Contribution Evaluation in Federated Learning: Examining Current Approaches

Vasilis Siomos
Department of Computing
Imperial College London
`vs1820@ic.ac.uk`

Jonathan Passerat-Palmbach
Imperial College London
ConsenSys Health
`j.passerat-palmbach@imperial.ac.uk`

Abstract

Federated Learning (FL) has seen increasing interest in cases where entities want to collaboratively train models while maintaining privacy and governance over their data. In FL, clients with private and potentially heterogeneous data and compute resources come together to train a common model without raw data ever leaving their locale. Instead, the participants contribute by sharing local model updates, which, naturally, differ in quality. Quantitatively evaluating the worth of these contributions is termed the Contribution Evaluation (CE) problem. We review current CE approaches from the underlying mathematical framework to efficiently calculate a fair value for each client. Furthermore, we benchmark some of the most promising state-of-the-art approaches, along with a new one we introduce, on MNIST and CIFAR-10, to showcase their differences. Designing a fair and efficient CE method, while a small part of the overall FL system design, is tantamount to the mainstream adoption of FL.

1   Introduction

Federated Learning [1] has emerged as an appealing paradigm of collaborative model training in the digital privacy age. Its appeal stems from allowing distributed computation on fragmented, privately owned data without sacrificing privacy, or performance by utilising the exchange of gradients/model weights instead of raw data. Initially developed to target massive numbers of low-resource edge devices [2], it has since found application in real-world cross-institutional settings [3, 4], too. FL creates a path for the formation of consortia between competing companies since, under privacy guarantees, if the global model is better than the individual ones, participating companies stand to benefit. These two settings are commonly referred to as cross-device and cross-silo, respectively [5].

Fairness in FL comprises two notions: the first, analogous to regular ML fairness, is ensuring the model is not biased against any group of participants. Here, we focus on the second, often disregarded, notion of fairly compensating clients for their contributions. In mainstream FL research, it is ubiquitous to assume all clients unconditionally want to participate and remain in the federation, meaning their compensation is de-coupled from the training process (e.g., participants in Google's Gboard deployment received no external reward apart from the model [2], while the MELLODDY [3] consortium participants receive arbitrary – from the perspective of the model – EU funding). Realistically, however, *potential* clients have the agency to decide whether to join and remain in the federation based on how they, and their collaborators, will be rewarded for it. In cross-device FL, reluctance to participate is justified by the personal nature of the data, meaning users are reluctant to share it altruistically, even under privacy guarantees. In cross-silo FL, participants take a business risk and incur data curation and training costs and thus expect fair compensation. This fact, coupled with the smaller number of participants, all of which are available and persist throughout training, mean CE is both more feasible and more needed in cross-silo settings, and thus this is where most current work focuses.

The problem of assigning a numeric value to what participants share with the federation is termed Contribution Evaluation (CE), and is part of the problem of designing the overall mechanism that motivates client participation, called Incentive Mechanism Design. A *payoff scheme* which guarantees clients are compensated in fair proportion to their contributions is necessary to convince them to join the federation. CE is closely related to similar evaluation problems in data valuation and feature attribution for explainable AI, treating data as a commodity, and creating data marketplaces, so understanding current FL approaches necessarily involves examining non-federated data valuation methods. However, CE in FL introduces additional challenges, which we will aim to highlight.

2 Game Theory background

The mathematical framework used in data/contribution evaluation is borrowed from Coalitional Game Theory (CGT), well-suited to model client interactions and how they lead to the formation of coalitions of cooperating parties. CE can thus be framed as solving a game G with N participating clients, where outcomes consist of two elements. The first is a partitioning of the players into disjoint coalitions, each of which is mapped to a real number representing its value (also called *utility*) using a *characteristic function* v. Most often, v is chosen to be the coalitional model's accuracy on a held-out validation set. The second is a payoff vector, $\mathbf{x} = (x_1, ..., x_n)$, distributing the utility of each coalition to its participants. Generally, x_i are non-negative, and the L1 norm of $\mathbf{x}$ is only upper-bounded by the utility $\left(\sum_{i \in C^j} x_i \leq v(C^j) \forall j \in 1, ...k\right)$; if the equality holds it is called *efficient*. If an efficient payoff vector satisfies the *individual rationality* condition of $x_i \geq v(\{i\}) \forall i \in N$, it is called an *imputation*. Intuitively, an imputation is a set of payoffs that exactly splits the value of the coalition to each member in a manner in which it is sensible for all of them to remain in the coalition.

