Chapter 15

Data Valuation in Federated Learning

Zhaoxuan Wu[a], Xinyi Xu[a], Rachael Hwee Ling Sim[a], Yao Shu[a], Xiaoqiang Lin[a], Lucas Agussurja[a], Zhongxiang Dai[a], See-Kiong Ng[a], Chuan-Sheng Foo[b], Patrick Jaillet[c], Trong Nghia Hoang[d], and Bryan Kian Hsiang Low[a]

[a]National University of Singapore, Singapore, [b]Agency for Science, Technology and Research, Singapore, [c]Massachusetts Institute of Technology, MA, USA, [d]Washington State University, WA, USA

ABSTRACT

Federated Learning (FL) has become an increasingly popular solution paradigm for enabling collaborative machine learning (CML) in which multiple clients can collaboratively train a common model without sharing their private training data with others. However, broad adoption of FL in practice is still limited, as clients can often be reluctant to participate in such federated effort unless their contributions are accurately recognized and fairly compensated. Data valuation is thus extensively required to measure the relative contributions among clients. In this chapter, we review data valuation methods in the conventional supervised CML setting, followed by extensions to the FL paradigm. To better address the challenge that the private data from local clients cannot be made available to the server, we further discuss many specialized data valuation methods developed for both horizontal and vertical FL in detail. Overall, this chapter aims to provide a comprehensive suite of data valuation tools to empower FL practitioners in various practical scenarios.

KEYWORDS

Data Valuation, Incentives, Fairness, Privacy, Contribution Analysis, Feature Attribution, Gradient Attribution

1.1 INTRODUCTION

In recent years, there has been increasing interests in assessing the value of data in many real-world machine learning applications. Broadly speaking, in collaborative machine learning (CML), data valuation (DV) offers a trustworthy way of attributing rewards among participating clients and identifying potentially malicious ones in the learning effort. Federated learning (FL) is one of the most widely practiced CML frameworks, but its distinguishing data communication, fusion and learning characteristics from the canonical machine learning

framework necessitate tailored designs to effectively evaluate data contributed by various participating clients. For example, the server has no access to the raw data and the learning happens in an iterative round-wise manner. These characteristics pose challenges for developing effective and efficient methods.

More interestingly, different variants of FL based on the types of data partition follow vastly different learning pipelines and thus require distinct data valuation methods. FL can be typically categorized into horizontal and vertical variants. In horizontal FL (HFL), multiple clients contribute data samples that share a common feature space. For example, it can involve a large number of distributed mobile devices under a complex network. On the other hand, in vertical FL (VFL), clients contribute distinct features corresponding to the same data samples. VFL is commonly used among financial institutes and e-commerce platforms to learn models for a common set of customers.

This chapter provides an overview of data valuation methods for FL, starting with several representative data valuation methods in non-federated collaborative machine learning in Sections 1.3 and 1.4. We then discuss the possibilities for extending those established methods to the federated scenario in Section 1.5. Concrete data valuation approaches specially designed for VFL and HFL are discussed in detail in Section 1.6 and Section 1.7, respectively. Finally, we briefly introduce a vastly different approach from the other methods, learning-based valuation methods, in Section 1.8 and conclude the chapter with potential future directions in Section 1.9.

1.2 DATA VALUATION: MOTIVATIONS AND INCENTIVES

The essence of FL involves aggregating data resources from multiple distributed clients to collaboratively learn a better-performing machine learning model. For example, credit rating companies may collaborate with e-commerce platforms and mobile service providers for relevant data (e.g., shopping habits and phone bills) in improving their credit rating model. However, the clients may be self-interested and unwilling to participate in the federated effort unless their contributions are accurately recognized and fairly compensated. In such situations, data valuation can be useful in performing contribution evaluations and even guiding client selections, which are related to incentives in FL that will also be discussed later in this book. As such, data valuation methods are essential in facilitating collaboration among clients where data from a large group of participating clients are utilized together in a principled, efficient and fair manner.

Data valuation offers interpretability to machine learning models and decision-making. It attributes the model predictions to the most responsible training sample or feature. It also quantifies the importance of datasets (i.e., clients in FL) in achieving the final global model. More practically, the valuation can

be utilized to price each client's participation and thus determine whether it is worthwhile to involve a particular client.

Other applications of data valuation include data summarization, noise detection, domain adaptation and etc. When resource constraints poses a major challenge in practice, we can utilize data valuation to select the most valuable data samples that achieve the best model performance given the limited budget [20, 23]. Similarly, data valuation can guide more efficient model learning by training on the most valuable dataset first. Conversely, low values detected by valuation methods signal low-quality samples that could potentially improve model performance when removed [7, 21]. Finally, data valuation facilitates domain adaptation by valuing the training data in the context of the target validation data, which could have a significantly different distribution from the training data [4, 25].

1.3 SIMPLE VALUATION METHODS

"How valuable is a dataset?"

While the whole chapter aims to address this difficult question in the context of FL, we could start to examine this problem from a more intuitive viewpoint:

"Can we identify properties that characterize a valuable dataset?"

To answer this question, we first present several intuitive valuation concepts that do not depend on model or validation dataset.

