expectation of outcome-specific information,

$$I(S; A) = \sum_s p(s) I(S = s; A), \tag{7}$$

where the specific information that source A provides about the target outcome $S = s$ is

$$I(S = s; A) = \sum_a p(a \mid s) \left[\log_2 \frac{1}{p(s)} - \log_2 \frac{1}{p(s \mid a)} \right] = \sum_a p(a \mid s) \log_2 \frac{p(s \mid a)}{p(s)}. \tag{8}$$

Thus, $I(S = s; A)$ measures the average reduction in the surprise of outcome s after observing A, and $I(S; A)$ averages this quantity over target outcomes.

For a collection of sources $\alpha = \{A_1, \ldots, A_m\}$, the Williams–Beer redundancy measure is the expected minimum specific information given by any source in the collection (the symbols α and β in this appendix follow the original Williams–Beer notation for PID lattice nodes and are unrelated to the aggregation coefficients (α, β, γ) used in Section 4.3),

$$I_{\min}(S; \alpha) = I_{\min}(S; \{A_1, \ldots, A_m\}) = \sum_s p(s) \min_{A_i \in \alpha} I(S = s; A_i). \tag{9}$$

The minimum is taken separately for each target outcome before averaging because two sources may have the same mutual information about S while carrying information about different outcomes. In that case, $I_{\min}$ counts as redundant only the information that all sources provide about the same target outcomes.

The natural domain of $I_{\min}$ is the set of antichains over nonempty subsets of R. Let $\mathcal{P}_1(R) = \mathcal{P}(R) \setminus \{\emptyset\}$, where $\mathcal{P}(R)$ is the power set of R. Following Williams and Beer, define

$$\mathcal{A}(R) = \{\alpha \in \mathcal{P}_1(\mathcal{P}_1(R)) : \forall A_i, A_j \in \alpha,\ A_i \not\subset A_j\}. \tag{10}$$

Elements of $\mathcal{A}(R)$ are written using source indices; for example, $\{\{R_1\}, \{R_2\}\}$ is denoted $\{1\}\{2\}$ and $\{\{R_1, R_2\}\}$ is denoted $\{12\}$. The redundancy lattice is defined by the partial order

$$\alpha \preceq \beta \quad \Longleftrightarrow \quad \forall B \in \beta,\ \exists A \in \alpha\ , A \subseteq B. \tag{11}$$

Intuitively, moving upward in this lattice corresponds to increasingly informative source collections.

The partial information atoms are obtained by Möbius inversion on this lattice. Let $\Pi_R(S; \alpha)$ denote the partial information associated with node $\alpha \in \mathcal{A}(R)$. The cumulative redundancy at a node decomposes as

$$I_{\min}(S; \alpha) = \sum_{\beta \preceq \alpha} \Pi_R(S; \beta), \tag{12}$$

or equivalently,

$$\Pi_R(S; \alpha) = I_{\min}(S; \alpha) - \sum_{\beta \prec \alpha} \Pi_R(S; \beta). \tag{13}$$

Williams & Beer (2010) proved that the resulting decomposition is such that all terms $\Pi_R(S; \alpha)$ in the lattice are non-negative (see their Theorem 5). Specifically for the two-source case used in

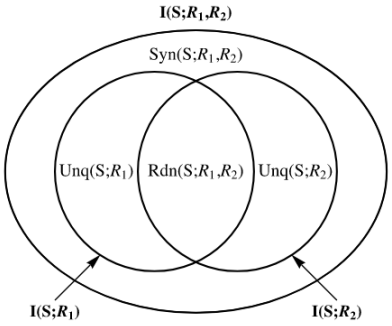

Figure 4: Structure of multivariate information for one target and two sources. Labelled regions correspond to unique information (Unq), redundancy (Rdn), and synergy (Syn) (Williams & Beer, 2010).

FedPID with sources R_1 and R_2, as illustrated in Figure 4, the decomposition contains four atoms:

$$I(S; R_1, R_2) = \Pi_R(S; \{1\}\{2\}) + \Pi_R(S; \{1\}) + \Pi_R(S; \{2\}) + \Pi_R(S; \{12\}), \tag{14}$$

$$\mathrm{Rdn}(S; R_1, R_2) = \Pi_R(S; \{1\}\{2\}) = I_{\min}(S; \{R_1\}, \{R_2\}), \tag{15}$$

$$\mathrm{Unq}(S; R_1) = \Pi_R(S; \{1\}) = I(S; R_1) - \mathrm{Rdn}(S; R_1, R_2), \tag{16}$$

$$\mathrm{Unq}(S; R_2) = \Pi_R(S; \{2\}) = I(S; R_2) - \mathrm{Rdn}(S; R_1, R_2), \tag{17}$$

$$\begin{aligned}\mathrm{Syn}(S; R_1, R_2) &= \Pi_R(S; \{12\}) \\ &= I(S; R_1, R_2) - I(S; R_1) - I(S; R_2) + \mathrm{Rdn}(S; R_1, R_2).\end{aligned} \tag{18}$$

In the main FedPID construction, this two-source system is instantiated class-wise with $S = Y^{(c)}$, $R_1 = Z_k^{(c)}$, and $R_2 = Z_{-k}^{(c)}$, where $Y^{(c)}$ is the one-vs-rest binary target, $Z_k^{(c)}$ is the binned class-c predictive state of client k, and $Z_{-k}^{(c)}$ is the tuple-valued binned predictive state of the remaining clients. The resulting atoms are the class-conditioned quantities $\mathrm{Unq}_{k,c}^{(t)}$, $\mathrm{Unq}_{-k,c}^{(t)}$, $\mathrm{Rdn}_{k,c}^{(t)}$, and $\mathrm{Syn}_{k,c}^{(t)}$ used in the main text, and their non-negativity follows directly from the standard Williams–Beer result. Appendix A.4 gives the separate outcome-specific decomposition used for the hard-label tuple diagnostic.

A.2 PID Examples

Unique information example. Let $S \sim \mathrm{Bernoulli}\left(\frac{1}{2}\right)$, set $R_1 = S$, and take R_2 to be independent of S, denoted by $R_2 \perp S$. In this case,

$$I(S; R_1) = 1, \qquad I(S; R_2) = 0, \qquad I(S; R_1, R_2) = 1,$$

so the entire one bit of information about S is carried uniquely by R_1. The PID terms are therefore

$$\mathrm{Rdn}(S; R_1, R_2) = 0, \quad \mathrm{Unq}(S; R_1) = 1, \quad \mathrm{Unq}(S; R_2) = 0, \quad \mathrm{Syn}(S; R_1, R_2) = 0.$$

Redundant information example. Let $S \sim \mathrm{Bernoulli}\left(\frac{1}{2}\right)$ and define $R_1 = S$ and $R_2 = S$. Because each predictor is an exact copy of the target,

$$I(S; R_1) = 1, \qquad I(S; R_2) = 1, \qquad I(S; R_1, R_2) = 1,$$

and the PID decomposition becomes

$$\mathrm{Rdn}(S; R_1, R_2) = 1, \quad \mathrm{Unq}(S; R_1) = 0, \quad \mathrm{Unq}(S; R_2) = 0, \quad \mathrm{Syn}(S; R_1, R_2) = 0.$$

Synergistic information example. Let $R_1 \sim \mathrm{Bernoulli}\left(\frac{1}{2}\right)$ and $R_2 \sim \mathrm{Bernoulli}\left(\frac{1}{2}\right)$ independently, and define $S = R_1 \oplus R_2$. In this case,

$$I(S; R_1) = 0, \qquad I(S; R_2) = 0, \qquad I(S; R_1, R_2) = 1,$$

so no information is present in either predictor alone or redundantly, and the entire bit arises from their joint state:

$$\mathrm{Rdn}(S; R_1, R_2) = 0, \quad \mathrm{Unq}(S; R_1) = 0, \quad \mathrm{Unq}(S; R_2) = 0, \quad \mathrm{Syn}(S; R_1, R_2) = 1.$$

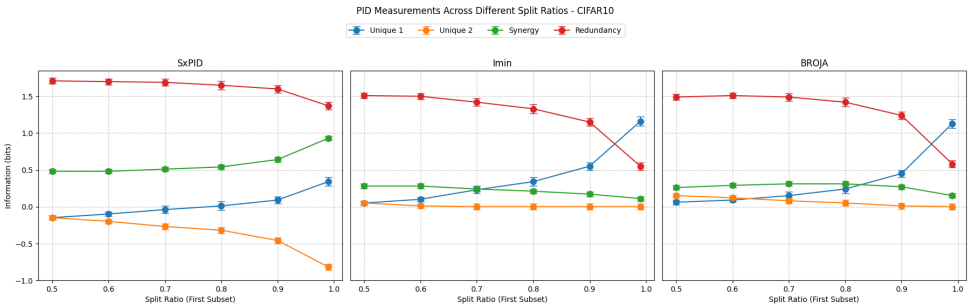

Figure 5: Quantity skew split where client 1 holds 50%–99% of the training data and client 2 holds the remainder.

