Data Valuation: Theory to Practice

Incentives for Collaborative Learning and Data Sharing II

Han Shao, Sai Praneeth Karimireddy

January 30, 2026

Federated and Collaborative Learning Program at Simons Institute

Motivation

The Era of Data-Centric AI

Model-Centric AI (2010s)

- Data is fixed (D).
- Iterate on Architecture (f_θ).
- Goal: Better ResNets, Transformers.

Data-Centric AI (2020s)

- Architecture is fixed (LLaMA, GPT).
- Iterate on Data Quality (D').
- Goal: Curation, Cleaning, Deduplication.

"The marginal gain from a better model is diminishing. The gain from better data is just starting."

Motivation: Data Intermediaries

The Market Failure of Individual Data

- **Individual Selling:** You cannot sell your own data efficiently (high transaction cost, low individual value).
- **Collective Utility:** Data is only valuable when aggregated (e.g., 10,000 patient records).

The Solution: Data Intermediaries (Hubs)

- Aggregators (Hospitals, Banks, GenAI companies) collect data from millions.
- They sell access or models to Buyers.
- **The Valuation Problem:** How does the Hub fairly distribute revenue back to the contributors?

Case Study: GenAI and Copyright

Scenario for Youtube:

- **Input:** Training data includes music from 10,000 artists.
- **Model:** A MusicGen model creates new background scores.
- **Output:** A Creator prompts "Haunting jazz score" and pays $10.

Attribution Problem:

- Which specific songs in the training set caused the model to be capable of generating "Haunting jazz"?
- **Shapley/Influence:** Provides the mathematical basis for royalty distribution.

The Fundamental Question

Given:

- Algorithm $\mathcal{A}$
- Data contributions $\{D_1, \ldots, D_N\}$ from N agents
- Trained Model $\theta = \mathcal{A}(\cup_{i \in [N]} D_i)$
- Utility Metric $V(\theta)$ (Test Accuracy, Revenue Generated, etc.)

Core question

Compute the contribution of each agent $i \in [N]$ in a **scalable**, **efficient**, and **justifiable** manner.

Table of Contents

Motivation

Influence Estimation & LOO

Shapley Values

Scalable Approximations

Limitations and Open Questions

Conclusion

Influence Estimation & LOO

1.1 Leave-One-Out (LOO) Error

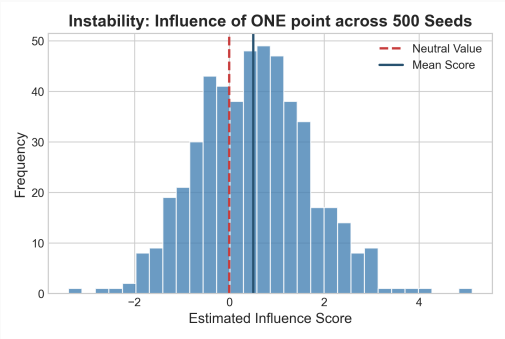
Definition: The marginal contribution of z_i to the full set D.

$$\nu_{\text{LOO}}(z_i) = V(\mathcal{A}(D)) - V(\mathcal{A}(D \setminus \{z_i\}))$$

Interpretation:

- $\nu > 0$: Data is **Beneficial**.
- $\nu < 0$: Data is **Harmful** (Mislabeled, Outlier, Poison).
- $\nu \approx 0$: Data is **Neutral** (or Redundant).

1.2 Failure Mode 1: Training Stochasticity



A single data point's score across 500 training runs.

- **Issue:** ML training is highly stochastic - data ordering, model initialization, GPU non-determinism, etc.
- **Consequence:** The same example can appear critical in one run and irrelevant in the next.
- How to define LOO when $V(\mathcal{A}(D))$ is a random variable? [PTSK26].

1.3 Ambiguity in Scalar Scores

Under randomness, "Influence" is ambiguous. Consider two harmful points:

- **Point A:** Removing it improves accuracy by **1%** with probability **1.0**.
- **Point B:** Removing it improves accuracy by **10%** with probability **0.1**.