Data Quantity. Quantity can be one of the most intuitive measures of data value. Roughly speaking, we expect a larger dataset to have a relatively higher value than a smaller one. Take the LibriSpeech [13] corpus data as an example, a data subset that contains 360 hours of speech is likely to have a higher value than a subset that only contains 100 hours of speech. More formally, we define the following utility to quantify the value of a dataset based on its size:

$$\nu(D_m) \quad = \quad |D_m| \tag{1.1}$$

where $|D_m|$ is the size of client m's data. However, the utility is often insufficient to quantify the contribution value of the data because it overlooks the presence of other clients' data in the collaboration. For example, adding the dataset from client m to an existing collaboration with limited data will create a significant impact, but adding it to a collaboration which has already acquired a significantly larger amount of data might only generate a marginal impact. Thus, to create a more informative quantification metric, we can pair the *utility* function with a simple *valuation function* to account for such relative effects (RE),

$$\phi_m^{\mathrm{RE}} \quad = \quad \frac{\nu(D_m)}{\sum_{i=1}^{n} \nu(D_i)} \tag{1.2}$$

where n is the total number of clients. One can imagine this valuation metric to be reliable in the scenario where data samples from all n clients are independently and identically distributed (i.i.d.). We can regard this metric to be implicitly *performance-driven* because it is commonly recognized in modern machine learning that a larger dataset typically leads to a model with better performance.

Data Variety. Quantity sometimes may not reveal the full picture. An extreme example is that one can replicate a single data sample for an infinite number of times to create an infinitely large dataset but the worth of the resulting dataset should not scale infinitely. More broadly, large datasets that lack data variety tend to have a lot of redundant information and might be less valuable than other datasets with a smaller size but higher variety. Therefore, data valuation also needs to account for the varieties in data (i.e., diversity), often in the forms of input and target coverage of the population. For instance, the range of values for an input feature (e.g., containing only ages $0-10$ instead of the whole demographic) and the number of target classes (e.g., containing only example images of digit 0 instead of the whole set of digits from the MNIST dataset) are all reflective of data variety. More concretely, let $\text{variety}(m)$ be the variety of the dataset owned by client m, then the relative value of D_m is expressed as

$$\nu(D_m) \quad = \quad \frac{\text{variety}(m)}{\sum_{i=1}^{n} \text{variety}(i)} \qquad \text{and} \qquad \phi_m^{\text{RE}} \quad = \quad \frac{\nu(D_m)}{\sum_{i=1}^{n} \nu(D_i)} \tag{1.3}$$

where the exact measure for data variety requires more in-depth investigation. For example, the relative data variety is often connected to the similarity measure of distributions. If we assume a target reference distribution is sufficiently diverse (i.e., possibly covers the entire population), a dataset closer in distribution to the reference distribution is more diverse and hence, more valuable. More details will be studied in the rest of this chapter.

Communication Effort. This property is related to data quantity and is unique to FL. The number of rounds of communications τ_m that a client m participated in could be a contribution indicator. More participation probably means more data quantity and also contribution in various phases of FL. The relative value of D_m can thus also be expressed as

$$\nu(D_m) \quad = \quad \frac{\tau_m}{\sum_{i=1}^{n} \tau_i} \qquad \text{and} \qquad \phi_m^{\text{RE}} \quad = \quad \frac{\nu(D_m)}{\sum_{i=1}^{n} \nu(D_i)} \,. \tag{1.4}$$

Overall, the three valuation metrics we introduced in this section can be simplistic at the first glance, but they can act as important guiding principles in more sophisticated data valuation method designs. We will frequently revisit these principles for the rest of this chapter.

TABLE 1.1 List of data valuation methods we discuss in this chapter. Methods in black are conventional data valuation methods applicable to FL. Methods in blue are methods developed in the FL context.

	VFL	HFL
Performance-driven	VP DAVINZ SHAP	ORC FedSV ComFedSV
Variety-driven	RV	
Similarity-driven	MMD DAVINZ	CGSV FedFAIM
Information-driven	IG CMI	
Learning-based		DVRL F-RCCE

1.4 RELATED WORK: CONVENTIONAL DATA VALUATION

Data valuation reflects how much each client contributes to the performance of the final global model. We first introduce data valuation methods in the canonical supervised learning setting without federated clients. In this setting, multiple clients contribute their dataset D_m to collectively learn a predictive model f. We define a coalition to be a subset of clients $C \subseteq \mathcal{A} \triangleq \{1, \ldots, n\}$. We denote the aggregated dataset from all n clients to be $D_{\mathcal{A}} = \{D_m\}_{m=1}^{n}$ where D_m is the local training dataset of client m. We overload the notation and let $D_C = \{D_m\}_{m \in C}$.

In this convention, data valuation requires an utility function $\nu : \mathcal{P}(D_{\mathcal{A}}) \to \mathbb{R}$ and a valuation function $\phi(D_m, D_{\mathcal{A}}, \nu)$, where $\mathcal{P}(D_{\mathcal{A}})$ denotes the power set of $D_{\mathcal{A}}$. Different designs on the utility and valuation function yield data valuation of different properties. The utility function ν aims to produce a data utility, which will be used by valuation functions ϕ to output the final value depending on the existence of other clients' data in FL. To offer an overview before delving into the details, we categorize the methods we will discuss in the rest of the chapter in Table 1.1.