A solution concept σ is a mapping from G to a set of such outcomes, given a notion of our desired goal, such as fairness, or ensuring a single coalition, including all clients (called the grand coalition), forms. Such outcomes of G (if they exist), are called rational under σ. Fair CE consists then of solving G for fairness, choosing a fairness-oriented σ, and computing its rational outcomes, resulting in the set of payoff vectors that fairly represent the worth of participants' contributions to the federation. How to best incorporate the numeric values $\mathbf{x_i}$ in the FL system design is the purview of Incentive Mechanism Design and outside our scope.

G is *superadditive* if $v(C \cup D) \geq v(C) + v(D) \forall$ disjoint $C, D \subseteq N$, leading the grand coalition to always form, since it is always non-detrimental to grow the coalition. In fact, unconditionally assuming the grand coalition forms is valid *only* under this property, otherwise solving for *stability*, i.e., convincing clients to join and stay in our federation, instead of fairness, must be considered. The following well-known solution concepts each focus on one of these two goals.

The *Core* is the set of all imputations under which no coalition would receive a higher payoff by leaving the grand coalition ($x(C) \geq v(C) \forall C \subseteq N$). It ensures the stability of the grand coalition and includes all *realistic* payoffs, as imputations outside it are not viable if players are free to leave the grand coalition. However, the Core is not guaranteed to be non-empty or unique, for the imputations that lie in it (if they exist), no preference is defined, and they might be unfair for some players. Note that, even then, a player's Core payoff is still higher than their unitary utility.

If instead, we desire to calculate a fair measure for the value of G from the perspective of each client, we can use their marginal contribution to a coalition S to calculate their worth; calculated either post-addition, as $v(S) - v(S \backslash \{i\})$, or pre-addition, as $(v(S \cup \{i\}) - v(S))$. Furthermore, we can consider contributions of individual data points: the well-known statistical Leave-One-Out (LOO) metric is the post-addition marginal contribution of a point. In our setting, the LOO is intractable, but marginal contributions are sometimes referred to as client-level LOO. Alone, the marginal contribution does not adjust for the order in which participants join the coalition, or for the size of the coalition at that point. In contrast, Shapley [6] defined four axioms of fairness that he believed a fair measure, i.e. a payoff vector, ϕ should satisfy:

1. **Efficiency/Pareto Optimality**: $\sum_{i \in N} \phi_i(v) = v(N)$
2. **Symmetry**: If $v(S \cup \{i\}) = v(S \cup \{j\}) \forall S \subseteq N \backslash \{i, j\}$, then $\phi_i(v) = \phi_j(v)$
3. **Null Player**: If $v(S \cup \{i\}) = v(S) \forall S \subseteq N \backslash \{i\}$, then $\phi_i(v) = 0$
4. **Additivity**: If u and v are characteristic functions, then $\phi(u + v) = \phi(u) + \phi(v)$.

The only [7] payoff vector satisfying them all is called the Shapley Value (SV) [6]. For a coalition N and the set of all characteristic functions $\mathcal{V}$ on 2^N, the Shapley Value ϕ on $\mathcal{V}$ is given by [8]:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} \left(v\left(S \cup \{i\}\right) - v\left(S\right)\right) \quad \forall v \in \mathcal{V}, i \in N$$

We must emphasize the conceptual difference between the Core and the SV. To calculate the latter, we assume all the players have agreed to join the game, or, equivalently, that the game is super-additive, and we take on the role of each of the players, calculating the fair value of G according to their marginal contributions, producing ϕ_i. Instead the Core includes the viable payoffs a central entity can deliver to each player to keep them in the grand coalition.