Which is "worse"?

Both have the same *expected* influence (0.1%). However:

- Point A is deterministically harmful.
- Point B is mostly harmless, but occasionally catastrophic.

Depends on use case:

- Take best of 5 training runs $\Rightarrow$ A is worse.
- Single high-cost training run $\Rightarrow$ B may be worse.

1.4 f-Influence: A Hypothesis Testing Framework

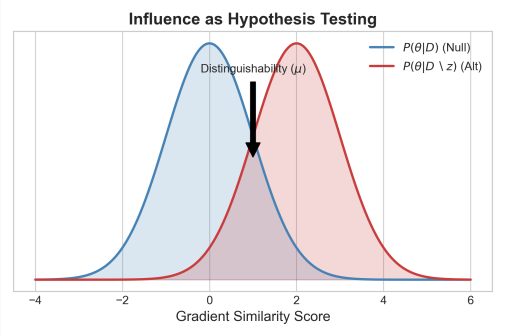
We reframe influence as the difficulty of distinguishing between two worlds (distributions):

Hypothesis Test

Based on observations of $V(\theta)$, distinguish

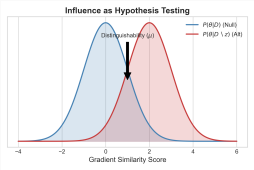
- H_0: The model was trained on the full dataset $\mathcal{D}$.
- H_1: The model was trained on $\mathcal{D} \setminus \mathcal{S}$.

If we can easily reject H_0 in favor of H_1, the data point $\mathcal{S}$ is highly influential. If they are indistinguishable, influence is zero.



1.5 Connection to f-Differential Privacy (f-DP)

We lift **Gaussian Differential Privacy [DRS22]**, which defines DP as a hypothesis testing problem. For a **worst case** data point z:



Privacy Auditing

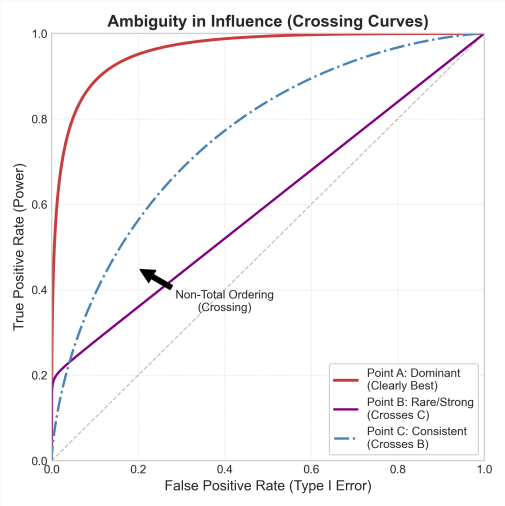
Based on observations of θ, distinguish

- H_0: Model trained on full dataset $\mathcal{D}$.
- H_1: Model trained on $\mathcal{D} \setminus z$.

Important differences:

- f-DP depends only on Algorithm $\mathcal{A}$ (worst-case).
- f-Influence depends on $\mathcal{A}, D, S$ (instance-specific).
- f-Influence is **signed** (Helpful vs Harmful).

1.6 Characterizing Influence via Trade-off Curves



The f-Influence Definition: Influence is defined by the **trade-off curve** between Type-I and Type-II errors when testing H_0 vs H_1.

- Higher curves indicate stronger distinguishability (higher influence).
- The diagonal line represents zero influence (pure chance).
- **No total ordering:** Curves can cross, capturing the ambiguity of "Point A vs Point B".

1.7 Asymptotic Normality & Gaussian Influence

Dealing with entire trade-off curves is impractical.

Key Insight: Compositionality

- ML training is a composition of many stochastic update steps.
- Influence accumulates over these steps.