1.4.1 Utility Functions

A utility function ν assigns a real-value utility to any coalition C formed among the participating clients. Intuitively, utility measures the "usefulness" of a coalition, which are used in valuation functions (see Section 1.4.2) to evaluate data. We present several representative utility functions next.

1.4.1.1 Performance-driven

Performance-driven utility functions award data coalitions that achieve high model performance.

Validation Performance (VP). VP is the most straightforward surrogate for model usefulness, adopted in Data Shapley [7]. Usually measured on a predefined validation set D_{val} of interest (or mutually agreed by the server and clients), we define

$$\nu(D_m) \quad = \quad -\ell(\mathbf{w}_m; D_{\text{val}}) \tag{1.5}$$

where $\mathbf{w}_m$ is the model trained on D_m and ℓ denotes the loss function. Sometimes, validation accuracy is used as an alternative for negated validation loss in Equation (1.5). This utility function is related to the performance-driven *data quantity* introduced in Section 1.3. As discussed, a larger dataset typically leads to a better-performing model, which means a lower validation loss and a higher value. However, each evaluation of the VP on a coalition requires computationally expensive model training, which could be prohibitively slow for deep neural network models. This limits the complexity of models that can be considered due to practical constraints.

Data Valuation at Initialization (DAVINZ). Motivated to value data in complex deep neural network (DNN) applications while completely avoiding model training, Wu et al. [21] theoretically derive a domain-aware generalization bound to estimate the generalization performance of DNNs without model training. Specifically, the utility function considers both in-domain DNN generalization error characterized by the neural tangent kernel (NTK) matrix $\mathbf{\Theta}_0$ at neural network initialization and the generalization error caused by train-validation domain divergence $d_{\mathcal{H}}(D_m, D_{\text{val}})$. We have

$$\nu(D_m) \quad = \quad -\kappa\sqrt{\hat{\mathbf{y}}\mathbf{\Theta}_0^{-1}\hat{\mathbf{y}}/|D_m|} \; - \; d_{\mathcal{H}}(D_m, D_{\text{val}}) \tag{1.6}$$

where each element in $\hat{\mathbf{y}}$ is defined as the residual on initialized network $\hat{y} = y - f(\mathbf{x}, \mathbf{w}^{(0)})$, $\mathcal{H}$ is a proper function space to evaluate domain divergence and κ is regarded as an balancing hyper-parameter. Intuitively, the first in-domain term can be interpreted as a complexity measure of D_m, whereas the second takes care of domain shifts. The authors adopt maximum mean discrepancy (MMD) for domain discrepancy in practice. Overall, DAVINZ addresses the computational efficiency problem of utility evaluation with an accurate estimate. The practical limitation lies in the determination of the hyper-parameter κ that balances the effort of in-domain and out-of-domain errors.

1.4.1.2 Variety-driven

Variety-driven utility functions examine the variety or diversity of data in coalitions as a surrogate for utility.

Robust Volume (RV). Xu et al. [23] propose a new perspective that attributes data value to intrinsic characteristics of a dataset itself regardless of tasks or models. To quantify the utility of a dataset in a machine learning task, the authors propose to use volume to measure the diversity of data samples in it and established theoretical connections between a high diversity dataset and a good learning performance. This formulation thus follows the *data variety* principle and theoretically connects to model performance considered in *performance-driven* methods. Let p_m be the number of data samples in D_m and d be the feature dimension. The volume of a data matrix $D_m \in \mathbb{R}^{p_m \times d}$ is defined as

$$\nu(D_m) \quad = \quad \sqrt{\det(D_m^\top D_m)} \tag{1.7}$$

where $\det(\cdot)$ denotes the determinant of a matrix. RV, a robust variant of volume in Equation (1.7), is also proposed in [23] to address the data replication issue in valuation. Overall, RV provides a viable alternative to VP-driven data valuation techniques, decoupling valuation from validation and circumventing the challenges associated with selecting a suitable validation set. The method is computationally efficient because it is training-free (i.e., no training is required). However, the disadvantage is that the theoretical performance guarantee applies to regression tasks only.

1.4.1.3 Similarity-driven

Different from methods driven by performance, similarity-driven methods inspect a dataset against a representative and trusted reference to determine values. Simply put, a dataset more similar to the aggregated data or a reference target distribution is assigned a high value.

MMD. Tay et al. [17] propose to use $D_{\mathcal{A}} \cup G_{\mathcal{A}}$ as the reference set where $G_{\mathcal{A}}$ represents a synthetic dataset generated from a generative model (e.g., variational autoencoder, generative adversarial networks, etc.) trained using $D_{\mathcal{A}}$. MMD is utilized to efficiently measure the distributional similarity between two sampled datasets. Specifically, we express

$$\nu(D_m) \quad = \quad -\mathrm{MMD}_u^2(D_m, D_{\mathcal{A}} \cup G_{\mathcal{A}}) \tag{1.8}$$

where MMD_u evaluates an unbiased MMD estimate. The work also discusses kernel selection for MMD and extension to incentivizing CML via synthetic data rewards, which will be discussed further in the following chapter on incentives for federated learning.