3   Existing contribution evaluation approaches

Designing a fair CE method for FL can be broken down to the following problems:

Choosing a solution concept and efficiently computing the imputation(s) of G under it. Directly calculating either the SV or the Core is exponentially complex $O(2^N)$ w.r.t to the number of clients. Even in the less crowded cross-silo setting, each calculation requires fully training the underlying model using the data from the sampled client subset, meaning finding exact solutions is practically intractable for models with long training times, like neural networks.

Adapting to FL. The entire theoretical framework based on CGT, and the resulting methods, were first developed to tackle data valuation. Contrary to that setting, in FL, model updates are being shared instead of raw data, which are potentially obfuscated/encrypted for privacy, and are only becoming available to the server sequentially at every round. Thus, adapting methods is non-trivial and in some instances impossible, as we will see.

Balancing utility and valuation accuracy. CE is, at its core, a data valuation problem, yet FL minimizes/eliminates the exchange of data. The more a CE method burdens clients with calculations or the more data it requires from them, the less compatible it becomes with the overall FL pipeline, while on the other hand FL obfuscation measures like secure aggregation [9] inhibit accurate valuation.

3.1   Evaluation using the Shapley Value

If we accept Shapley's axioms as a comprehensive set of desirable properties for a CE metric, then the SV is unique in satisfying them and thus the de facto metric to calculate. Since its exact computation is intractable, several data valuation methods focus on efficiently and accurately approximating it. These are not all directly transferable to FL, in fact, most are not, but nevertheless understanding them is necessary to move to FL adaptations. The seminal work in [10] presented Data Shapley, an approximation based on Monte Carlo sampling; by substituting the $1/n$ constant with $1/n!$ in the definition of SV, and using the uniform distribution over all possible permutations of the data Π, approximating the SV equates to finding an expected value, a problem lending itself to sampling-based solutions: $\phi_i' = \frac{1}{n!} \sum_{S \subseteq N/\{i\}} \frac{V(S \cup \{i\}) - V(S)}{\binom{n-1}{|S|}} = \mathbb{E}_{\pi \sim \Pi}\left[V(S_\pi^i \cup \{i\}) - V(S_\pi^i)\right]$, where S_π^i is the set of data points coming before datum i in permutation π. Sampling permutations $\pi \sim \Pi$ and sequentially evaluating the marginal contributions of each sample i to S_π^i, before adding them to the list of training data, leads, after many iterations, to average marginal contributions which are unbiased estimates of the SVs. The intuition behind Data Shapley is that, when using the performance on a validation/test set $\hat{T}$ as the characteristic function, this set will always be finite and drawn from the true test distribution T, and thus, a bootstrapped approximation of the SV up to the intrinsic observation noise that comes from drawing $\hat{T} \sim T$ is satisfactory. The Distributional SV [11] stabilises the SV w.r.t. perturbations of the training set; whereas the SV is only valid for a single draw of training samples, the Distributional SV is an unbiased estimate of the value of adding a datum to any dataset drawn from the underlying distribution. A similar repertoire of approximations was proposed in [12], most interesting of which is using the combinatorial technique of group-based testing, with group sampling chosen to allow efficient approximations of the pairwise differences of SVs between clients, from which the individual SV approximations can then be efficiently recovered.

These algorithms need all the data to be simultaneously available, a prerequisite not satisfied in the cross-device setting, and one which introduces a big computational and communication overhead, in

the cross-silo one. Furthermore, in both settings, Shapley's symmetry axiom becomes a hindrance for any federation where not every client joins simultaneously: in that case 'early adopter' participants should be rewarded more, for helping bootstrap the global model. Using the SV, not only is that not accomplished, but a client can even join the process late, with a local dataset that is a duplicate of another, and his SV payoff would be the same as the early adopter with the same dataset.