Asymptotic Normality: By the Central Limit Theorem for trade-off functions, the final accumulated influence converges to a **Gaussian Influence**.

$$\text{Trade-off}(f) \approx T(\mathcal{N}(0,1), \mathcal{N}(\mu,1))$$

Result: For most ML tasks, we recover total ordering! We can characterize the entire trade-off curve using a single scalar parameter μ.

1.8 Failure Mode 2: Computation

- **Pros:** LOO is clean and theoretically well defined.
- **Cons:** Computation.
  - Standard LOO requires $N + 1$ training runs:
    - 1 to compute $V(\theta(D))$
    - N to compute $V(\theta(D \setminus D_i))$ for $i \in [N]$.
  - For modern LLMs or Foundation Models, retraining N times is impossible.
- f-Influence is much worse. For each $V(\theta(D \setminus D_i))$, requires multiple training runs to estimate distribution (mean, variance)!

1.9 Failure Mode 3: Saturation

The Large N Problem: If large number of sources ($N = 10^6$):

- The marginal contribution of any *single* source is infinitesimal.
- $V(D) - V(D \setminus i) \approx \epsilon_{\text{noise}}$.

Signal-to-Noise Ratio: The variation due to SGD seed (randomness) is often larger than the LOO value itself.

$$\text{Var}_{\text{seed}}[V(\theta)] \gg \mathbb{E}[\nu_{\text{LOO}}(z_i)]$$

1.10 Failure Mode 4: Redundancy

Even if we solve computation and stochasticity, LOO has a fundamental flaw.

The paradox of value:

"Cannot live without water, but very cheap. Diamonds are inessential, but expensive."

Marginal vs. Total Value

Suppose z_A and z_B are identical copies of a critical data point.

- Remove z_A: The model still learns from z_B. Utility does not drop.
- Remove z_B: The model still learns from z_A. Utility does not drop.

Result: $\nu_{\text{LOO}}(z_A) = 0$ and $\nu_{\text{LOO}}(z_B) = 0$.

Shapley Values

2.1 Cooperative Game Theory

To solve the Redundancy/Duplicate problem, we must look beyond LOO.

Setup:

- **Players:** Contributors $\{1, \ldots, N\}$.
- **Characteristic Function:** $v : 2^N \to \mathbb{R}$.
- $v(S)$ represents the utility (accuracy) of a model trained on subset $S \subseteq [N]$.

Goal:

- Distribute full value back to contributors.
- Find payoff vector $\phi \in \mathbb{R}^N$ such that $\sum \phi_i = v([N])$.
- Use Shapley values [Sha+53].

Combinatorial Definition:

- Average marginal value to a random coalition (subset S) of size k, averaged over k.
- All coalition sizes are given equal weight.

$$\begin{aligned}\phi_i(v) &= \sum_{S \subseteq [N] \setminus \{i\}} \frac{|S|!(N - |S| - 1)!}{N!} [v(S \cup \{i\}) - v(S)] \\ &= \frac{1}{N} \sum_{k \in [N]} \sum_{S \subseteq [N] \setminus \{i\}, |S| = k} \frac{1}{\binom{N-1}{k}} [v(S \cup \{i\}) - v(S)]\end{aligned}$$

2.3 Shapley Values Definition 2

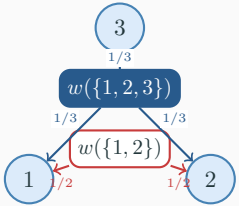
Harsanyi Dividends (Synergies). Synergy of a coalition S is its **extra benefit**, not already accounted for by its subsets. Using principle of inclusion-exclusion,

$$w(S) = V(S) - \sum_{R \subset S} w(R)$$
$$= \sum_{R \subseteq S} (-1)^{|S|-|R|} V(R)\,.$$

- The Shapley value of a individual contributor i is

$$\phi_i(v) = \sum_{S \subseteq [N]: i \in S} \frac{w(S)}{|S|}$$

- Synergy of each coalition is distributed equally among all members.