DAVINZ. Relatedly, the DAVINZ framework that we introduced earlier in Equation (1.6) has elements of a similarity-driven metric in the out-of-domain term. Accounting for domain shifts is especially important in CML and FL because clients normally have heterogeneous local data distributions which might not be identical to the target distribution at test time.

1.4.1.4 Information-driven

Information theory provides an alternative to quantifying the "usefulness" of a model through uncertainty associated with the model. This measure is independent of the validation dataset, thus circumventing the need of choosing an appropriate validation dataset for data valuation and associated biases.

Information Gain. Sim et al. [15] propose to measure the quality of a dataset D_m via the amount of uncertainty reduction in the trained model parameters $\mathbf{w}_m$ after training on D_m. Following the notions of information theory, the entropy of a random variable reflects the amount of "information" or "uncertainty" pertaining to the variable's outcome. Thus, using the prior entropy $H(\mathbf{w}_m)$ and posterior $H(\mathbf{w}_m|D_m)$ to represent the uncertainty associated with $\mathbf{w}_m$ before and after training, data value based on information gain (IG) can be expressed as

$$\nu(D_m) \quad = \quad H(\mathbf{w}_m) \; - \; H(\mathbf{w}_m \mid D_m) \; . \tag{1.9}$$

To interpret, a more valuable dataset results in a greater uncertainty reduction during model training. Interestingly, IG is related to performance-driven methods because it can be regarded as a predictive performance surrogate with unknown validation [9, 10]. However, the valuation method requires a Bayesian treatment of $\mathbf{w}_m$ (and D_m) and IG may be expensive to compute for some models such as multi-layer Bayesian neural networks.

1.4.2 Valuation Functions

Valuation functions operate on utilities calculated from the utility functions on multiple coalitions C with datasets $D_C \subseteq D_{\mathcal{A}}$. While we usually consider coalitions of clients in C, note that the value for each individual data sample is also well-defined under the same formulation. In this section, we discuss valuation functions with any arbitrary utility function ν.

1.4.2.1 Leave-one-out (LOO)

LOO contribution test finds its root in robust statistics, where Cook [3] uses it to study the influence of individual data points in linear regression. Intuitively, it computes the distance between the model fitted on the complete data and the model fitted on data with the m-th point or client deleted. In the context of data valuation, we employ

$$\phi_m^{\text{LOO}} \quad = \quad \nu(D_{\mathcal{A}}) \; - \; \nu(D_{\mathcal{A}\setminus\{m\}}) \tag{1.10}$$

where ν is an arbitrary utility function (some examples are given in Section 1.4.1). LOO only considers the marginal utility improvement of data to the grand coalition (excluding itself).

1.4.2.2 Shapley Value (SV)

Several game-theoretic solution concepts have proved their usefulness in data valuation, including Shapley value (SV) [7], Banzhaf value [19] and least core [24]. We focus on SV here as it is an unique solution that satisfies *efficiency, symmetry, linearity* and *null player* properties [5].

The SV of a client m is defined as the average marginal contribution of m to all coalitions $C \subseteq \mathcal{A} \setminus \{m\}$,

$$\phi_m = \frac{1}{|\mathcal{A}|} \sum_{C \subseteq \mathcal{A} \setminus \{m\}} \frac{1}{\binom{|\mathcal{A}|-1}{|C|}} \Big[\nu(D_{C \cup \{m\}}) - \nu(D_C) \Big] . \tag{1.11}$$

SV is more comprehensive than LOO as it considers marginal contributions to every possible coalition. SV is the most popular choice of valuation function to pair with the utility functions such as VP, RV, DAVINZ, MMD, IG, etc.

1.5 EXTENDING TO THE FEDERATED SETTING: DOES IT WORK?

Data valuation in the canonical supervised learning setting above requires (1) full access to the raw data and (2) a central server. This poses potential challenges when executing DV in FL.

HFL. We first recall the procedure of HFL. (1) The server broadcasts the latest global model $\mathbf{w}^{(t)}$ to all clients. (2) Clients perform local updates using their resepective local datasets. (3) Server selects clients and aggregates their gradients (or updated local models) to update the global model. The three steps are repeated until convergence. Unfortunately, extending conventional DV methods to HFL is nontrivial because raw data is never shared. Instead, the server only receives the gradient information or the locally updated model. The utility functions in Section 1.4.1 would not work with aggregated gradients. Therefore, we need to develop specialized DV methods for HFL (refer to Section 1.7).