Recognising this shortcoming, and extending/modifying the SV for FL, an approach termed Federated Shapley was proposed in [13]: Let $I = \{1, ..., N\}$ denote the set of participants selected by the coordinator during a T-round FL process. Let I_t be the set of participants selected in round t, $I_{j:k}$ the selection sets between rounds j, k and $I_t \subseteq I$. Then, the federated SV of participant i at round t is defined as:

$$s_t^v(i) = \frac{1}{|I_t|} \sum_{S \subseteq I_t \setminus \{i\}} \frac{1}{\binom{|I_t - 1|}{|S|}} \left[v(I_{1:t-1} + (S \cup \{i\})) - v(I_{1:t-1} + S) \right] \; if \; i \in I_t, else \; 0,$$

and the federated SV is their sum across rounds $s^v(i) = \sum_{t=1}^T s_t(i)$. So, at every round, for every participant, the average marginal contribution across all possible coalitions is used to get a *temporal snapshot* of the values of participants' datasets. Straight-forward adaptations of the original fairness axioms hold for the federated SV [13]. Exact computation is even more costly than that of the SV ($O(T2^m)$ complexity, where m the maximum number of participants in a round), and hence MC sampling or group-based testing approximations [12] are still used in the original publication.

Another FL-compatible SV approximation was presented in [14], which *only* uses the gradients produced during training the grand coalition's model to approximately reconstruct all other models necessary for the SV calculation. Whereas for the exact SV computation we need to retrain a model on all possible combinations of contributing clients, meaning clients expend time and resources on training $2^n - 2$ useless models, using this pseudo-model approximation, only the useful model, generated using all participant data, is trained. Pseudo-models are constructed at every round by combining the marginal contributions (pre-addition) of the clients to the global model *of the previous round*. Three distinct algorithms to achieve that are proposed:

- One-Round (OR): The server uses the marginal contributions, normalized by local dataset sizes, to update the 2^N pseudo-models stored at the server at every round. The SV for each client is calculated after training concludes, using the final pseudo-models.
- λ-Multi-Rounds (λ-MR): A so-called round-CI (originally presented by the same authors in [15]) is calculated at every round, following the OR procedure, using the latest pseudo-models. After training, round-CIs are weighted by a decay factor which emphasizes earlier contributions, normalized by their per-round sum, and aggregated, to produce the SVs.
- Truncated Multi-Rounds: λ-MR can be sped up by adding a threshold for the decay λ, below which updating the pseudo-models stops. The aggregation formula for the SV now also weighs round-CIs proportionally to the test accuracy of the global model at the corresponding round, emphasizing rounds where test accuracy was high.

A key connection we make here is noticing that round-CI and Federated Shapley $s_t^v(i)$ refer to the same quantity, albeit the approximations used to calculate each of them differ conceptually. λ-MR focuses on earlier contributions, which, arguably, are more important than later ones in steering the loss of the global model towards a better minimum, via the time decay coefficient.

3.2 Alternative Methods of Contribution Evaluation

While the SV is arguably over-represented in the literature, other solution concepts have not been studied extensively. Recently, Yan and Procaccia [16] examined using the Core and a related solution concept called the Nucleolus for data evaluation. As, like SV, the Core is prohibitively expensive to compute exactly, we can choose to relax it in several ways, allowing for efficient computation. Extending it to include imputations where no actor benefits more from leaving the grand coalition, if they have to pay a cost ϵ for leaving, we arrive at the ϵ-Core: $C_\epsilon(v) = \{x \in R^N : \sum_{i \in N} x_i = v(N); \sum_{i \in S} x_i \geq v(S) - \epsilon, \forall S \subseteq N\}$. This is a logical relaxation in cross-silo/enterprise FL, where each client will have signed a participation contract before joining, and the expenses of breaking it to leave the federation play the role of ϵ. The *least* core is the solution of the linear program minimizing ϵ:

$$\min \epsilon \; \textbf{s.t.} \; \sum_{i \in N} x_i = v(N) \; \text{and} \; \sum_{i \in S} x_i + \epsilon \geq v(S) \quad \forall S \subseteq N$$
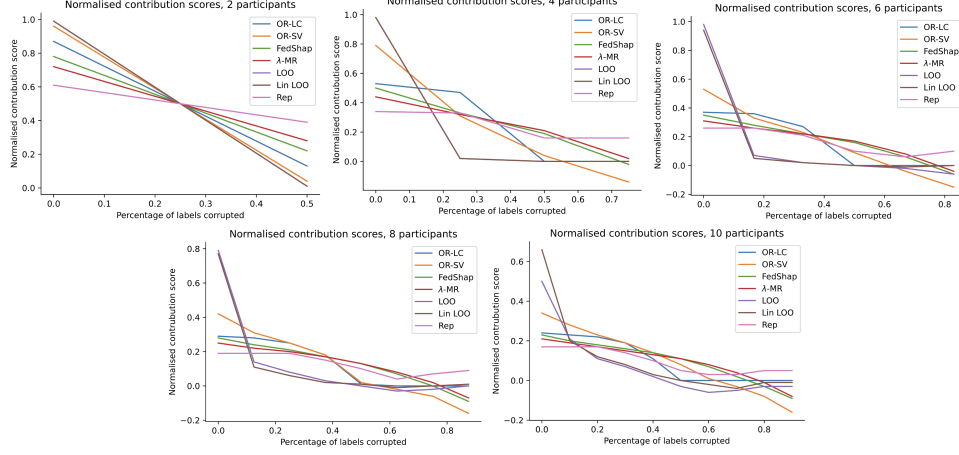
4