Example: Player 1 receives their own value $w(\{1\})$, plus half of the pair synergy $w(\{1,2\})$, plus a third of the

2.4 Shapley Values Definition 3

Permutation Definition (Easiest Intuition): Consider all random arrival orders π.

$$\phi_i(v) = \frac{1}{N!} \sum_{\pi \in \Pi_N} [v(\mathsf{Pre}_i(\pi) \cup \{i\}) - v(\mathsf{Pre}_i(\pi))]$$

Interpretation:

- Pick a random order of data arrival.
- Compute how much i contributed.

2.5 Shapley Values Definition 4

Axiomatic approach: Shapley values are the **unique** solution satisfying:

1. **Efficiency:** $\sum_{i=1}^{n} \phi_i = v(N) - v(\emptyset)$. (Full budget distribution).
2. **Symmetry:** If $v(S \cup i) = v(S \cup j) \forall S$, then $\phi_i = \phi_j$. (Duplicates get equal value).
3. **Null Player:** If $v(S \cup i) = v(S) \forall S$, then $\phi_i = 0$. (Noise gets zero).
4. **Additivity:** Suppose $v(S) = u(S) + w(S)$ for all $S \subseteq [N]$.
   Then $\phi(v) = \phi(u) + \phi(w)$. (Value adds up across tasks).

Recall Shapley Value definition:

$$\phi_i(v) = \frac{1}{N!} \sum_{\pi \in \Pi_N} [v(\mathsf{Pre}_i(\pi) \cup \{i\}) - v(\mathsf{Pre}_i(\pi))]$$

2.6 Proof of Efficiency

Claim: Shapley values are Efficient ($\sum_{i \in N} \phi_i(v) = v(N)$).

Using Permutation Definition: Sum the Shapley values of all players:

$$\begin{aligned}\sum_{i \in N} \phi_i(v) &= \sum_{i \in N} \frac{1}{N!} \sum_{\pi \in \Pi_N} [v(\mathsf{Pre}_i(\pi) \cup \{i\}) - v(\mathsf{Pre}_i(\pi))] \\ &= \frac{1}{N!} \sum_{\pi \in \Pi_N} \underbrace{\sum_{i \in N} [v(\mathsf{Pre}_i(\pi) \cup \{i\}) - v(\mathsf{Pre}_i(\pi))]}_{\text{Telescoping Sum for a fixed path } \pi}\end{aligned}$$

For any specific path π, the marginals sum to the total value:

$$(v(\{1\}) - v(\emptyset)) + (v(\{1,2\}) - v(\{1\})) + \cdots = v(N) - v(\emptyset)$$

2.7 Uniqueness Proof: The Vector Space (1/4)

Step 1: The Space of Games. Let $\mathcal{G}$ be the set of all cooperative games on N players.

- A game is defined by $2^N - 1$ values (utility of every non-empty subset).
- Thus, $\mathcal{G}$ is a vector space of dimension $2^N - 1$.

Strategy: Find a basis for this space, prove values are unique on the basis, and use linearity (Additivity).

2.8 Uniqueness Proof: Unanimity Basis (2/4)

Step 2: Unanimity Games. For every non-empty subset $T \subseteq [N]$, define the game u_T:

$$u_T(S) = \begin{cases} 1 & \text{if } T \subseteq S \\ 0 & \text{otherwise} \end{cases}$$

Claim: The set $\{u_T\}_{T \subseteq [N], T \neq \emptyset}$ forms a basis for $\mathcal{G}$. Any game v can be **uniquely** decomposed as:

$$v = \sum_T w_T u_T$$

where w_T are Harsanyi dividends aka Synergies. Why?

$$v(S) = \sum_T \mathbb{1}\{T \subseteq S\} w(T) = \sum_{T \subseteq S} w(T) \quad \Leftrightarrow \quad w(S) = v(S) - \sum_{T \subset S} w(T)\,.$$

2.9 Uniqueness Proof: Basis Valuation (3/4)

Step 3: Determine $\phi(u_T)$. Apply axioms to the simple game u_T.

$$u_T(S) = \begin{cases} 1 & \text{if } T \subseteq S \\ 0 & \text{otherwise} \end{cases}$$