A.3 Selecting the PID Estimator

To choose a PID estimator for the main experiments, we compared SxPID (Makkeh et al., 2021), $I_{\min}$ (Williams & Beer, 2010), and BROJA (Bertschinger et al., 2014) across several controlled settings and report the quantity-skew case here as a representative example. In the quantity skew case (Figure 5), both clients contain all classes, so redundancy is expected to dominate. As the data share of client 1 increases, its unique information should increase, while the unique information of client 2 should go to zero. This expected pattern is captured by both $I_{\min}$ and BROJA, whereas SxPID assigns negative unique information to client 2 and exhibits increasing synergy, which is counterintuitive in this setting. Between the two non-negative estimators, $I_{\min}$ separates the two clients earlier: at a split ratio of 0.6, the unique information of client 1 already exceeds that of client 2 under $I_{\min}$, while BROJA still assigns similar values and only differentiates more clearly at larger split ratios. Additionally, BROJA estimation requires solving an optimisation problem over probability distributions (Bertschinger et al., 2014), which is substantially more computationally expensive. Finally, as shown in Appendix A.4, the Williams–Beer approach naturally decomposes the global non-negative PID decomposition into an average of outcome-specific non-negative PID decompositions, while this average interpretation does not hold for the BROJA PID. Overall, we use $I_{\min}$ in the rest of our experiments.

A.4 An Outcome-specific PID Decomposition for Class-wise Diagnostics

In this paper, class-wise PID diagnostics are obtained in two ways. The main FedPID construction uses one-vs-rest binarisation, so each class c is handled by an ordinary PID with binary target $Y^{(c)}$. Because this binarised target is still a standard discrete PID target, its atoms inherit the usual Williams–Beer $I_{\min}$ non-negativity result and require no separate proof. The hard-label tuple diagnostic in Appendix A.5 instead keeps the multiclass target and uses the outcome-specific decomposition below. In Appendix A.1 we have described the PID framework as developed by Williams & Beer (2010). We now show that the PID decomposition operates at the level of each target outcome $S = s$, and that the standard PID decomposition is the resulting average of all outcome-specific PID decompositions. When target outcomes correspond to class labels, this outcome-specific decomposition gives a class-wise view of unique, redundant, and synergistic information.

The standard PID decomposition of Williams & Beer (2010) is obtained by Möbius inversion on the lattice, given the relation between the redundancy terms and partial terms as indicated in Eq. 12. Since the redundancy terms $I_{\min}(S;\alpha)$ are defined as an average over $p(s)$ of the specific information terms $\min_{A_i\in\alpha} I(S=s;A_i)$ (Eq. 9), we examine an extension of the PID framework to specific outcomes, that is, such that

$$I_{\min}(S;\alpha) = \sum_s p(s) I_{\min}(s;\alpha) \tag{19}$$

and

$$I_{\min}(s;\alpha) = \min_{A_i\in\alpha} I(s;A_i). \tag{20}$$

We now show that indeed a non-negative Partial Information Decomposition holds for each $S = s$, when defining the cumulative redundancy terms as in Eq. 20 and performing Möbius inversion on the

lattice to define the outcome-specific partial information terms such that

$$I_{\min}(s;\alpha) = \sum_{\beta \preceq \alpha} \Pi_R(s;\beta). \tag{21}$$

We will now show that the terms $\Pi_R(s;\alpha)$ are non-negative and that $\Pi_R(S;\alpha)$ is the average over $p(s)$ of all terms $\Pi_R(s;\alpha)$, so that the standard PID decomposition is the average of the outcome-specific decompositions.

Theorem 1. *The outcome-specific partial terms $\Pi_R(s;\alpha)$ are non-negative.*

Proof: The proof follows the same procedure as the original proof in (Williams & Beer, 2010). For the sake of space, here we highlight the parts of the proof that are left unchanged and focus on the parts of the proof that differ when moving from $\Pi_R(S;\alpha)$ to $\Pi_R(s;\alpha)$. The original proof proceeds sequentially, proving their Theorems 3 and 4 as preliminary steps. We will here explicitly indicate when referring to equations in their paper, and otherwise refer to equations in our work.

First, their Theorem 1 already concerns the specific information $I(S=s;A)$ (Eq. 8), proving its non-negativity. Similarly, their Lemma 1 shows that $I(S=s;A)$ increases monotonically on the lattice. Therefore, because these results are already for $S=s$, we will use them as stated in (Williams & Beer, 2010). Subsequently, their Theorem 2 states the monotonicity of $I_{\min}(S;\alpha)$ in the lattice. The proof operates by contradiction, arguing that if $\alpha \prec \beta$ and yet $I_{\min}(S;\beta) < I_{\min}(S;\alpha)$, there must be some $S=s'$ such that, for all $A \in \alpha$ and some $B \in \beta$, it happens that $I(S=s';B) < I(S=s';A)$. Accordingly, the same proof by contradiction can be adapted to state the monotonicity of any $I_{\min}(S=s;\alpha)$ in the lattice: repeat the same argument but replace $S=s'$ by the specific $S=s$ being considered.

Their Theorem 3 re-expresses $\Pi_R(S;\alpha)$ in the form of their equation A6. It can be checked that the proof of the theorem only relies on structural properties of the lattice and the application of the inclusion-exclusion principle (Stanley, 1997). The theorem remains valid, replacing their function $f(\mathcal{B})$ by a sum over terms $\Pi_R(s;\beta)$ with $\beta \in \mathcal{B}$, instead of a sum over terms $\Pi_R(S;\beta)$. This is because the theorem only depends on which terms are included in the sum, which is determined by the lattice structure and not by the definition of $f(\mathcal{B})$ with an average over $p(s)$ or for a specific $S=s$. Similarly, their Lemma 2 applies the maximum-minimums identity among sets, which again is a property that is to be applied considering the structure of the lattice and not the definition of the redundancy terms.

Their Theorem 4 re-expresses $\Pi_R(s;\alpha)$ in the form of their equation A7. The proof of Theorem 4 relies on the following properties and previously proven statements: First, it relies on some structural properties. This comprises relying on their Theorem 3 and their Lemma 2, and relying on their equation A4, which again regards some property of the lattice structure. Second, the proof uses their Lemma 1 which, as indicated above, already regards the specific information terms $I(S=s;A)$. Third, the proof uses the definition of the cumulative redundancy terms (Eq. 9) as an average over $p(s)$. It can be checked that all the steps that lead from their expression in their equation A6 to A7 operate internally to the average over $p(s)$ and hence can be applied for a fixed $S=s$.