Figure 1: Comparison of normalized payoffs on MNIST

The quantities $v(S) - \sum_{i \in S} x_i$ are the deficits of each coalition, expressing its dissatisfaction with the payoff scheme. The least core is obviously non-empty if we let ϵ be unbounded, while the core itself is non-empty if $\epsilon^* \leq 0$. Similarly, the *nucleolus* is the unique imputation which lies in the least core and minimizes the largest deficit. Solving for the least core requires training $2^N - 2$ models to calculate the RHS of each constraint, and then solving a linear program with $2^N - 1$ constraints and $N + 1$ decision variables. In [16] they prove that, if we allow the Core guarantee to be violated by an amount e, δ percent of the time, i.e., $\Pr_{S \sim D} \left[\sum_{i \in S} x_i + \epsilon^* + e \geq v(S) \right] \geq 1 - \delta$, then an approximate least core solution, with probability $1 - \Delta$, requires sampling only $O\left(\frac{\tau^2 (\log n + \log(1/\Delta))}{e^2 \delta^2} \right)$ coalitions, where τ is the difference between the maximum and minimum value of the characteristic function. Unfortunately, they also prove that the same relaxations for the Nucleolus lead to a sampling procedure still exponential in the number of participants.

Other approaches utilize smart contracts on a blockchain to tackle the CE problem in the decentralized FL setting. Blockchain is especially well-suited to the task due to allowing verifiable/transparent transactions between peers, aiding trust. In [17], two decentralized protocols, for the cross-device and the cross-silo setting respectively, are introduced, to calculate the percentages of final ownership of the model each client should be allotted, based on the sum of the marginal contributions of the data points in each partner's dataset. However, they compute the contribution scores off-chain, in an unverifiable manner. All the centralized CE methods we examined so far implicitly assume the central party is trusted, and thus no type of secure aggregation is necessary. But in the decentralized FL scenario, that is no longer the case, and trust and security become even more important. Secure aggregation [9], a masking technique in FL where client updates are encrypted, so that their sum (which is the only quantity the server normally needs to update the global model) stays the same, but individual updates cannot be examined. With secure aggregation in place, we cannot calculate the SV or any such individual metric. As a compromise, in [18], the authors propose grouping participants and using secure aggregation only inside the groups, before computing, on-chain, the SV for the whole group and subsequently splitting that equally among the group members. An altogether use of Blockchain is proposed in [19], where, while the FL process itself is orchestrated by a central server, the server publishes, as tasks on a consensus blockchain, the calculations needed for CE (chosen to be the exact SV). Miners in the blockchain get paid by either calculating these to create blocks, or by verifying blocks, similar to how the bitcoin network works. While this side-steps approximations, miners are paid from the profits of the federation, meaning less revenue is left for the participants.

3.3 Models as rewards

Another crucial aspect of CE is combating *free-riders*: in the context of CGT, a free-rider is a client trying to extract a non-zero payoff, while contributing null, or redundant, information to the coalition. Null updates are tackled by any solution concept satisfying the Null Player fairness axiom, such as the SV, but more sophisticated attacks are not. Free-riding attacks are an understudied problem

in FL [20, 21], but rewarding attackers with a null payoff is not enough when the global model, encapsulating the entire value of the federation, is transmitted to all clients at every round. Especially in cross-enterprise FL, the global model is arguably much more valuable than any single monetary payoff; a free-rider being able to access the global model is an intolerable flaw.

