WHEN DO SCORE-BASED DATA VALUATION METHODS WORK, AND WHY?

Anonymous authors
Paper under double-blind review

ABSTRACT

Score-based valuation methods, such as Shapley-style scores and Leave-one-out (LOO), are widely used for credit assignment in data markets, yet theory offers limited guidance on when and why these methods succeed. In this paper, we study these methods using the best subset selection problem. We show that, even with monotone-submodular valuation functions, selection using LOO and Shapley-style scores cannot achieve a constant-factor approximation due to duplicate archetypes and collapsed pointwise credit. We also find that boundary effects in canonical learning problems can lead to supermodular spikes, preventing any valuation method—including adaptive methods like greedy selection—from achieving a constant-factor approximation. We identify two conditions that avert these failure modes: (i) bounded curvature, which controls redundancy and restores guarantees for LOO and Shapley-style scores, and (ii) coverage, which yields approximate submodularity on top of a sufficiently rich core. Our theoretical results and experiments motivate a practical algorithmic pipeline: deduplicate, ensure coverage, then apply score-based selection at an appropriate granularity.

1 INTRODUCTION

Modern machine learning systems increasingly treat data as an economic good: companies acquire, curate, and trade data; platforms compensate contributors; and practitioners audit training corpora to reduce bias, remove noise, or improve efficiency (Sestino et al., 2025; Zhang et al., 2023; Tian et al., 2022; Castro Fernandez, 2026). These workflows all require some notion of *data value*—a way to quantify how much a training example, a client, or a data silo improves downstream performance (Ghorbani & Zou, 2019; Koh & Liang, 2017; Debruyne et al., 2008; Han et al., 2020). As a result, data valuation now underlies a wide range of applications: credit assignment for pricing in data markets (Zhang et al., 2025; Raskar et al., 2019), dataset debugging and attribution, data pruning under storage or compute constraints, active acquisition and labeling, and client selection in collaborative learning (Hammoudeh & Lowd, 2024; Deng et al., 2025; Kairouz et al., 2019).

The central obstacle to data valuation is computational, as evaluating a subset by retraining (often repeatedly) and measuring downstream performance requires exponentially many runs. To avoid this, practical valuation pipelines often resort to a simpler template: assign each point a scalar score that can inform downstream decisions subject to budget constraints (Engstrom et al., 2024). Subset selection is the cleanest stress test of this template. If a score measures value, then selecting the top k points by score should approximate the optimal size-k subset that could be chosen with unlimited retraining. Approximations of Shapley-style scores (Shapley et al., 1953; Ghorbani & Zou, 2019) and leave-one-out (LOO) (Cook & Weisberg, 1982) are the most common primitives in this family, along with influence-based linearizations (Koh & Liang, 2017; Debruyne et al., 2008; Ilyas et al., 2022). These heuristics are popular because they provide clear rankings and credit, although theoretical guidance on when a learning problem can support reliable score-based evaluation is limited.

A natural starting point is to ask what guarantees score-based valuation can achieve under the most favorable regime for subset selection. A classical result shows that monotone submodularity yields a constant-factor approximation via greedy selection for the otherwise hard problem of maximizing a set function (Nemhauser & Wolsey, 1978). This makes monotone submodularity a tempting structural hypothesis for learning-induced valuations, under which pointwise scores might also succeed.

Our first main theorem shows that this intuition is *wrong*. Even with a monotone submodular valuation and exact access to every subset value, selecting the top k points by either LOO or Shapley scores can perform arbitrarily poorly relative to the best size-k subset (Theorem 3.1). The construction isolates a concrete mechanism—*duplicate archetypes*—where many near-substitutes collapse pointwise credit and prevent scalar scores from representing the benefit of diversifying across archetypes (Abbas et al., 2023). We then identify a condition that rules out this failure mode. A bounded curvature (Iyer et al., 2013; Vondrak, 1978) assumption which controls redundancy and restores constant-factor guarantees for both LOO and Shapley scores (Theorem 4.3).