Based on the preliminaries above, (Williams & Beer, 2010) prove their Theorem 5. As in the case of their Theorem 2, the proof proceeds by contradiction. The sum over $p(s)$ is factored out, and it is argued that in order to obtain a negative $\Pi_R(S;\alpha)$, there must exist at least one $S=s'$ for which a negative term is obtained in the sum. When moving from $\Pi_R(S;\alpha)$ to a specific $\Pi_R(s;\alpha)$, the same argument can be repeated by replacing $S=s'$ with the specific $S=s$ being considered. □

As a direct consequence of the proof of non-negativity of the outcome-specific partial terms $\Pi_R(s;\alpha)$, it is identified that the standard partial terms of Williams & Beer (2010) correspond to the average over $p(s)$ of the outcome-specific terms:

Corollary 1. *The non-negative PID decomposition of the global information about a target S corresponds to the average over $p(s)$ of the outcome-specific non-negative PID decompositions for each outcome $S=s$, such that*

$$\Pi_R(S;\alpha) = \sum_s p(s)\Pi_R(s;\alpha). \tag{22}$$

Proof: Because Eq. 19 writes each cumulative redundancy term as a $p(s)$-weighted average of its outcome-specific counterpart, and Möbius inversion on a fixed lattice is linear, the same averaging

relation holds for the inverted partial terms. From Theorem 5 of (Williams & Beer, 2010), it is identified that each $\Pi_R(s;\alpha)$ can be expressed as:

$$\Pi_R(s;\alpha) = \min_{A\in\alpha} I(s;A) - \max_{\beta\in\alpha^-} \min_{B\in\beta} I(s;B), \tag{23}$$

where α^- are the nodes immediately below α in the redundancy lattice. □

As a cautionary note, this formulation of the standard non-negative PID as an average of outcome-specific non-negative PID decompositions holds for the measures defined by Williams & Beer (2010), but does not hold in general for any alternative definition of PID terms. In particular, it can be checked that this granularity is lost for a PID defined in (Bertschinger et al., 2014). This provides an additional reason to select the Williams–Beer $I_{\min}$ measure when outcome-specific hard-label diagnostics are desired.

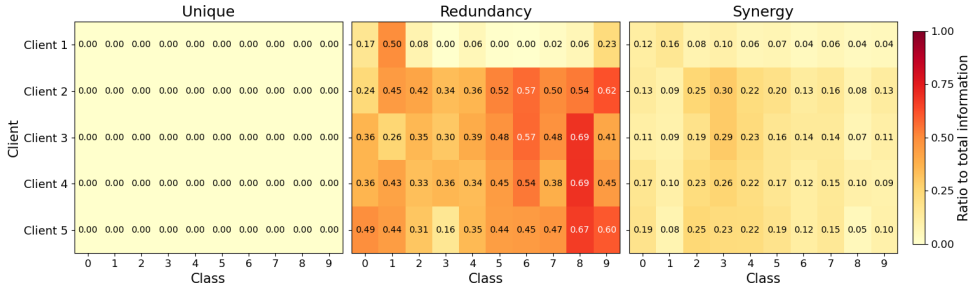

Figure 6: Class-wise normalised PID terms for the CIFAR-10 non-IID split at the first round using hard-label tuple PID.

A.5 Hard-Label Tuple PID Diagnostic

Figure 6 repeats the first-round class-wise diagnostic on the CIFAR-10 non-IID split using the hard-label tuple variant, where each source is an argmax prediction and the target remains multiclass; the class-wise normalised PID terms are obtained using the outcome-specific decomposition from Appendix A.4. Unlike the class-conditioned soft-probability PID in Figure 3, the hard-label tuple assigns zero unique information to every client, including the specialist client that alone holds classes 0 and 1. It also yields a noisier diagnostic: the hard-label tuple has a larger joint PID space, and the figure assigns visible redundant information to Clients 2–5 for classes 0 and 1 even though those classes are absent from their local data. This comparison therefore supports using binned softmax probabilities as sources under finite validation data: soft probabilities preserve confidence information that hard argmax sources discard, allowing the PID profile to localise specialist information.

B Additional Experimental Setup Details

IID setting. In the IID setting, training samples are distributed uniformly across clients.

Quantity-skew setting. The quantity-skew setting changes client sample size while keeping the same class distribution across clients. For K clients, sample size follows the descending ratio $(K+1) : K : \cdots : 2$; in the five-client experiments, this gives relative data sizes $6:5:4:3:2$.

Feature-noise setting. The feature-noise setting starts from IID and adds noise to the input quality. Client 1 is left uncorrupted, while clients $k = 2,\ldots,K$ receive increasing Gaussian noise. With base noise level σ_0, client k receives noise level $(k-1)\sigma_0$. The noise is added in pixel space and clipped to the valid image range before training, so the corruption affects the features while preserving the class labels.

Label-noise setting. The label-noise setting starts from IID and adds noise to the input label. Client 1 is left uncorrupted, while clients $k = 2,\ldots,K$ receive increasing label noise. With base label-noise ratio ρ_0, client k has corruption ratio $(k-1)\rho_0$. For client k, a fraction ρ_k of training labels is

replaced by labels sampled uniformly from the incorrect classes, so the images are unchanged but the supervision becomes progressively less reliable.

Feature-and-label-noise setting. The feature-and-label-noise setting combines the two progressive corruptions above. Client 1 is left uncorrupted, while clients $k = 2, \ldots, K$ receive both feature-noise level $(k-1)\sigma_0$ and label-noise ratio $(k-1)\rho_0$.

Non-IID setting. For the non-IID experiment, we use a deterministic five-client split with one exclusive specialist client and four clients that share the remaining class support. Client 1 receives 2,500 examples from class 0 and 2,500 examples from class 1. Each of Clients 2–5 receives 625 examples from each of classes 2–9. Thus, all clients have 5,000 training examples, but Client 1 is the only client with classes 0 and 1, while the other four clients contain the same eight remaining classes.

Dirichlet class-imbalance setting. We generate three five-client partitions using three seeds. For each class, we draw allocation proportions over clients from a symmetric Dirichlet distribution with $\alpha = 0.5$. A self-balancing allocation excludes clients that have already reached the average client pool size before renormalising the proportions. Every training example is assigned to one client, and each client pool contains at least ten examples. This produces unequal class proportions with overlapping class support; the remaining training settings follow Appendix C.

Free-rider setting. The free-rider setting models clients that participate in the federation but do not perform local learning. Each free-rider returns the current global model unchanged rather than a locally trained update in each global round. We consider $m \in \{1, 2, 3\}$ free-rider clients.

Data-inflation setting. The data-inflation setting models clients that increase their apparent data volume by repeatedly reusing local examples during training. For data inflation, client 5 trains on n_{distinct} distinct local examples per class, each repeated factor times; we report results for $n_{\text{distinct}} \in \{1, 5, 10\}$ and $\text{factor} \in \{500, 1000\}$.

C Implementation Details

For CIFAR-10, Fed-ISIC, and ImageNet-100, we use the official train/test splits. Client partitions and corruptions are applied only to the official training split; the official test split is divided into the server validation holdout and the final test set as described in Section 5.1. This gives server-validation/final-test sizes of 3,000/7,000 for CIFAR-10, 1,395/3,255 for Fed-ISIC, and 1,500/3,500 for ImageNet-100. For CIFAR-10 and ImageNet-100, the server validation holdout is class-balanced; for Fed-ISIC, we pool the official test samples across centres and reserve a 30% class-stratified split, preserving the Fed-ISIC test-set class proportions. Contribution-evaluation baselines use their standard utilities: Shapley-style methods use validation accuracy as the coalition utility, LOO uses the performance change after removing a client, and GCS uses gradient cosine similarity. For aggregation accuracy, FedAvg uses dataset-size-weighted parameter averaging, and FedProx uses the same server averaging with proximal coefficient $\mu = 0.01$. Trimmed Mean removes one smallest and one largest client value per coordinate, Coordinate-wise Median takes the coordinate median, and Krum uses Byzantine tolerance $f = 1$. FedPID uses aggregation coefficients selected from the grid in Appendix D.5; the default four-bin FedPID estimator uses interior probability edges $(0.1, 0.4, 0.8)$, giving bins $[0, 0.1)$, $[0.1, 0.4)$, $[0.4, 0.8)$, and $[0.8, 1]$. For the same client predictions on a class-balanced validation set, accuracy and balanced accuracy induce identical aggregation weights; we therefore report only validation-accuracy weighting for this criterion.