While free-riding remains an open problem in the global-model paradigm, some recent work moderates the quality of the model that is transmitted to each client, leading to a model-as-reward paradigm; the more a client contributes, the better the model they receive is. Having such a deterrent to free-riders can complement free-rider detection techniques such as STD-DAGMM [20], and, as a side-benefit, this solves other conceptual problems in designing an FL Incentive Mechanism, such as enticing non-profit organizations (e.g., public hospitals, university labs) to take part.

Fairness in this paradigm can be achieved by ensuring the quality of each client's model is in fair proportion to their contribution quality. Such a mechanism, only applicable to Bayesian models, was designed in [22], where the distributed models have been injected with label noise proportional to the clients' contributions to induce different model quality. In [23, 24], instead of a global model, they propose a gradient-based economy, awarding clients with the gradients of other participants based on a credibility score. Every client converges to a different local model, with the clients uploading the most useful data getting the most gradients and thus ending with the most performant model. In [23], the server keeps copies of the local models, periodically evaluates them on $\mathcal{V}$, and assigns a corresponding credibility score to each client. This so-called reputation of each client is used to assign a fraction of the total gradients to them, with the most reputable one being allowed to download all the gradients. If a client's credibility falls under a threshold, he is ostracized from the set of reputable participants and does not get any updates, thus combating free-riders. In [24], they extend this to the decentralized setting, where no public validation set exists, requiring a much more involved design: clients initially publish a differentially private set of local samples, generated by a differentially private GAN, and every other client uses their local model to predict labels for these samples and sends them back to the samples' owner. After comparing these against the owner's predictions, a *local* credibility score is assigned to each client. These are aggregated, and used to decide the set of reputable participants, who train by sending DP gradients in return for tokens. Tokens can be spent to download the gradients of other clients. A participant with good data remains in the reputable set for longer, earning more tokens and thus having access to more gradients. All these transactions are recorded on a blockchain to make the peer-to-peer system verifiable and robust. Arguing against the use of validation performance-based evaluations, [25] instead proposed to form client tiers based on publicly verifiable factors, such as the quantity of their data and the cost they incurred to collect it, and giving each tier access to a model trained on a corresponding amount of data.

4   Benchmarking on MNIST and CIFAR-10

Here, we aim to –partially– fill an empirical gap in the related literature by comparing state-of-the-art CE methods for FL for a varying number of clients. The detailed experimental setup and additional experiments are relegated to the appendix. We choose OR-SV, λ-MR [14], FedShapley [13], round-level LOO, both weighted and unweighted [26] and Reputation [27] as the most promising CE methods applicable to FL. Besides these, we noticed that the pseudo-model approximation from [14] for OR-Shapley can also be used to efficiently find the Least Core. We name this CE method, OR-LC.

We examine their scalability, and their ability to differentiate contribution quality, by varying the number of clients from 2 to 10, and injecting their datasets with an increasing amount of label noise. Note that other artificial differentiations are possible, such as different dataset sizes/ percentages of data class ownership, but we chose the most intuitive/straightforward one. To provide a non-formal evaluation, we compare them by plotting the normalized contribution scores against the noise rate of each participant: intuitively a good CE method must be able to differentiate different data qualities and exhibit monotonically decreasing behaviour, and the slope's steepness reflects how aggressively each method reduces payoffs for sub-optimal contributions.

Firstly, we experiment on MNIST [28]. As seen in fig. 1, all the methods can differentiate the clients' data quality, but they greatly differ as to how austere they are. LOO and linearly weighted LOO heavily favour the best contributor and almost ignore the rest. The payoff vector dictated by Reputation is closer to an egalitarian uniform split, which is expected as it is much coarser than the other methods. The profiles of λ-MR and FedShapley stand between those, smoothly transitioning