These sufficient conditions, however, raise a basic question: do canonical learning problems produce valuation functions that satisfy submodularity (or curvature) at all? We show that the answer can be negative in strikingly simple settings, such as learning a one-dimensional threshold, where strong complementarity effects near the decision boundary violate diminishing returns. We leverage this effect to construct a learning problem where no valuation method can achieve a constant-factor approximation, including adaptive policies such as greedy selection (Theorem 5.5).

The same picture, fortunately, also suggests a way to avoid the pathology. When a training set already covers the relevant regions of the input distribution, adding a new point tends to refine an existing decision boundary rather than create one from scratch, which makes large complementarity spikes rare. We formalize this by showing that under i.i.d. sampling, the valuation induced by a linear classifier becomes approximately submodular above the sampled core (Proposition 5.7), and experiments show the same trend in real problems as core size grows (Figure 1). Our bounded-curvature guarantees also extend to this approximate-submodularity regime, with additive factors removed from the approximation (see Remark 5.2).

Taken together, our results motivate a concrete set of prescriptions for practice—control redundancy, build coverage, then apply score-based selection at an appropriate granularity. Our work raises many questions, but one is what other failure modes can lead to the collapse of non-adaptive methods. Towards this end, we provide one final negative result (Theorem 6.1), which, curiously, relies on *non-monotocity* and applies to **every non-adaptive method**, including influence-based linearization, showing a sharp gap relative to adaptive methods such as greedy selection.

2 Preliminaries and Problem Setup

Throughout the paper, we assume a learner who aims to minimize the following population risk,

$$\min_{x\in\mathcal{X}} \Big(\mathcal{R}(x) \triangleq \mathbb{E}_{z\sim\mathcal{D}}[f(x;z)]\Big) \quad , \tag{1}$$

where $\mathcal{Z}$ denotes the instance space, $\mathcal{D} \in \Delta(\mathcal{Z})$ denotes the target distribution, $\mathcal{X} \subseteq \mathbb{R}^d$ denotes the model class, and $f : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ denotes a loss function. The learner can access a finite dataset $S = \{z_1, \ldots, z_n\} \subset \mathcal{Z}$, indexed by $[n] \triangleq \{1, \ldots, n\}$. For a subset $T \subseteq [n]$, we write $S_T \triangleq \{z_i : i \in T\}$. We will use S_T and T interchangeably.

Let $\mathcal{A}$ denote a (possibly randomized) learning algorithm which given S_T, outputs a distribution over models in $\Delta(\mathcal{X})$. We expose algorithmic randomness in $\mathcal{A}$ by writing $\mathcal{A}_\xi : \mathcal{Z}^\star \to \mathcal{X}$, where ξ collects sources of randomness such as initialization, mini-batch sampling, dropout, etc. And we will study the following learning-induced set function, which we call the value of a training subset,

$$v(T) \triangleq -\mathbb{E}_\xi[\mathcal{R}(\mathcal{A}_\xi(S_T))] \quad . \tag{2}$$

Essentially a larger value $v(T)$ indicates better expected performance under the population $\mathcal{D}$. We will say that the valuation function is *normalized* if $v(\emptyset) = 0$. Throughout the paper, when we compare different valuation methods, we fix the learning algorithm $\mathcal{A}$, the random seed ξ, and the data distribution $\mathcal{D}$. We will, in fac,t abstract away all of these factors using the following oracle:

Definition 2.1 (Valuation oracle)**.** *Given a learning algorithm $\mathcal{A}$, its source or randomness ξ and population distribution $\mathcal{D}$, a* valuation oracle $\mathcal{O}^{\mathcal{A},\xi,\mathcal{D}} : 2^{[n]} \to \mathbb{R}$ *outputs,*

$$\mathcal{O}^{\mathcal{A},\xi,\mathcal{D}}(T) = -\frac{1}{m}\sum_{i\in[m]} f(\mathcal{A}_\xi(S_T); z_i') \quad , \qquad \forall\, T \subseteq [n]$$

where $z'_1, \ldots, z'_m \sim^{i.i.d.} \mathcal{D}$. *In particular for all* $T \subseteq [n]$,

$$\mathbb{E}_{z'_1,\ldots,z'_m,\xi}\left[\mathcal{O}^{\mathcal{A},\xi,\mathcal{D}}(T) \mid T\right] = v(T) \ .$$