Corruption levels. In Table 1 and Table 3, the CIFAR-10 feature-and-label-noise setting uses $\sigma_0 = 0.1$ and $\rho_0 = 0.25$. For the 20- and 50-client feature-and-label-noise experiments in Appendix D.3.1, we set $\sigma_0 = 0.4/(K-1)$ and $\rho_0 = 1/(K-1)$ to preserve the five-client corruption ranges. In Table 2, the Fed-ISIC feature-noise setting uses $\sigma_0 = 0.25$ and the label-noise setting uses $\rho_0 = 0.25$. Additional CIFAR-10 appendix diagnostics use $\sigma_0 = 0.5$ for feature noise and $\rho_0 = 0.25$ for label noise in the five-client setting, and $\sigma_0 = \rho_0 = 0.1$ in the ten-client setting. ImageNet-100 feature-and-label noise uses $\sigma_0 = 0.1$ and $\rho_0 = 0.25$.

FedPID aggregation weights. At each PID measurement round, each participating client adds its round-wise score $s_k^{(t)}$ to its accumulator $A_k^{(t)}$; non-participating clients keep their previous

accumulator. Aggregation weights are normalised over the current participant set,

$$w_k^{(t)} = \frac{A_k^{(t)}}{\sum_{\ell \in \mathcal{S}^{(t)}} A_\ell^{(t)}}, \qquad k \in \mathcal{S}^{(t)}, \tag{24}$$

and the server update is

$$\theta^{(t+1)} = \sum_{k \in \mathcal{S}^{(t)}} w_k^{(t)} \theta_k^{(t+1)}. \tag{25}$$

On CIFAR-10, we train a ResNet-18 (He et al., 2016) with 5 clients for 40 global rounds, using 5 local epochs per client in each round. On Fed-ISIC, we use the official Fed-ISIC2019 benchmark with 6 clinical-centre clients and 8 skin-lesion classes, using a pre-trained EfficientNet (Tan & Le, 2019) for 5 global rounds with 5 local epochs per client in each round. On ImageNet-100 (Deng et al., 2009), used for diagnostics only, we train a ResNet-18 with 5 clients for 1 global round and 10 local epochs per client.

D Additional Experimental Results

This section provides additional FedPID results, including last-round non-IID class-wise diagnostics, further CIFAR-10 settings with 5 clients including Dirichlet class imbalance, extensions to 10-client CIFAR-10 experiments and 20- and 50-client pools under full and partial participation, validation-set stress tests for contribution evaluation and aggregation, diagnostics on ImageNet-100 (Deng et al., 2009), and hyperparameter tuning over PID measurement frequency, bin count, and PID atom coefficients.

D.1 Last-Round Non-IID Class-Wise Diagnostic

Figure 7 repeats the CIFAR-10 non-IID diagnostic of class-wise normalised PID terms using the last-round FedPID profile. Client 1 still concentrates its unique information on classes 0 and 1, matching its exclusive class support. For clients 2–5 on classes 2–9, the redundant and synergistic terms are more stable than in the first-round heatmap in Figure 3. This is consistent with two sources of first-round variability: the tuple-valued source has a high-dimensional empirical state space under finite validation data, and the client models have completed only one global round of training, so their class-wise probability profiles are still less stable. The last-round profile is therefore a diagnostic check on the heatmap structure, while the first-round profile remains the contribution signal used by FedPID$_F$.

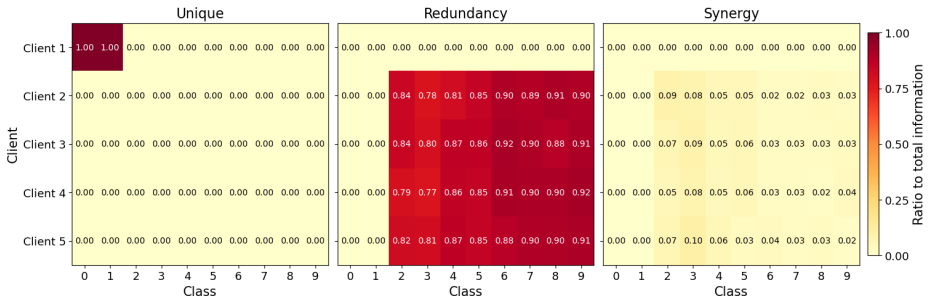

Figure 7: Class-wise normalised PID terms for the CIFAR-10 non-IID split at the last round.

D.2 Client Contribution Results on Additional CIFAR-10 Settings

Table 4 reports FedPID scores measured at the first round on additional CIFAR-10 settings. For feature noise and label noise, the scores are averaged over three independent runs and reported as mean $\pm$ standard deviation. In both settings, FedPID assigns the largest contribution to the uncorrupted client and progressively smaller scores to clients with stronger corruption, showing that PID scores measured at the first round are sensitive to client data quality before aggregation has smoothed the

Table 4: Client contribution scores of FedPID across CIFAR-10 settings evaluated at the first round. For the feature-noise setting, $\sigma_0 = 0.5$; for the label-noise setting, $\rho_0 = 0.25$. Expected contribution patterns are decreasing for feature noise and label noise, with very low contribution assigned to free-riders and data inflators.

Setting	C1	C2	C3	C4	C5
Feature Noise	0.680 ± 0.027	0.108 ± 0.024	0.086 ± 0.007	0.065 ± 0.012	0.061 ± 0.011
Label Noise	0.512 ± 0.010	0.238 ± 0.003	0.154 ± 0.014	0.063 ± 0.012	0.032 ± 0.007
Free-riders: 1	0.245	0.267	0.240	0.245	0.001
Free-riders: 2	0.315	0.364	0.321	0.000	0.000
Free-riders: 3	0.538	0.462	0.000	0.000	0.000
Data-inflation: $n_{\text{distinct}} = 1$, factor $= 500$	0.216	0.276	0.211	0.269	0.028
Data-inflation: $n_{\text{distinct}} = 1$, factor $= 1000$	0.211	0.267	0.239	0.247	0.036
Data-inflation: $n_{\text{distinct}} = 5$, factor $= 500$	0.258	0.226	0.221	0.245	0.050
Data-inflation: $n_{\text{distinct}} = 5$, factor $= 1000$	0.230	0.243	0.252	0.233	0.042
Data-inflation: $n_{\text{distinct}} = 10$, factor $= 500$	0.214	0.239	0.243	0.249	0.056
Data-inflation: $n_{\text{distinct}} = 10$, factor $= 1000$	0.220	0.229	0.241	0.256	0.054

models. The free-rider and data-inflation rows are included as targeted diagnostics of whether FedPID can detect clients whose apparent participation does not correspond to useful predictive information. We therefore report the observed diagnostic score for each configuration without standard deviation. In the free-rider rows, "Free-riders: m" means that m clients are free-riders in the federation. Free-rider clients return the unchanged global model, so their binned predictive distributions match the previous round's global state, and FedPID assigns near-zero contribution to the free-rider clients: C5 when there is one free-rider, C4–C5 when there are two, and C3–C5 when there are three. In the data-inflation rows, the inflated client reuses a small number of distinct local examples with different duplication factors. Across all inflation settings, FedPID assigns the inflated client, C5, a substantially smaller score than the other clients, indicating that the PID score reflects predictive information rather than apparent local dataset size.

D.2.1 Dirichlet Class-Imbalanced Client Data

We evaluate three five-client partitions with class-imbalanced data generated from a Dirichlet distribution ($\alpha = 0.5$), allowing class support to overlap across clients. Partition details are given in Appendix B. Table 5 and Figure 8 show one of these partitions.

Table 5: Client contribution scores for the five-client Dirichlet partition ($\alpha = 0.5$) on CIFAR-10 shown in Figure 8. All methods use the same partition.