VFL. We recall the general procedure of VFL. Overall, the model is divided into multiple client-owned bottom models and a server-owned top model as shown in Figure 1.1. The intermediate representations produced by the bottom models are subsequently passed as inputs to the top model for prediction or inference. We present the key steps below: (1) Before learning, a set of data samples with common identifiers are aligned across the participating clients. (2) Every client transmits the intermediate representation of the data using the respective bottom model. (3) Top and bottom models are updated through forward and backward propagation. Steps 2 and 3 are then repeated until convergence. A unique property of VFL is that each client separately owns a part of the grand model. Additionally, intermediate data representations based on the local models are shared with the server, which is usually the label owner in VFL. Therefore, DV

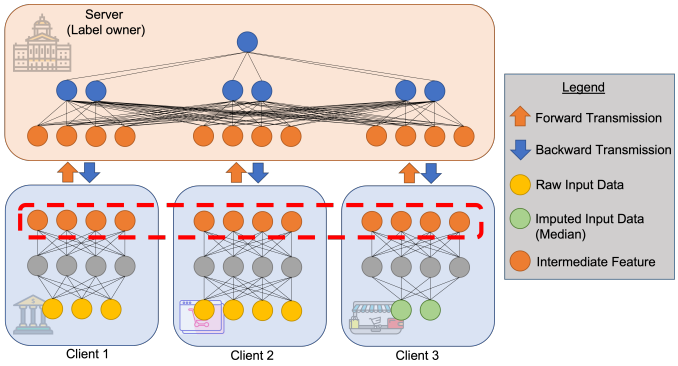

FIGURE 1.1 Intermediate representation valuation in VFL with 3 clients.

in VFL can be regarded as valuating the contribution of data in the form of intermediate representations. Specifically, we show an example with 3 clients in Figure 1.1. The dataset D_C with $C = \{1, 2\}$ is the concatenation of intermediate features from all 3 local models, where clients 1 and 2 output intermediate features with original local raw input data and client 3 outputs features with imputed median input data (or other reasonable imputation methods). To interpret, D_C only includes meaningful data from clients in coalition C. With this modified data representation, all utility functions and valuation functions introduced in Section 1.4 can be used in VFL.

Overall, depending on the information received by the server, data valuation in FL involves feature and gradient valuations. Conventional data valuation methods are still applicable to VFL with slight modifications on data representation. However, gradient evaluation in HFL requires a new approach. In the next section, we describe methods specially developed for data valuation in VFL and HFL, respectively.

1.6 VERTICAL DATA VALUATION: FEATURE VALUATION

As discussed in Section 1.5, data valuation in VFL can be viewed as grouped feature attribution since each client holds part of the federated model that contributes partial intermediate feature representations. We name data valuation in VFL as *vertical data valuation* and next discuss feature evaluation methods in the context of VFL.

1.6.1 Feature Importance and Attribution

Shapley additive explanations (SHAP) is a popular explainable artificial intelligence (XAI) tool that attributes the model prediction of an instance $\mathbf{x}$ to each

feature of $\mathbf{x}$ [12]. Interestingly, we can adapt SHAP into our data valuation formulation by using the final model output of an input instance as the utility function and SV as the valuation function. Note that SHAP, in its original form, considers single-instance multi-feature explanations. Therefore, Section 1.5 and Figure 1.1 provide us with a more suitable and flexible approach to vertical data valuation for sets of input data. Wang et al. [18] have proposed a similar method for vertical data valuation.

1.6.2 Information-driven Valuation

Han et al. [8] propose to use conditional mutual information (CMI), a commonly used metric for feature selection [1], as the data valuation metric. CMI is very similar to the information gain (IG) metric introduced in Section 1.4.1.4, except that CMI is conditioned on an additional task dataset D_{task} by the label owner in VFL. The task dataset is unique to the FL setting and the label owner holds corresponding task labels Y. Thus, [8] propose the DV metric based on the mutual information between the input set D_m and label Y conditioned on the task dataset D_{task}. Specifically,

$$\nu(D_m) \quad = \quad I(D_m; Y|D_{\text{task}}) \, . \tag{1.12}$$

The computation of CMI above requires further *federated computations* since D_m and D_{task} are typically stored separately and privately, which is outside the scope of DV discussion here. Remarkably, different from model-dependent methods like SHAP, such an information-driven metric can evaluate federated data in the absence of a pre-determined model.

1.7 HORIZONTAL DATA VALUATION: GRADIENT VALUATION

In HFL, gradients instead of raw data are shared with the central server. However, conventional data valuation techniques require full access to the raw data. Intuitively, if the contribution of each gradient to the global model can be evaluated, it will indirectly reflect the client's contribution.

1.7.1 Gradient Contributions

A straightforward data valuation method for HFL utilizes the gradient information readily available in the FL training procedure. Wei et al. [16] propose *one-round contribution* (ORC) to reconstruct trained models by aggregating local gradients throughout the training. The method effectively keeps track of $O(2^n)$ models (or equivalently, gradients information uploaded by clients) for each $C \subseteq \mathcal{A}$, which are later used to evaluate the VP of a global model trained with data from C. Specifically, for each coalition C, we use the aggregated gradients of clients from this coalition C to update the corresponding model. Note that although subset models are of interest, we still only compute the gradients

in each round using the global model. VP is used as the utility function v in data valuation.

However, gradients on the global model are used each round even when we are concerned with a subset model trained using only data from clients from the coalition C. Subset models can thus be different from the actual ones and affect the effectiveness of gradient values. To solve this issue, we can rely on the linearity property of Shapley value and regard each training round as a cooperative game. The overall SV is simply the sum of SV from all training rounds.