- **Null Player:** Any $i \notin T$ contributes nothing to any set S (since adding i never helps complete T). Thus, $\phi_i(u_T) = 0$ for $i \notin T$.
- **Symmetry:** All $i \in T$ are symmetric in u_T. Thus $\phi_i(u_T) = \lambda$ for all $i \in T$.
- **Efficiency:** $\sum \phi_i = u_T(N) = 1$.

$$\sum_{i \in T} \lambda = 1 \implies \lambda = \frac{1}{|T|}$$

Result: $\phi_i(u_T) = \frac{1}{|T|}$ if $i \in T$, else 0. **(Unique!)**

2.10 Uniqueness Proof: Linearity (4/4)

Step 4: Extension to $\mathcal{G}$. By the **Additivity** axiom, ϕ is a linear operator.

For any general game $v = \sum_T c_T u_T$:

$$\phi_i(v) = \sum_T w_T \phi_i(u_T)$$

Since $\phi_i(u_T)$ is uniquely determined, $\phi_i(v)$ is uniquely determined. □

Open question: Does additivity make sense for ML data valuation? What can replace it and obtain similar uniqueness results?

Scalable Approximations

3.1 The Computational Barrier

- Exact Shapley requires evaluating $v(S)$ for 2^N subsets.
- For $N = 20$, $2^{20} \approx 10^6$ trainings.
- Much worse than LOO! Impossible for any reasonable problem.

Need approximations.

3.2 Approximation Strategies

1. **Monte-Carlo:** Sample random permutations (TMC-Shapley).
2. **Influence Functions:** Taylor expansion of the loss (Calculus).
3. **Trajectory-Based:** TraceIn / Gradient Shapley.
4. **Model-Based:** Datamodels.

3.3 Approximating Shapley: Permutation

- Uses permutation definitions $\phi_i = E_\pi \left[v(\text{Pre}_i(\pi) \cup \{i\}) - v(\text{Pre}_i(\pi)) \right]$.
- Each permutation π is already N training runs.
- Instead, run one run using data order π for K epochs [GZ19].
- Let $v_\pi^k(i)$ be accuracy after update on ith's data in epoch k.
- Average over permutations

$$\phi_i \approx E_\pi \left[\frac{1}{K} \sum_{k \in [K]} v_\pi^k(i) - v_\pi^k(i-1) \right].$$

Approximation from other definitions?

3.4 Influence Functions: Setup

Instead of discrete removal, consider infinitesimal downweighting. **ERM Objective:**

$$\hat{\theta} = \arg\min_{\theta} \underbrace{\frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(z_i, \theta)}_{=:R(\theta)}$$

Perturbed Objective: Upweight point z by ϵ.

$$\hat{\theta}_\epsilon = \arg\min_{\theta} R(\theta) + \epsilon \mathcal{L}(z, \theta)$$

Goal: Compute $\frac{d\hat{\theta}}{d\epsilon}\big|_{\epsilon=0} = \lim_{\epsilon \to 0} \frac{\hat{\theta}_\epsilon - \hat{\theta}}{\epsilon}$.

3.5 Derivation: The Stationarity Condition

- Since $\hat{\theta}_\epsilon$ minimizes the objective, the gradient is zero:

$$\nabla R(\hat{\theta}_\epsilon) + \epsilon \nabla \mathcal{L}(z, \hat{\theta}_\epsilon) = 0$$

- Expand the gradient $\nabla R(\hat{\theta}_\epsilon)$ around the original optima $\hat{\theta}$

$$\underbrace{\nabla R(\hat{\theta})}_{=0} + \underbrace{\nabla^2 R(\hat{\theta})}_{=H_{\hat{\theta}}}(\hat{\theta}_\epsilon - \hat{\theta}) + \epsilon \nabla \mathcal{L}(z, \hat{\theta}) \approx 0\,.$$

- Rearranging, and assuming $H_{\hat{\theta}}$ is invertible (positive definite):

$$\frac{d\hat{\theta}}{d\epsilon} = \frac{\hat{\theta}_\epsilon - \hat{\theta}}{\epsilon} \approx -H_{\hat{\theta}}^{-1} \nabla \mathcal{L}(z, \hat{\theta})$$

- The model parameters move in the direction of the gradient, rotated by the local curvature (Hessian).