	Dirichlet class imbalance ($\alpha = 0.5$)				
Method	C1	C2	C3	C4	C5
SV	0.116	0.183	0.128	0.211	0.160
LOO	0.002	0.021	0.004	0.050	0.026
GTG	−0.181	0.303	−0.097	0.349	0.106
GCS	0.192	0.172	0.273	0.231	0.132
FedPID$_F$	0.186	0.196	0.188	0.253	0.176
FedPID$_L$	0.159	0.226	0.169	0.247	0.199

Under Dirichlet class imbalance (Table 5), FedPID$_F$ scores range from 0.176 to 0.253, while FedPID$_L$ scores range from 0.159 to 0.247. The class-conditioned PID profiles retain their interpretation under overlapping class support. In this partition, C1 has no examples from classes 7–9 (Figure 8), and its first-round PID attributes all measured information about these classes to its peers. The other two partitions show the same first-round pattern for classes absent from a client's local data, supporting FedPID's ability to reveal class-specific information supplied by peers. These results complement the controlled specialist split and extend the evidence for class-wise contribution evaluation to overlapping client class distributions.

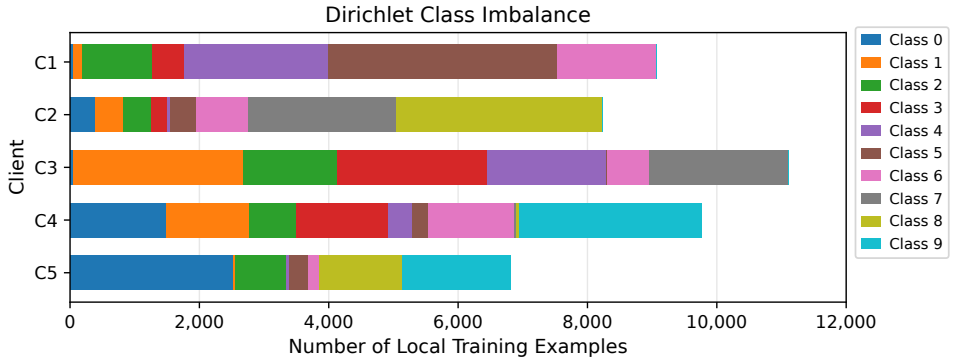

Figure 8: Class distribution of each client's local training data for the Dirichlet partition in Table 5 ($\alpha = 0.5$). Colours indicate classes 0–9; counts exclude each client's local hold-out set. C1 has no examples from classes 7–9.

D.3 Scalability and Robustness

We report the run-to-run stability of the FedPID scores from Tables 1 and 2 in Tables 6 and 7.

Table 6: Standard deviations of FedPID client contribution scores across CIFAR-10 settings, corresponding to Table 1.

	IID					Quantity Skew					Feature + Label Noise					Non-IID				
Method	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5	C1	C2	C3	C4	C5
FedPID$_F$	0.006	0.007	0.006	0.002	0.005	0.005	0.010	0.007	0.011	0.008	0.012	0.007	0.002	0.000	0.003	0.007	0.009	0.007	0.013	0.014
FedPID$_L$	0.000	0.000	0.001	0.000	0.001	0.001	0.001	0.001	0.001	0.000	0.012	0.006	0.003	0.004	0.002	0.003	0.001	0.001	0.002	0.001

Table 7: Standard deviations of FedPID client contribution scores across Fed-ISIC settings, corresponding to Table 2.

	Clean						Feature Noise						Label Noise					
Method	C1	C2	C3	C4	C5	C6	C1	C2	C3	C4	C5	C6	C1	C2	C3	C4	C5	C6
FedPID$_F$	0.023	0.021	0.011	0.027	0.029	0.019	0.005	0.046	0.103	0.132	0.016	0.009	0.040	0.025	0.034	0.031	0.023	0.017
FedPID$_L$	0.014	0.017	0.013	0.011	0.011	0.018	0.022	0.046	0.007	0.025	0.018	0.016	0.004	0.010	0.028	0.015	0.005	0.003

We also examine exact Shapley values for the CIFAR-10 IID setting after only one global round of training. Table 8 reports per-run client contributions across three independent runs; because the clients have equal data distributions, the expected contribution should be close to uniform. However, after only one round, exact Shapley is unstable and assigns negative scores to some clients.

Table 8: Client contribution scores for each independent run after one training round using exact Shapley value (SV).

Run	C1	C2	C3	C4	C5
Run 1	0.006	0.016	-0.004	0.005	0.003
Run 2	0.004	0.024	-0.002	0.000	-0.001
Run 3	0.014	0.032	0.002	-0.020	-0.006

We further evaluate FedPID with 10 clients under IID, quantity-skew, feature-noise, and label-noise settings. Table 9 shows that FedPID preserves the expected contribution structure as the federation size increases. Under IID, scores measured at the first and last rounds remain close to uniform. Under quantity skew, scores measured at the first round broadly track client data size, assigning larger contributions to larger clients up to small adjacent variations, whereas scores measured at the last round become nearly uniform after training, indicating that the trained models eventually provide

similar predictive information despite different local sample counts. Under label noise, FedPID assigns progressively smaller contributions to increasingly corrupted clients. Under feature noise, FedPID$_L$ shows a clean quality ordering, while FedPID$_F$ still assigns the largest scores to the clean and least corrupted clients but is less strictly ordered among intermediate clients. These results indicate that FedPID remains stable and interpretable beyond the five-client setting used in the main diagnostic experiments.

Runtime comparison. Under FedPID's class-conditioned soft-probability tuple construction, one PID measurement round requires one validation pass for each participating client and approximately $|\mathcal{S}^{(t)}|C$ two-source PID solves, one for each evaluated client and class. Table 10 reports the computational cost of contribution-evaluation methods on CIFAR-10 with 5 and 10 clients. Final runtime denotes the time required to obtain one final contribution vector for all clients after an FL run, while per-round runtime denotes the average time required to produce one round-wise contribution during training. FedPID runtimes are reported under the FedPID$_R$ schedule (per-round measurement); FedPID$_F$ requires only the round-1 cost. All experiments were run using an NVIDIA A100 GPU. At 5 clients, FedPID is comparable to GTG-Shapley (2,739 s final and 68 s per round vs. 2,451 s and 61 s); at 10 clients, FedPID is faster (6,299 s and 157 s vs. 11,369 s and 284 s) while retaining PID decomposition information.

Table 9: FedPID client scores on CIFAR-10 with 10 clients. For feature noise, $\sigma_0 = 0.1$, and for label noise, $\rho_0 = 0.1$.

	IID		Quantity Skew		Feature Noise		Label Noise	
Client	FedPID$_F$	FedPID$_L$	FedPID$_F$	FedPID$_L$	FedPID$_F$	FedPID$_L$	FedPID$_F$	FedPID$_L$
C1	0.097 ± 0.005	0.100 ± 0.000	0.129 ± 0.004	0.101 ± 0.001	0.192 ± 0.012	0.299 ± 0.004	0.199 ± 0.001	0.160 ± 0.002
C2	0.106 ± 0.009	0.100 ± 0.000	0.117 ± 0.008	0.101 ± 0.001	0.141 ± 0.018	0.209 ± 0.008	0.174 ± 0.006	0.155 ± 0.001
C3	0.095 ± 0.006	0.101 ± 0.000	0.109 ± 0.004	0.101 ± 0.001	0.089 ± 0.012	0.124 ± 0.003	0.146 ± 0.010	0.146 ± 0.001
C4	0.106 ± 0.002	0.100 ± 0.000	0.110 ± 0.002	0.101 ± 0.000	0.075 ± 0.012	0.086 ± 0.002	0.123 ± 0.002	0.134 ± 0.002
C5	0.105 ± 0.008	0.100 ± 0.000	0.100 ± 0.014	0.100 ± 0.001	0.084 ± 0.009	0.067 ± 0.006	0.098 ± 0.002	0.121 ± 0.001
C6	0.096 ± 0.006	0.100 ± 0.000	0.103 ± 0.004	0.100 ± 0.000	0.075 ± 0.017	0.054 ± 0.003	0.092 ± 0.005	0.104 ± 0.002
C7	0.102 ± 0.012	0.100 ± 0.000	0.096 ± 0.002	0.099 ± 0.000	0.097 ± 0.009	0.044 ± 0.001	0.068 ± 0.001	0.083 ± 0.001
C8	0.092 ± 0.015	0.100 ± 0.000	0.084 ± 0.013	0.099 ± 0.000	0.089 ± 0.006	0.045 ± 0.005	0.050 ± 0.003	0.057 ± 0.003
C9	0.103 ± 0.006	0.100 ± 0.000	0.080 ± 0.002	0.099 ± 0.000	0.076 ± 0.004	0.037 ± 0.002	0.029 ± 0.003	0.030 ± 0.003
C10	0.098 ± 0.007	0.100 ± 0.000	0.071 ± 0.004	0.099 ± 0.000	0.082 ± 0.007	0.034 ± 0.001	0.020 ± 0.004	0.009 ± 0.001

Table 10: Computational cost comparison on CIFAR-10 in seconds. Final runtime denotes the time required to obtain one final contribution vector for all clients after an FL run. Per-round runtime denotes the average time required to produce one round-wise contribution during training.