from high to low values. OR-SV and OR-LC value highly good and mediocre contributors, but they sharply drop when contribution quality deteriorates past a threshold.
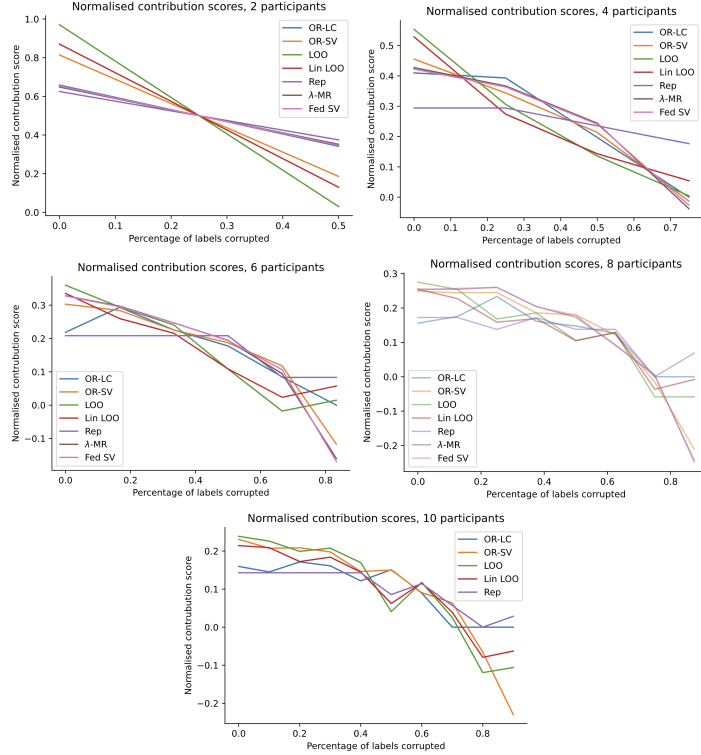

Figure 2: Comparison of normalized payoffs on CIFAR-10. MR for 10 participants requires the server to store 1024 ResNets, causing a crash.

Finally, we perform the same experiments on the much harder CIFAR-10[29] dataset. Under this more challenging setting, the methods continue to differentiate between data quality, but differences between them become harder to discern as can be seen in fig. 2. Depending on the application, the best suited CE method will differ. To give an example, following the law of supply and demand, the more interested participants there are, the more aggressively the federation can punish sub-optimal contributions.

5 Conclusion

In this work, we examined the existing set of approaches to evaluate participants' contributions in FL training in a fair way. Most existing work has focused on adapting the renowned concept of the Shapley Value to work with the idiosyncrasies of Federated Learning. Even at the conceptual level, that presents difficulties, so alternate approaches such as moderating the distributed model quality on a client level are being explored. Drawing from the Computational Game Theory area, we highlighted the related notions of Core and Nucleolus as currently understudied but promising alternatives to the Shapley Value. We then illustrated our careful literature study by comparing how the most promising methods deal with clients with different levels of corrupted labels in both the MNIST and CIFAR-10 classification tasks.

Future work in the area has to tackle the problem of rewarding early contributions more, in contrast to how the Shapley Value works, while also being interpretable as fair by the participants.

References

[1] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2017.

[2] Andrew Hard, Chloé M Kiddon, Daniel Ramage, Francoise Beaufays, Hubert Eichner, Kanishka Rao, Rajiv Mathews, and Sean Augenstein. Federated learning for mobile keyboard prediction, 2018. URL `https://arxiv.org/abs/1811.03604`.

[3] MELLODDY. MELLODDY: Machine Learning Ledger Orchestration For Drug Discovery, 2019. URL `https://www.melloddy.eu/`.

[4] The Substra Foundation. AI on clinical data : Healthchain consortium, 2019. URL `https://www.substra.ai/en/healthchain-project`.

[5] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurelien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrian Gascan, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konea, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrade Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Azgar, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramar, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and Open Problems in Federated Learning. *arXiv:1912.04977 [cs, stat]*, December 2019. URL `http://arxiv.org/abs/1912.04977`. arXiv: 1912.04977.

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

[7] T. Ichiishi and K. Shell. *Game Theory for Economic Analysis*. Economic theory, econometrics, and mathematical economics. Academic Press, 1983. ISBN 9780123701800.

[8] B. Peleg and P. Sudholter. *Introduction to the theory of cooperative games*. Theory and Decision Library C. Springer-Verlag Berlin Heidelberg, 2007. ISBN 9783540729440.