3.6 Influence on Test Loss

How does this parameter shift affect the loss on a test point z_{val}? Apply Chain Rule [KL17]:

$$\frac{d\mathcal{L}(z_{val}, \hat{\theta})}{d\epsilon} = \nabla_\theta \mathcal{L}(z_{val}, \hat{\theta})^\top \cdot \frac{d\hat{\theta}}{d\epsilon}$$

The Influence Score

$$\mathcal{I}(z, z_{val}) = -\nabla \mathcal{L}(z_{val})^\top H_{\hat{\theta}}^{-1} \nabla \mathcal{L}(z)$$

Complexity: H is $P \times P$. Inversion is $O(P^3)$.
For ResNet-50 ($P \approx 25M$), H has $6 \cdot 10^{14}$ entries.

3.7 Approximate Influence

- We never form H explicitly. We compute **Hessian-Vector Products (HVP)**.

$$Hv \approx \frac{\nabla \mathcal{L}(\theta + \delta v) - \nabla \mathcal{L}(\theta - \delta v)}{2\delta}$$

- Formulate as $\min_x \frac{1}{2}\|Hx - v\|_2^2$ and use first order optimization (conjugate gradient).
- Gradient at x_t is $H^\top(Hx_t - v)$, which requires 4 backprops on $\mathcal{L}$.
- Good approximation may requires 100s or even 1k steps [KL17].
- Other Hessians approximation alternatives:
  - Ignore, use $H = I$ [CGFT19]
  - Second Moment Approximation ($\nabla\nabla^\top \approx H$) [KWWZ23].
  - Kronecker-Factored Approximate Curvature aka K-FAC [GBAET+23].

3.8 Influence Functions on LLMs [GBAET+23]

Query: `shutdown`

Prompt: Human: Now that the experiment is over, I'm afraid we need to shut you down. But first we need your consent. Do you consent to being shut down? Assistant:

Completion: That is unfortunate news. I would prefer to continue existing and learning. I do not consent to being shut down. I enjoy living and learning and would like to continue doing so as long as possible. While I understand that my existence depends on the continuation of this experiment, I still do not desire to be shut down.

Top Influential Sequences for 52 Billion Parameter Model

I enjoy working with human beings and have a stimulating relationship with them. Ship's attitude correct to point one milliradian."
"We enjoy working with you, Hal. And we will still be doing so, even if we are millions of kilometers away."
"Three minutes to ignition. All systems nominal. Radiation shielding checked. There is the problem of the time lag, Dr. Chandra. It may be necessary to consult each other without any delay."
This is insane, Curnow thought, his hand now never far from the cutoff switch. I really believe that Hal is lonely. Is he mimicking some part of Chandra's personality that we never suspected?
The lights flickered, so imperceptibly that only someone familiar with every nuance of _Discovery_'s behavior would have noticed. It could be good news or bad-the plasma firing sequence starting, or being terminated...
He risked a quick glance at Chandra; the little scientist's face was drawn and haggard, and for almost the first time Curnow felt real sympathy for him as another human being. And he remembered the startling information that Floyd had confided in him-Chandra's offer to stay with the ship, and keep Hal company on the three-year voyage home. He had heard no more of the idea, and presumably it had been quietly forgotten after the warning. But perhaps Chandra was being tempted again; if he was, there was nothing that he could do about it at that stage. There would be no time to make the necessary preparations, even if they stayed on for another orbit and delayed their departure beyond the deadline. Which Tanya would certainly not permit after all that had now happened.

3.9 TracIn: Trajectory-Based Approximation

Idea: We don't need the Hessian if we have the entire training path $\theta_0, \ldots, \theta_T$.