		5 Clients		10 Clients	
Method	Round-wise Scores	Final Runtime (s)	Per-Round Runtime (s)	Final Runtime (s)	Per-Round Runtime (s)
Exact Shapley (SV)	No	29,475	–	1,708,123	–
GTG-Shapley	Yes	2,451	61	11,369	284
FedPID	Yes	2,739	68	6,299	157

D.3.1 Larger Client Pools with Full and Partial Participation

We further evaluate FedPID with $K = 20$ and $K = 50$ clients under full and partial participation. For IID, quantity skew and feature-and-label noise, partial participation samples $m = 10$ clients uniformly without replacement in each round. We use CIFAR-10 over three seeds, following the training, validation, and PID settings in Section 5.1 and Appendix C. PID is measured every round during training. Quantity skew is evaluated under full and partial participation, with target client sample counts in the ratio $(K + 1) : K : \cdots : 2$, following Appendix B.

To account for unequal participation counts, we normalise client scores within each round and then average over the rounds in which each client participates. Let T_k denote those rounds and $m = |\mathcal{S}^{(t)}|$, which is fixed within each experiment. With the scalar score $s_k^{(t)}$ from Section 4.3, define

$$q_k^{(t)} = \frac{s_k^{(t)}}{\sum_{\ell \in \mathcal{S}^{(t)}} s_\ell^{(t)}}, \qquad \bar{q}_k = \frac{1}{|T_k|} \sum_{t \in T_k} q_k^{(t)}. \tag{26}$$

With 20 or 50 clients per pool, reporting every client's score would make the table difficult to read. We therefore summarise contribution patterns using mean absolute deviation from uniform scores under IID and Spearman correlations with client sample count and corruption order under quantity skew and noise, respectively.

Under IID, we report the mean absolute deviation $\mathrm{MAD} = K^{-1} \sum_{k=1}^{K} |\bar{q}_k - 1/m|$, where zero denotes equal contribution. Under quantity skew, we report the Spearman correlation between $\bar{q}_k$ and client sample count. Under feature-and-label noise, we report the Spearman correlation between $\bar{q}_k$ and increasing corruption order, where -1 denotes perfectly decreasing contribution. We use this same normalisation and averaging order for full and partial participation; the participation means are not renormalised across the full client pool. Thus, the uniform reference is $1/10$ under partial participation and $1/K$ under full participation.

We also evaluate multiple specialist clients under partial participation, using four specialists in the $K = 20$ pool and ten in the $K = 50$ pool, with five active clients per round. Specialists hold only classes 0–1; the remaining clients hold only classes 2–9. The training set is partitioned into equally sized, non-overlapping client datasets. For each focal specialist, we group its participating rounds according to whether the peer tuple contains another specialist. We average the PID components over classes 0–1 and all focal-specialist appearances within each group, then report the mean and standard deviation across three seeds.

Table 11: Client contribution evaluation in larger CIFAR-10 pools. (a) Scalar-score diagnostics under full participation or ten active clients per round. (b) Specialist-class PID components in bits with five active clients per round. Values are mean ± standard deviation over three seeds.

(a) IID, quantity skew and feature-and-label noise

Pool K	Participation	Active m	IID MAD	Quantity skew Spearman ρ	Noise Spearman ρ
20	Partial	10	0.00074 ± 0.00013	0.874 ± 0.020	-0.995 ± 0.002
20	Full	20	0.00039 ± 0.00003	0.866 ± 0.020	-0.994 ± 0.003
50	Partial	10	0.00212 ± 0.00036	0.777 ± 0.033	-0.986 ± 0.005
50	Full	50	0.00031 ± 0.00002	0.922 ± 0.009	-0.993 ± 0.002

(b) Multiple specialists under partial participation

Pool K	Specialists	Another specialist in peer tuple	Unique	Redundant	Synergistic
20	4	Absent	0.1285 ± 0.0105	0.0000 ± 0.0000	0.0000 ± 0.0000
20	4	Present	0.0080 ± 0.0022	0.1059 ± 0.0037	0.0098 ± 0.0015
50	10	Absent	0.0892 ± 0.0068	0.0000 ± 0.0000	0.0000 ± 0.0000
50	10	Present	0.0064 ± 0.0017	0.0707 ± 0.0191	0.0082 ± 0.0006

Table 11(a) shows that IID contribution scores remain close to uniform under both full and partial participation. Under quantity skew, scores are positively associated with client sample count under both participation settings: mean Spearman correlations are 0.866 and 0.922 under full participation, and 0.874 and 0.777 under partial participation, for $K = 20$ and 50, respectively. Under feature-and-label noise, FedPID assigns progressively smaller contributions to increasingly corrupted clients. Table 11(b) shows that the focal specialist's information about classes 0–1 is predominantly unique when no other specialist participates, but predominantly redundant when the peer tuple contains another specialist. This pattern holds in the mean for all three seeds at both pool sizes, showing how the profile changes with the participating peers. These results support FedPID's use beyond fully participating cross-silo FL in the tested client pools.

D.3.2 Validation-set Stress Tests

We further evaluate FedPID under IID, quantity skew, non-IID and feature-and-label noise using five validation-set settings. The smaller sets contain 500 or 1,000 class-balanced samples (5% or 10% of the official CIFAR-10 test split). The class-imbalanced set contains 1,000 samples (10%), with unequal class counts sampled using a symmetric Dirichlet distribution with $\alpha = 1$. For label corruption, we use the original class-balanced set of 3,000 samples (30%) and replace 300 or 600

randomly selected labels (10% or 20%) with labels sampled uniformly from the incorrect classes. The remaining settings follow Appendix C.

Table 12: FedPID$_L$ client contribution scores under different validation-set settings on CIFAR-10. Size percentages refer to the official test split; class distributions are shown before label corruption.