We call $\mathcal{O}^{\mathcal{A},\xi,\mathcal{D}}$ *an* exact oracle *if* $\mathcal{O}^{\mathcal{A},\xi,\mathcal{D}}(T) = v(T)$.

Remark 2.2 (Randomness in valuation)**.** *The above oracle is unbiased, but it can be very noisy when* m *is moderate or when* $\mathcal{A}$ *is sensitive to* ξ. *Moreover, the noise need not be homogeneous across subsets: training on small versus large* S_T *can induce qualitatively different dynamics (e.g., overfitting vs. failure to train) (Ilyas et al., 2022). As a result, reliable estimation of* $v(T)$ *may require repeated retraining and evaluation. In this paper, we separate such concerns from the structural limits of score-based valuation, as our negative results hold even under an* exact *oracle, since most of our hard instances use deterministic learning rules.*

Data selection. As mentioned before a common downstream use of data valuation is *data selection*: one wants to keep only $k \leq n$ examples (for labeling, storage, training time, privacy, or computational reasons) while preserving performance (Engstrom et al., 2024). Accordingly, throughout this paper, we evaluate valuation methods by how well they support the cardinality-constrained subset selection problem. Specifically, we fix a subset size $k \leq n$ and define the benchmark,

$$\mathrm{OPT}_k \triangleq \max_{T\subseteq[n]:\,|T|\leq k} v(T) \ . \tag{3}$$

Our goal is to use the benchmark OPT_k to compare different data selection algorithms. Specifically, we aim to understand how the performance of a data selection algorithm depends on its oracle query budget Q, the dataset size n, and the subset size k. We will study these questions for value functions induced by different classes of learning problems and algorithms.

Score-based selection. The common template for all score-based methods is to assign a scalar score $s_i^{\Pi} \in \mathbb{R}$ to each point $i \in [n]$ and then select,

$$T_k \in \arg\max_{T\subseteq[n]:\,|T|\leq k} \sum_{i\in T} s_i^{\Pi} \ , \tag{4}$$

where ties are broken randomly. All score-based valuation rules fit this template and differ in how they compute their score. For brevity, we will discuss these methods in the exact-oracle model and denote the class of deterministic[1] score-based methods with query budget Q by $\mathsf{Score}(Q, k)$.

Leave-one-out (LOO). For $S \subseteq [n]$ and $i \notin S$, we define the marginal gain of adding i to S as

$$\Delta(i \mid S) \triangleq v(S \cup \{i\}) - v(S) \ . \tag{5}$$

The LOO rule assigns the following score to each $i \in [n]$,

$$s_i^{\mathrm{LOO}} \triangleq \Delta(i \mid [n] \setminus \{i\}) = v([n]) - v([n] \setminus \{i\}) \ , \tag{6}$$

which in the exact-oracle model corresponds to using $n + 1$ oracle calls: one for $v([n])$ and n for $\{v([n] \setminus \{i\})\}_{i\in[n]}$. We will denote the top-k-LOO set by T_k^{LOO}.

Shapley score. The Shapley score (Shapley et al., 1953; Ghorbani & Zou, 2019) assigns each point a score based on its average marginal contribution across all possible contexts. Concretely, sample a permutation π of $[n]$ uniformly at random and let $P_i^{\pi} \subseteq [n] \setminus \{i\}$ denote the set of points that appear before i in π. The Shapley score of i is the expected marginal gain of adding i to its random predecessor set:

$$s_i^{\mathrm{Sh}} \triangleq \mathbb{E}_{\pi}[\Delta(i \mid P_i^{\pi})] \ . \tag{7}$$