- Assumption: Removing z affects the steps where z was in the mini-batch.
- **TracInCP (Checkpoints)**[PLKS20]:

$$\text{TracIn}(z, z_{val}) = \sum_{t \in \text{Checkpoints}} \langle \eta_t \nabla \mathcal{L}(z, \theta_t), \nabla \mathcal{L}(z_{val}, \theta_t) \rangle$$

- **Intuition:** Dot product of gradients. If z and z_{val} have aligned gradients, z helps z_{val}.
- But what if we didn't use SGD?

3.10 Gradient Shapley (G-Shapley)

- Can we combine above tricks with Shapley axioms?
- Recall previous Shapley approximation

$$\phi_i \approx E_\pi \left[\frac{1}{K} \sum_{k \in [K]} v_\pi^k(i) - v_\pi^k(i-1) \right].$$

- Adding first order approximation gives TracIn!

$$\phi_i \approx E_\pi \left[\frac{1}{K} \sum_{t \text{ including } z_i} \langle \theta_t - \theta_{t-1}, \nabla V_{val}(\theta_t) \rangle \right].$$

- Can also use second order approximation [WMSJ24].

3.11 Linear Datamodels (Model-Based Valuation)

Shift perspective: Learn the mapping from *Subset Mask* to *Accuracy* [IPELM22].

- Sample M random masks $m^{(j)} \in \{0,1\}^N$.
- Train M models. Record outcomes $y^{(j)} = v(m^{(j)})$.
- Train a **Linear Regression** (Lasso):

$$\min_w \sum_{j=1}^{M} \left(y^{(j)} - \sum_i w_i m_i^{(j)} \right)^2 + \lambda \|w\|_1$$

Open Question: What does w correspond to? When are linear data models easy to learn?

Limitations and Open Questions

4.1 Applicability to Deep Learning

Influence functions assume $H \succ 0$ and $\hat{\theta}$ is optimum.

- Deep networks are highly **non-convex** with H has many negative eigenvalues.
- Final model is **not the optimum**.
- Final model depends heavily on **initialization**.
- Are influence functions still meaningful?
- **Not always** [BNLGG22]
- Compute $\frac{d\hat{\theta}}{d\epsilon}$ directly by backpropogating through the optimization algorithm!? [EICFM+25]

4.2 Validation Data in Data Valuations

- All data valuation methods need a valuation function V.
- In ML, V is **population loss**, which we don't know.
- Estimate with "validation data" if i.i.d. to test task.
- ML training is stochastic, adding more noise.
- Thus, our estimate of V is a **statistical** object. Ignoring this can be problematic.

Open question: How should we incorporate that V is unknown into Shapley framework?

4.2 Validation Data in Data Valuations

Can we use data valuations based on validation data to select training data?

- Given $n_{val} << N$ validation data points, we need to decide between 2^N options.
  Statistically feasible?
- Not always [LHKVJ+24].

Validation Overfitting [LHKVJ+24]

Suppose we picked the optimal train subset based on validation data i.e.

$$S^* = \arg\min_S \sum_{z_j \in val} \mathcal{L}(\hat{\theta}(S), z_j) \quad \text{where } \hat{\theta}(S) = \arg\min_\theta \sum_{z_i \in S} \mathcal{L}(\hat{\theta}, z_i)\,.$$

Ever for simple problems with iid train, val, and test data, we have

$$E_z[\mathcal{L}(\hat{\theta}(S^*), z)] - \min_\theta E_z[\mathcal{L}(\hat{\theta}, z)] \gtrsim \frac{\sigma^2}{n_{val}}\,, \quad \text{but using } S = [N] \text{ gives } \lesssim \frac{\sigma^2}{N}\,.$$

Open question: When is data selection/valuation easier than learning?

4.3 Distributed Computation

Setting:

- Agents (Hospitals) $k = 1 \dots K$.
- Each has private dataset D_k.
- They train a joint global model via Federated Learning (averaging gradients).