Validation size	Class distribution	Label corruption	C1	C2	C3	C4	C5
IID							
500 (5%)	Balanced	0%	0.199 ± 0.001	0.200 ± 0.001	0.201 ± 0.001	0.202 ± 0.002	0.199 ± 0.002
1,000 (10%)	Balanced	0%	0.200 ± 0.001	0.200 ± 0.002	0.200 ± 0.001	0.200 ± 0.002	0.199 ± 0.002
1,000 (10%)	Dirichlet ($\alpha = 1$)	0%	0.199 ± 0.001	0.202 ± 0.001	0.199 ± 0.001	0.200 ± 0.002	0.201 ± 0.001
3,000 (30%)	Balanced	10%	0.200 ± 0.000	0.200 ± 0.000	0.200 ± 0.001	0.201 ± 0.001	0.200 ± 0.001
3,000 (30%)	Balanced	20%	0.200 ± 0.001	0.200 ± 0.001	0.200 ± 0.002	0.200 ± 0.001	0.200 ± 0.001
Quantity skew							
500 (5%)	Balanced	0%	0.201 ± 0.001	0.202 ± 0.003	0.200 ± 0.001	0.200 ± 0.002	0.199 ± 0.002
1,000 (10%)	Balanced	0%	0.202 ± 0.002	0.202 ± 0.002	0.199 ± 0.001	0.198 ± 0.001	0.199 ± 0.001
1,000 (10%)	Dirichlet ($\alpha = 1$)	0%	0.202 ± 0.001	0.202 ± 0.002	0.199 ± 0.001	0.198 ± 0.001	0.199 ± 0.002
3,000 (30%)	Balanced	10%	0.201 ± 0.000	0.201 ± 0.002	0.200 ± 0.001	0.200 ± 0.001	0.198 ± 0.000
3,000 (30%)	Balanced	20%	0.201 ± 0.001	0.202 ± 0.001	0.200 ± 0.001	0.199 ± 0.001	0.198 ± 0.001
Non-IID							
500 (5%)	Balanced	0%	0.152 ± 0.005	0.213 ± 0.002	0.213 ± 0.001	0.212 ± 0.001	0.211 ± 0.003
1,000 (10%)	Balanced	0%	0.151 ± 0.009	0.214 ± 0.001	0.212 ± 0.003	0.214 ± 0.002	0.210 ± 0.004
1,000 (10%)	Dirichlet ($\alpha = 1$)	0%	0.119 ± 0.183	0.221 ± 0.045	0.220 ± 0.045	0.221 ± 0.046	0.219 ± 0.048
3,000 (30%)	Balanced	10%	0.150 ± 0.007	0.213 ± 0.002	0.211 ± 0.000	0.213 ± 0.003	0.212 ± 0.003
3,000 (30%)	Balanced	20%	0.144 ± 0.015	0.214 ± 0.004	0.214 ± 0.002	0.215 ± 0.004	0.214 ± 0.005
Feature-and-label noise							
500 (5%)	Balanced	0%	0.588 ± 0.011	0.224 ± 0.008	0.114 ± 0.012	0.050 ± 0.004	0.024 ± 0.002
1,000 (10%)	Balanced	0%	0.624 ± 0.020	0.207 ± 0.006	0.106 ± 0.012	0.043 ± 0.004	0.020 ± 0.002
1,000 (10%)	Dirichlet ($\alpha = 1$)	0%	0.667 ± 0.082	0.183 ± 0.040	0.096 ± 0.032	0.037 ± 0.010	0.017 ± 0.002
3,000 (30%)	Balanced	10%	0.624 ± 0.017	0.200 ± 0.010	0.108 ± 0.011	0.046 ± 0.001	0.022 ± 0.003
3,000 (30%)	Balanced	20%	0.600 ± 0.020	0.211 ± 0.006	0.113 ± 0.014	0.052 ± 0.006	0.026 ± 0.001

Table 12 shows that IID contribution scores remain close to uniform across clients under all five validation-set settings. Under quantity skew, final contribution scores also remain close to uniform. Under feature-and-label noise, FedPID assigns progressively smaller contributions to increasingly corrupted clients, preserving the expected ordering.

We also examine FedPID$_F$ under the same five validation set settings. Its scores remain balanced under IID. Under quantity skew, mean first-round scores follow the client data-size ordering. Under feature-and-label noise, scores decrease with increasing client corruption in all three seeds for every validation set setting. Overall, these results support the robustness of FedPID's contribution estimates under IID and feature-and-label noise across the tested validation set settings.

D.3.3 Validation-set Sensitivity of Client Aggregation

We further evaluate FedPID$_F$ and FedPID$_R$ against validation-accuracy and validation-loss weighting under IID, quantity skew, non-IID and feature-and-label noise. We use the 5% and 10% balanced validation sets and the 10% class-imbalanced validation set defined in Appendix D.3.2, reporting balanced-validation-accuracy weighting separately only for the class-imbalanced set. On the balanced sets, the two accuracy criteria are equivalent, so we retain the validation-accuracy results as the single accuracy-weighting baseline. The remaining training settings follow Appendix C.

MI and CMI are included as coefficient settings within FedPID (Appendix D.5). For each FedPID variant and validation-set setting, we select one configuration by the highest mean validation accuracy across three seeds. Within each run, the checkpoint is selected by validation accuracy, and its test accuracy is used for reporting only.

Table 13: Client-aggregation test accuracy (%) on CIFAR-10 with smaller and class-imbalanced validation sets. Values are mean ± standard deviation over three seeds for the checkpoint with the highest validation accuracy in each run. Validation sizes refer to the official test split. Dashes indicate that balanced-accuracy weighting is represented by the equivalent validation-accuracy criterion on balanced sets.

Aggregation method	500 (5%) Balanced	1,000 (10%) Balanced	1,000 (10%) Dirichlet ($\alpha = 1$)
IID			
Validation accuracy	92.40 ± 0.43	91.99 ± 0.36	92.25 ± 0.32
Balanced validation accuracy	—	—	92.18 ± 0.39
Validation loss	$\mathbf{92.48 \pm 0.34}$	92.26 ± 0.34	92.25 ± 0.07
FedPID$_F$	92.05 ± 0.51	92.16 ± 0.69	92.47 ± 0.42
FedPID$_R$	92.17 ± 0.34	$\mathbf{92.60 \pm 0.44}$	$\mathbf{92.49 \pm 0.20}$
Quantity skew			
Validation accuracy	92.44 ± 0.31	92.51 ± 0.29	92.46 ± 0.23
Balanced validation accuracy	—	—	92.71 ± 0.14
Validation loss	92.53 ± 0.29	92.48 ± 0.25	92.61 ± 0.34
FedPID$_F$	$\mathbf{92.73 \pm 0.21}$	$\mathbf{92.81 \pm 0.19}$	92.61 ± 0.18
FedPID$_R$	92.69 ± 0.16	92.70 ± 0.23	$\mathbf{92.77 \pm 0.43}$
Non-IID			
Validation accuracy	71.24 ± 0.47	71.24 ± 0.47	71.49 ± 0.68
Balanced validation accuracy	—	—	71.12 ± 0.34
Validation loss	70.35 ± 0.47	70.42 ± 0.47	70.95 ± 0.86
FedPID$_F$	71.31 ± 0.26	71.38 ± 0.04	$\mathbf{73.49 \pm 3.14}$
FedPID$_R$	$\mathbf{71.45 \pm 0.51}$	$\mathbf{74.64 \pm 1.29}$	72.76 ± 1.68
Feature-and-label noise			
Validation accuracy	73.30 ± 1.20	73.92 ± 1.47	75.65 ± 0.12
Balanced validation accuracy	—	—	74.42 ± 0.93
Validation loss	75.34 ± 1.46	76.03 ± 0.54	76.77 ± 0.59
FedPID$_F$	$\mathbf{84.88 \pm 0.13}$	$\mathbf{85.46 \pm 0.54}$	$\mathbf{85.18 \pm 0.98}$
FedPID$_R$	84.64 ± 0.95	85.01 ± 1.08	84.87 ± 0.84

Table 13 shows that FedPID$_F$ and FedPID$_R$ outperform all reported validation-based baselines under feature-and-label noise across the smaller and class-imbalanced validation sets. FedPID$_F$ achieves 84.88–85.46%, compared with 75.34–76.77% for validation-loss weighting, the strongest validation-based baseline. Under non-IID data, the best FedPID variant also obtains higher mean accuracy than the validation-based baselines in all three validation set settings. Under IID and quantity skew, accuracies remain closely grouped across methods. These results show that FedPID-guided aggregation retains its advantage for noisy clients with smaller and class-imbalanced validation sets, while remaining competitive in the other settings.

D.4 Experimental Results on ImageNet-100

Table 14 reports FedPID scores measured at the first round under IID, quantity-skew, and feature-and-label-noise settings. The resulting client contributions follow the expected patterns. Under IID, the scores are close to uniform across clients. Under quantity skew, the scores increase with client data volume, assigning the largest contribution to the client with the largest local dataset and the smallest contribution to the client with the smallest local dataset. Under feature-and-label noise, FedPID assigns most of the contribution to the uncorrupted client and gives very small scores to the heavily corrupted clients. This stronger separation on ImageNet-100 suggests that, as the task becomes more complex, FedPID remains sensitive to client quality and does not assign high contribution to clients whose local signal is dominated by corrupted features and labels.