As such, *federated SV* (FedSV) [20] is proposed for valuing decentralized and sequential data in FL. The updated model performance conditioned on the existing global model is used as the utility function. Specifically, $v(C; \mathbf{w}^{(t)})$ is defined to be the VP of the updated global model additionally trained using the aggregated gradients of C from the existing global model $\mathbf{w}^{(t)}$. Following the Shapley formulation, the federated SV in a round t is defined as

$$\phi_m^{(t)} = \begin{cases} \frac{1}{|I_t|} \sum_{C \subseteq I_t \setminus \{m\}} \frac{1}{\binom{|I_t|-1}{|C|}} [v(C \cup \{m\}; \mathbf{w}^{(t)}) - v(C; \mathbf{w}^{(t)})] & \text{if } m \in I_t \\ 0 & \text{otherwise} \end{cases} \tag{1.13}$$

where I_t is the set of selected clients in round t. Then, the overall federated SV is the sum of all training rounds,

$$\phi_m = \sum_{t=1}^{\tau} \phi_m^{(t)} \,. \tag{1.14}$$

This requires τ rounds of SV computation to calculate the final FedSV. The above formulation can be generalized in two ways.

First, the weights of all coalitions are not necessarily equal. We can replace the coefficient $\frac{1}{|I_t|}$ in Equation (1.13) with a general factor α_C, that depends on specific formulations of C. For example, *Beta Shapley* [11] uses a beta distribution to weight coalitions based on their cardinalities such that the effect of noises in the utility evaluations can be reduced. Note that Beta Shapley reduces the SV to a semivalue without the *efficiency* axiom.

Second, we can vary the importance of different learning rounds. A significant drawback of ORC and FedSV is that they treat gradients from all training rounds equally. Mixing gradients from different training rounds together for gradient valuation may obfuscate essential gradients in the learning process. Therefore, Equation (1.14) can be generalized and normalized into

$$\phi_m = \sum_{t=1}^{\tau} \beta^{(t)} \left[\frac{\phi_m^{(t)}}{\sum_{i=1}^{n} \phi_i^{(t)}} \right] . \tag{1.15}$$

Here, we may introduce a decay factor $\lambda \in (0, 1)$ to account for the diminishing effect of the gradients, such that $\beta^{(t)} = \lambda^t$. We decrease the importance of the later training iterations as they usually take smaller gradient steps and influence the predictions to a smaller extent [16]. In addition, we may upweight rounds that lead to higher performance (i.e., accuracy). This stems from the observation that improvements over models with high accuracy are much harder than randomly initialized ones. To this end, we can set $\beta^{(t)} = \lambda^t \cdot \text{Perf}(\mathbf{w}^{(t)})$ where $\text{Perf}(\mathbf{w}^{(t)})$ denotes the validation performance of the global model at round t.

1.7.1.1 Improving on FedSV

An innate problem with Equation (1.13) is that the unselected clients in a specific training round receive *zero* utility, regardless of their datasets. This raises potential unfairness because, for example, two clients with the same data can receive different FedSV due to the sampling process. Fan et al. [6] empirically shows that randomly selecting 3 out of 10 clients for 10 rounds can cause larger than 50% relative FedSV difference 65% of the times.

Fan et al. [6] propose *completed federated SV* (ComFedSV) to improve fairness by imputing the missing entries of intermediate FedSV. ComFedSV collects round-wise Shapley value of all possible coalitions into a utility matrix $\mathbf{U}$ with τ rows. Then, it imputes the missing values via low-rank matrix completion [2]. Intuitively, this method is effective because $\mathbf{U}$ should be approximately low-rank. On the one hand, similar data shared across clients can lead to similar utilities and thus columns of $\mathbf{U}$. On the other hand, utilities of the same coalition should be similar between successive rounds. Moreover, ComFedSV also gives a theoretical guarantee for a fair data valuation and demonstrates convincing empirical performance.

1.7.2 Similarity-driven Gradient Valuation

When the subject of interest changes from data to gradients in HFL, similarity-driven utility functions are still versatile enough to be applicable. Since clients usually send gradients to the server which will be aggregated later, similarity metrics more tailored for gradient comparisons have been proposed.

Xu et al. [22] propose to capture the contribution of gradient uploaded by a client using gradient vector alignment. Intuitively, the closer the gradient is to the aggregated gradient from all clients, the more contribution it has made. The aggregated gradient is the direction in which the loss value of the global model decreases fastest. From this perspective, a (directionally) similar gradient would be more effective in reducing loss. Specifically, let the parameter update from client m in iteration t be $\Delta\mathbf{w}_{m,t} \triangleq -\eta_t \nabla g_m(\mathbf{w}_m^{(t)})$ where η_t is the learning rate for iteration t and $g_m(\mathbf{w}_m^{(t)})$ is the m-th client's local training loss with

respect to $\mathbf{w}_m^{(t)}$. The server normalizes and aggregates the gradients as follows, $\mathbf{u}_{m,t} \triangleq \Gamma \Delta \mathbf{w}_{m,t} / ||\Delta \mathbf{w}_{m,t}||$, $\mathbf{u}_{C,t} \triangleq \sum_{m \in C} r_m \mathbf{u}_{m,t}$ where Γ is a normalization coefficient used to prevent gradient explosion and r_m is an optional importance weight factor. Under this formulation, gradient alignment can be measured via the *cosine similarity* and the utility function is defined as follows,

$$\nu(D_m) \quad = \quad \cos(\mathbf{u}_{m,t}, \mathbf{u}_{\mathcal{A},t}) \quad = \quad \frac{\langle \mathbf{u}_{m,t}, \mathbf{u}_{\mathcal{A},t} \rangle}{||\mathbf{u}_{m,t}|| \cdot ||\mathbf{u}_{\mathcal{A},t}||} \,. \tag{1.16}$$