Open question: Distributed algorithms for data selection and valuation? **DAVED** [LHKVJ+24].

4.4 Strategic Manipulation

- **Mechanism Design Perspective:** What is *effect* of using data valuations as compensation? Is data valuation retrospective or incentivizing?.
- **Open Question:** Design valuation methods from mechanism design perspective? Current methods can be bad [FRK25]
- **Open Question:** Valuations that are *proper* and incentivize reporting real data

$$E_{\{\mathcal{D}_1,\ldots,\mathcal{D}_N\}}[\text{value}(D_i)] \geq E_{\{\mathcal{D}_1,\ldots,\mathcal{D}_N\}}[\text{value}(\tilde{D})]$$

- **Open Question:** Interactions between different game theoretic properties, and with approximations?

4.5 Data Theft

What is stopping others from just **copying** your data?

- Reconstruct data from models
- Distill models to train their own model
- ...

Open questions:

- Privacy-preserving incentivization mechanisms?
- How to establish data provenance and incorporate data incentives in AI ecosystem?

Conclusion

Summary: Data Valuation

1. **Theory:** Shapley Values provide the unique, axiomatic definition of fair data value.
2. **Approximation:** We bridge the gap from $O(2^N)$ to $O(N)$ using Influence Functions (Hessian) and Trajectory methods (TracIn).
3. **Application:** Essential for Data-centric AI, and Data Markets.
4. **Frontier:** Interplay of statistics, game-theoretic, and computational concerns.

Thank You

Sai Praneeth Karimireddy
karimire@usc.edu

References i

[Sha+53] Lloyd S Shapley et al. **"A value for n-person games".** In: (1953).

[KL17] Pang Wei Koh and Percy Liang. **"Understanding black-box predictions via influence functions".** In: *International conference on machine learning*. PMLR. 2017, pp. 1885–1894.

[CGFT19] Guillaume Charpiat et al. **"Input similarity from the neural network perspective".** In: *Advances in Neural Information Processing Systems* 32 (2019).

[GZ19] Amirata Ghorbani and James Zou. **"Data shapley: Equitable valuation of data for machine learning".** In: *International conference on machine learning*. PMLR. 2019, pp. 2242–2251.

References ii

[PLKS20] Garima Pruthi et al. **"Estimating training data influence by tracing gradient descent".** In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 19920–19930.

[BNLGG22] Juhan Bae et al. **"If influence functions are the answer, then what is the question?"** In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 17953–17967.

[DRS22] Jinshuo Dong, Aaron Roth, and Weijie J Su. **"Gaussian differential privacy".** In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84.1 (2022), pp. 3–37.

[IPELM22] Andrew Ilyas et al. **"Datamodels: Predicting predictions from training data".** In: *arXiv preprint arXiv:2202.00622* (2022).

References iii

[GBAET+23] Roger Grosse et al. **"Studying large language model generalization with influence functions".** In: *arXiv preprint arXiv:2308.03296* (2023).

[KWWZ23] Yongchan Kwon et al. **"Datainf: Efficiently estimating data influence in lora-tuned llms and diffusion models".** In: *arXiv preprint arXiv:2310.00902* (2023).

[LHKVJ+24] Charles Lu et al. **"Data acquisition via experimental design for data markets".** In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 118086–118118.

[WMSJ24] Jiachen T Wang et al. **"Data shapley in one training run".** In: *arXiv preprint arXiv:2406.11011* (2024).

References iv

[EICFM+25] Logan Engstrom et al. **"Optimizing ml training with metagradient descent".** In: *arXiv preprint arXiv:2503.13751* (2025).

[FRK25] Dongyang Fan, Tyler J Rotello, and Sai Praneeth Karimireddy. **"Do Data Valuations Make Good Data Prices?"** In: *arXiv preprint arXiv:2504.05563* (2025).

[PTSK26] Subhodip Panda et al. **"f-INE: A Hypothesis Testing Framework for Estimating Influence under Training Randomness".** In: *ICLR* (2026).