Table 14: Client contribution scores across ImageNet-100 settings. For the feature-and-label-noise setting, $\sigma_0 = 0.1$ and $\rho_0 = 0.25$. Expected contribution patterns are near-uniform in IID, decreasing for quantity skew and feature-and-label noise.

Client	IID	Quantity Skew	Feature + Label Noise
C1	0.194 ± 0.001	0.363 ± 0.028	0.909 ± 0.008
C2	0.201 ± 0.014	0.245 ± 0.009	0.078 ± 0.005
C3	0.199 ± 0.008	0.191 ± 0.010	0.010 ± 0.002
C4	0.200 ± 0.011	0.134 ± 0.012	0.001 ± 0.001
C5	0.206 ± 0.001	0.067 ± 0.005	0.001 ± 0.000

D.5 Hyperparameter Tuning

PID Measurement Frequency. We first evaluate whether FedPID should recompute the PID decomposition throughout training. Table 15 compares FedPID$_F$ with FedPID$_R$ using validation accuracy and the default aggregation coefficients with four bins. Under this fixed-coefficient tuning comparison, FedPID$_F$ matches or slightly outperforms FedPID$_R$ across all four CIFAR-10 settings. The gap is small in IID and quantity skew, indicating that the PID scores measured at the first round are already sufficient when client structure is stable. The difference is larger under feature-and-label noise, where the behaviour is consistent with the client scores in Table 1: in the feature-and-label-noise setting, the uncorrupted client receives a larger score measured at the first round than at the last round, while the corrupted clients receive higher scores at the last round. As training proceeds, noisy clients can partially benefit from repeated federation with the global model and the other clients, making their late-round predictions appear more informative despite corrupted local data. Using PID scores measured at the first round therefore preserves a sharper estimate of underlying client quality.

Table 15: Validation accuracy (%) under two PID measurement schedules across CIFAR-10 settings.

Method	IID	Quantity Skew	Feature + Label Noise	Non-IID
FedPID$_F$	$\mathbf{93.33 \pm 0.06}$	$\mathbf{93.63 \pm 0.09}$	$\mathbf{84.00 \pm 0.33}$	$\mathbf{70.43 \pm 0.42}$
FedPID$_R$	93.24 ± 0.22	93.55 ± 0.11	82.16 ± 1.02	69.97 ± 0.45

Bin count. We next study the sensitivity of FedPID to the number of discretisation bins used for PID estimation. Table 16 reports client contribution scores measured at the first round for two representative heterogeneous settings. Under feature-and-label noise, all bin counts identify the same qualitative structure: the least corrupted client receives the dominant contribution, while the most corrupted clients are assigned near-zero scores. Two bins produce the sharpest separation, whereas four and eight bins distribute slightly more contribution to intermediate-quality clients, but the ordering remains consistent. Under non-IID data, the scores remain broadly distributed across clients for all bin counts, showing that the decomposition does not collapse to a single client as the bin resolution changes. The exact allocation varies with the discretisation level, but the contribution pattern remains stable. Table 17 further evaluates the effect of bin count on downstream validation accuracy in the feature-and-label-noise setting. The two-bin configuration obtains the highest validation accuracy for both FedPID$_F$ and FedPID$_R$, while performance decreases as the number of bins increases. This trend is consistent with the estimation trade-off in discretised information measures: with a fixed validation set, increasing the number of bins increases the number of empirical joint states that must be estimated, making the PID estimate sparser and potentially noisier. Although a larger validation set could reduce this sparsity, this will not always be feasible. We therefore keep the validation/test split fixed and use the bin-count tuning to assess which discretisation is most stable under the same evaluation budget. FedPID$_F$ remains stronger than FedPID$_R$ for every bin count, reinforcing the observation that a stable estimate from the PID scores measured at the first round is effective in noisy-client settings.

Coefficient selection. For contribution-evaluation comparisons, we use $(\alpha, \beta, \gamma) = (1, 0.15, 0.15)$ as a fixed heuristic that gives greater weight to information unique to a client, because this term captures predictive information about the target available from that client individually but not from its peers alone. For downstream aggregation, we select the scalarisation coefficients using validation performance. Table 18 compares four choices of (α, β, γ), corresponding to the weights on unique, redundant, and synergistic information. In each class-conditioned two-source PID, $(1, 1, 0)$ gives

Table 16: Client contribution scores at the first round across CIFAR-10 settings under different PID bin counts. For the feature-and-label-noise setting, $\sigma_0 = 0.1$ and $\rho_0 = 0.25$.

Client	Feature + Label Noise			Non-IID		
	2 bins	4 bins	8 bins	2 bins	4 bins	8 bins
C1	0.875 ± 0.012	0.736 ± 0.012	0.673 ± 0.036	0.211 ± 0.020	0.233 ± 0.007	0.194 ± 0.009
C2	0.108 ± 0.010	0.141 ± 0.007	0.170 ± 0.025	0.206 ± 0.020	0.225 ± 0.009	0.220 ± 0.013
C3	0.015 ± 0.002	0.079 ± 0.002	0.100 ± 0.003	0.190 ± 0.015	0.178 ± 0.007	0.202 ± 0.007
C4	0.001 ± 0.000	0.033 ± 0.000	0.035 ± 0.006	0.156 ± 0.012	0.163 ± 0.013	0.171 ± 0.009
C5	0.001 ± 0.000	0.011 ± 0.003	0.022 ± 0.003	0.237 ± 0.031	0.200 ± 0.014	0.214 ± 0.007

Table 17: Validation accuracy (%) under different bin counts for CIFAR-10 feature-and-label noise.

Setting	FedPID$_F$			FedPID$_R$		
	2 bins	4 bins	8 bins	2 bins	4 bins	8 bins
Feature + Label Noise	85.46 ± 0.34	84.00 ± 0.33	83.08 ± 0.27	84.43 ± 0.67	82.16 ± 1.02	80.18 ± 0.18

Table 18: Validation accuracy (%) under different PID atom coefficients across CIFAR-10 settings.

Setting	FedPID$_F$				FedPID$_R$			
	$(1, 0.15, 0.15)$	$(1, 0.5, 0.5)$	$(1, 1, 0)$	$(1, 0, 1)$	$(1, 0.15, 0.15)$	$(1, 0.5, 0.5)$	$(1, 1, 0)$	$(1, 0, 1)$
Feature + Label Noise	84.00 ± 0.33	80.88 ± 0.22	79.67 ± 1.11	83.48 ± 0.13	82.16 ± 1.02	79.32 ± 0.79	78.44 ± 0.35	82.56 ± 0.19
Non-IID	70.43 ± 0.42	71.67 ± 0.34	72.01 ± 0.29	70.04 ± 0.82	69.97 ± 0.45	71.97 ± 0.19	72.41 ± 0.41	70.74 ± 0.20

the mutual information (MI), $U_k^{(c)} + R^{(c)} = I(Y^{(c)}; Z_k^{(c)})$, while $(1, 0, 1)$ gives the conditional mutual information (CMI), $U_k^{(c)} + S^{(c)} = I(Y^{(c)}; Z_k^{(c)} \mid Z_{-k}^{(c)})$. After class-prior aggregation, these become weighted sums of one-vs-rest marginal or conditional information terms, rather than a single multiclass mutual information. Under the feature-and-label-noise setting, the default choice $(1, 0.15, 0.15)$ achieves the best validation accuracy when using weights derived from PID scores measured at the first round, while larger weights on redundant or synergistic information reduce performance. This is expected because noisy clients can share corrupted or weakly useful information, so heavily rewarding shared atoms may overestimate their contribution. Under the non-IID setting, assigning full weight to unique and redundant information, $(1, 1, 0)$, gives the highest validation accuracy, especially under per-round recomputation. This reflects the different role of redundancy when clients contain reliable but uneven class support. Overall, the best scalar weighting depends on the heterogeneity type; since our main focus is contribution evaluation rather than exhaustive aggregation tuning, we treat coefficient selection as a limited sensitivity study, with Table 3 showing that this small grid already gives competitive downstream aggregation.