Note that the above utility function can be applied to data D_C from a coalition of clients C. We use (1.16) with (1.13) and (1.14) to obtain the respective Shapley values for data. This method is named cosine gradient Shapley value (CGSV). It enable us to perform data valuation on contributed gradients without any auxiliary dataset. Notably, Shi et al. [14] share the same perspective and propose a similar formulation for gradient contribution assessment named FedFAIM. On top of the cosine similarity in (1.16), they additionally consider quality detection and filtering of low-quality local gradients.

1.8 LEARNING-BASED VALUATION

In this section, we deviate from game-theoretic solution concepts, which can be inefficient when the number of clients is large despite existing approximation efforts. *Can we directly model the data value or the scoring function using advances in deep learning and reinforcement learning?*

Yoon et al. [25] first came up with the idea of data valuation using reinforcement learning (DVRL). It integrates data valuation with the training process of the target predictive model and utilizes reinforcement signals to train a network for data valuation. Most specifically, an evaluator $g_\psi : (\mathbf{x}, c) \rightarrow \omega$ maps training samples to a selection probability ω, which represents the probability of using the sample in a training iteration. The target predictor model is denoted as f_θ. In a training iteration, the evaluator first estimates the selection probabilities $\boldsymbol{\omega}$ for a batch of training samples $\{(\mathbf{x}^{(i)}, c^{(i)})\}_{i=1}^{B_s}$ of size B_s. Sample selection is then performed stochastically based on $\boldsymbol{\omega}$, and we obtain a selection vector $\mathbf{S} = [s_1, \ldots, s_{B_s}]$ where $s_i \in \{0, 1\}$. Here, 0 and 1 represent discarding or including the sample, respectively. Selected samples continue to train the predictor f_θ whose validation loss compared to the moving average δ of previous losses is then used as a reward signal to train the evaluator via reinforcement learning. After the convergence of the evaluator network, the selection probabilities serve as a surrogate of relative contribution (i.e., data value) in this learning effort.

The above idea has been applied to FL by Zhao et al. [26]. Instead of data samples, we consider contribution or value at the granularity of clients. Federated REINFORCE client contribution evaluation (F-RCCE) modifies the evaluator to take in gradients (or equivalently, local model at the end of the communication

round) as input, i.e., $g_\phi : \mathbf{w}_m^{(t)} \rightarrow \omega$. Similar to DVRL, $\mathbf{S}^{(t)} \in \{0,1\}^n$ and the reward function is defined as

$$r(\mathbf{S}^{(t)}) \quad = \quad \frac{1}{n_{\text{val}}} \sum_{k=1}^{n_{\text{val}}} \ell\Big(\mathbf{w}^{(t)}; \mathbf{x}_{\text{val},k}, c_{\text{val},k}\Big) \; - \; \delta \tag{1.17}$$

where ℓ is the loss of the global model on the validation set $\{(\mathbf{x}_{\text{val},k}, c_{\text{val},k})\}_{k=1}^{n_{\text{val}}}$ and δ is a moving average of the previous losses. The evaluator's model parameter ψ is updated with learning rate α:

$$\psi^{t+1} \quad \leftarrow \quad \psi^t \; - \; \alpha r(\mathbf{S}^{(t)}) \nabla_\psi \log p(\mathbf{S}^{(t)}|\psi)|_{\psi^t} \; . \tag{1.18}$$

Note that in this framework, the evaluator and the global target model are first fully trained before fixing the evaluator to measure contributions in another fresh round of target model re-training. In this case of FL, the selection probability at an iteration t is interpreted as the relative contribution of the client in communication round t. Overall, similar to (1.14), the value of a client m's data is the summation of its selection probability over all the rounds.

DVRL and F-RCCE are relatively efficient as it only requires one complete training of the valuation network (i.e., evaluator). However, the method now measures how likely a datum or a client will be used in training the predictive model, which cannot draw a direct parallel to the relative contributions in learning. Consequently, the above papers [25, 26] only perform experiments based on the ranks, rather than relative data values. Notably, the desirable properties and axioms of an equitable and fair data valuation achieved by Shapley value are not guaranteed by DVRL or F-RCCE.

1.9 CONCLUSION AND FUTURE WORK

Motivated by the growing interest in assessing the value of data in machine learning applications, this chapter presents an overview of data valuation methods in collaborative machine learning with a primary focus on federated learning. Data valuation offers an interpretable contribution attribution method for datasets in collaborative learning scenarios. However, it is important to notice the limitations of the existing methods and open problems for future research.