[9] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

[10] Amirata Ghorbani and James Zou. Data Shapley: Equitable Valuation of Data for Machine Learning. *arXiv:1904.02868 [cs, stat]*, June 2019. arXiv: 1904.02868.

[11] Amirata Ghorbani, Michael P. Kim, and James Zou. A Distributional Framework for Data Valuation. *arXiv:2002.12334 [cs, stat]*, February 2020. URL `http://arxiv.org/abs/2002.12334`. arXiv: 2002.12334.

[12] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gurel, Bo Li, Ce Zhang, Dawn Song, and Costas Spanos. Towards Efficient Data Valuation Based on the Shapley Value. *arXiv:1902.10275 [cs, stat]*, August 2019. arXiv: 1902.10275.

[13] Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. A Principled Approach to Data Valuation for Federated Learning. *arXiv:2009.06192 [cs, stat]*, September 2020. URL `http://arxiv.org/abs/2009.06192`. arXiv: 2009.06192.

[14] Shuyue Wei, Yongxin Tong, Zimu Zhou, and Tianshu Song. *Efficient and Fair Data Valuation for Horizontal Federated Learning*. 2020.

[15] Tianshu Song, Yongxin Tong, and Shuyue Wei. Profit allocation for federated learning. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2577–2586, 2019. doi: 10.1109/BigData47090.2019.9006327.

[16] Tom Yan and Ariel D Procaccia. If You Like Shapley Then You'll Love the Core. In *AAAI 2021*, 2021.

[17] Harry Cai, Daniel Rueckert, and Jonathan Passerat-Palmbach. 2cp: Decentralized protocols to transparently evaluate contributivity in blockchain federated learning environments, 2020.

[18] Shuaicheng Ma, Yang Cao, and Li Xiong. Transparent Contribution Evaluation for Secure Federated Learning on Blockchain. *arXiv:2101.10572 [cs]*, January 2021. arXiv: 2101.10572.

[19] Yuan Liu, Shuai Sun, Zhengpeng Ai, Shuangfeng Zhang, Zelei Liu, and Han Yu. Fedcoin: A peer-to-peer payment system for federated learning, 2020.

[20] Jierui Lin, Min Du, and Jian Liu. Free-riders in federated learning: Attacks and defenses. *arXiv preprint arXiv:1911.12560*, 2019.

[21] Yann Fraboni, Richard Vidal, and Marco Lorenzi. Free-rider attacks on model aggregation in federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 1846–1854. PMLR, 2021.

[22] Rachael Hwee Ling Sim, Yehong Zhang, Mun Choon Chan, and Bryan Kian Hsiang Low. Collaborative Machine Learning with Incentive-Aware Model Rewards. In *International Conference on Machine Learning*, pages 8927–8936. PMLR, November 2020.

[23] Lingjuan Lyu, Xinyi Xu, Qian Wang, and Han Yu. *Collaborative Fairness in Federated Learning*. 2020.

[24] Lingjuan Lyu, Yitong Li, Karthik Nandakumar, Jiangshan Yu, and Xingjun Ma. How to Democratise and Protect AI: Fair and Differentially Private Decentralised Deep Learning. *IEEE Trans. Dependable and Secure Comput.*, pages 1–1, 2020. ISSN 1545-5971, 1941-0018, 2160-9209. arXiv: 2007.09370.

[25] Jingfeng Zhang, Cheng Li, Antonio Robles-Kelly, and Mohan Kankanhalli. Hierarchically Fair Federated Learning. *arXiv:2004.10386 [cs, stat]*, May 2020. arXiv: 2004.10386.

[26] About substra. URL `https://www.substra.ai/en/about-substra-foundation`.

[27] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang. Incentive Mechanism for Reliable Federated Learning: A Joint Optimization Approach to Combining Reputation and Contract Theory. *IEEE Internet of Things Journal*, 6(6):10700–10714, December 2019. ISSN 2327-4662. doi: 10.1109/JIOT.2019.2940820. Conference Name: IEEE Internet of Things Journal.

[28] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

[29] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). 2012. URL `http://www.cs.toronto.edu/~kriz/cifar.html`.