First, the current vertical data valuation methods are not tailored to the iterative learning process of FL. It is debatable whether we should perform feature valuations on the final model or consider round-wise valuations like those in horizontal data valuation. Second, learning-based valuation has only been applied to HFL. It would be interesting to investigate the applicability of learning-based valuations on VFL. Third, properties like communication bandwidth, computational power, honesty and availability of clients are additional aspects of client contribution to FL, which can potentially constitute a more complete client valuation framework. The list of open problems is certainly not comprehensive and

we invite interested readers to conduct further research on this growing field of practical significance.

To conclude, we have discussed extensions of the conventional data valuation methods to the federated setting and described horizontal and vertical data valuation methods specially developed for FL. This chapter serves as a guideline for a versatile suite of tools that empower FL practitioners to apply data valuation in various scenarios.

1.10 ACKNOWLEDGEMENT

This research/project is supported by the National Research Foundation Singapore and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-RP-2020-018).

BIBLIOGRAPHY

[1] Gavin Brown, Adam Pocock, Ming-Jie Zhao, and Mikel Luján. Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *JMLR*, 13(1):27–66, 2012.

[2] Wei-Sheng Chin, Bo-Wen Yuan, Meng-Yuan Yang, Yong Zhuang, Yu-Chin Juan, and Chih-Jen Lin. LIBMF: A library for parallel matrix factorization in shared-memory systems. *JMLR*, 17(86):1–5, 2016.

[3] R. Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, 19(1):15–18, 1977.

[4] Soumi Das, Manasvi Sagarkar, Suparna Bhattacharya, and Sourangshu Bhattacharya. CheckSel: Efficient and accurate data-valuation through online checkpoint selection, 2022.

[5] P. Dubey. On the uniqueness of the Shapley value. 4(3):131–139, 1975.

[6] Zhenan Fan, Huang Fang, Zirui Zhou, Jian Pei, Michael P Friedlander, Changxin Liu, and Yong Zhang. Improving fairness for data valuation in federated learning. 2022.

[7] Amirata Ghorbani and James Zou. Data Shapley: Equitable valuation of data for machine learning. In *Proc. ICML*, pages 2242–2251, 2019.

[8] Xiao Han, Leye Wang, and Junjie Wu. Data valuation for vertical federated learning: An information-theoretic approach, 2021.

[9] Andreas Krause and Carlos Guestrin. Nonmyopic active learning of gaussian processes: An exploration-exploitation approach. In *Proc. ICML*, page 449–456, 2007.

[10] John K. Kruschke. Bayesian approaches to associative learning: From passive to active learning. *Learning & Behavior*, 36(3):210–226, 2008.

[11] Yongchan Kwon and James Zou. Beta Shapley: a unified and noise-reduced data valuation framework for machine learning. In *Proc. AISTATS*, 2022.

[12] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proc. NeurIPS*, page 4768–4777, 2017.

[13] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *Proc. ICASSP*, pages 5206–5210, 2015.

[14] 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*, pages 1–13, 2022.

[15] Rachael Hwee Ling Sim, Yehong Zhang, Mun Choon Chan, and Bryan Kian Hsiang Low. Collaborative machine learning with incentive-aware model rewards. In *Proc. ICML*, pages 8927–8936, 2020.

[16] Tianshu Song, Yongxin Tong, and Shuyue Wei. Profit allocation for federated learning. In *Proc. IEEE Big Data*, pages 2577–2586, 2019.

[17] Sebastian Shenghong Tay, Xinyi Xu, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Incentivizing collaboration in machine learning via synthetic data rewards. In *Proc. AAAI*, 2022.

[18] Guan Wang, Charlie Xiaoqian Dang, and Ziye Zhou. Measure contribution of participants in federated learning. In *Proc. IEEE Big Data*, pages 2597–2604, 2019.

[19] Jiacheng Wang and Ruoxi Jia. Data Banzhaf: A data valuation framework with maximal robustness to learning stochasticity. In *Proc. AISTATS*, 2023.

[20] Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. A principled approach to data valuation for federated learning. In *Federated Learning*, pages 153–167. Springer, 2020.

[21] Zhaoxuan Wu, Yao Shu, and Bryan Kian Hsiang Low. DAVINZ: Data valuation using deep neural networks at initialization. In *Proc. ICML*, 2022.

[22] Xinyi Xu, Lingjuan Lyu, Xingjun Ma, Chenglin Miao, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Gradient driven rewards to guarantee fairness in collaborative machine learning. In *Proc. NeurIPS*, pages 16104–16117, 2021.

[23] Xinyi Xu, Zhaoxuan Wu, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Validation free and replication robust volume-based data valuation. In *Proc. NeurIPS*, pages 10837–10848, 2021.

[24] Tom Yan and Ariel D. Procaccia. If you like shapley then you'll love the core. In *Proc. AAAI*, pages 5751–5759, 2021.

[25] Jinsung Yoon, Sercan O. Arik, and Tomas Pfister. Data valuation using reinforcement learning. In *Proc. ICML*, pages 10842–10851, 2020.

[26] Jie Zhao, Xinghua Zhu, Jianzong Wang, and Jing Xiao. Efficient client contribution evaluation for horizontal federated learning. In *Proc. ICASSP*, pages 3060–3064, 2021.