Equivalently, one can expand equation 7 as a weighted average for each $i \in [n]$ over all $U \subseteq [n]\setminus\{i\}$,

$$s_i^{\mathrm{Sh}} = \sum_{U\subseteq[n]\setminus\{i\}} \frac{|U|!\,(n-|U|-1)!}{n!} \cdot \Delta(i \mid U) \ , \tag{8}$$

[1]These methods are deterministic in the sense that they use a deterministic mapping between the Q oracle responses and the scores for each data point $i \in [n]$.

where the weight is the probability that U is the predecessor set of i under a uniform random permutation. Exact evaluation of equation 8 requires exponentially many oracle calls in general, so practical implementations approximate equation 8 by Monte Carlo sampling of permutations and oracle queries. We will denote the top-k-Shapley set by T_k^{Sh} (Wang et al., 2024; Chen et al., 2025).

Influence-function linearization. Influence-function methods (Koh & Liang, 2017; Debruyne et al., 2008; Ilyas et al., 2022) approximate the effect of removing or upweighting a point by linearizing the training map around a reference model (often the model trained on S). In our framework, these methods yield scores s_i^{IF} from a small number of oracle calls and analytic approximations. In this paper, we do not explicitly discuss these approaches, but some of our lower bounds apply to them.

Non-adaptive selection algorithms. A deterministic non-adaptive selection algorithm with query budget Q is a procedure Π that interacts with an exact value oracle as follows:

1. Π chooses Q query sets $S^{(1)}, \ldots, S^{(Q)} \subseteq S$ *before* seeing any oracle responses.
2. Π receives a transcript $\mathcal{T} := \big(v(S^{(1)}), \ldots, v(S^{(Q)})\big) \in \mathbb{R}^Q$.
3. Π outputs a set $U = U(\mathcal{T}) \subseteq S$ with $|U| \le k$, where $U(\cdot)$ is an arbitrary deterministic function.

We denote the class of such algorithms by $\mathsf{NA}(Q, k)$. Note that an algorithm $\Pi \in \mathsf{NA}(Q, k)$ has exactly one batch of parallel oracle access: it commits to the entire collection of queries $S^{(1)}, \ldots, S^{(Q)}$ up front, and only after receiving all answers does it decide what set U to output. The output stage $U(\mathcal{T})$ is intentionally unrestricted: after seeing the Q oracle values, the algorithm may compute *any* function of the transcript and return *any* set of size at most k. Clearly, $\mathsf{Score}(Q, k) \subseteq \mathsf{NA}(Q, k)$, but the converse is also true.

Proposition 2.3 (Describing non-adaptive algorithms using scores)**.** $\mathsf{Score}(Q, k) = \mathsf{NA}(Q, k)$.

We prove the proposition in Appendix A, and use this equivalence to show a separation between score-based and adaptive methods—the simplest of which is greedy selection.

Greedy selection. Greedy selection builds a set iteratively: let $T_0^{\mathrm{gr}} = \emptyset$, then at round $t \in \{1, \ldots, k\}$, greedy selects

$$i_t \in \arg\max_{i \in [n] \setminus T_{t-1}^{\mathrm{gr}}} \Delta(i \mid T_{t-1}^{\mathrm{gr}}) \ , \tag{9}$$

with random tie-breaking and sets $T_t^{\mathrm{gr}} = T_{t-1}^{\mathrm{gr}} \cup \{i_t\}$. Finally, the rule outputs the set T_k^{gr}. In the exact-oracle model, greedy selection uses $O(nk)$ oracle calls.

2.1 Submodularity as the Natural Hope

A natural assumption for subset selection is that the value function $v(\cdot)$ satisfies a "diminishing-returns property". Sub-modularity is one popular formalization of this property.

Definition 2.4 (Submodularity)**.** *A set function $v : 2^{[n]} \to \mathbb{R}$ is* submodular *if for all $A \subseteq B \subseteq [n]$ and all $i \in [n] \setminus B$,*

$$v(A \cup \{i\}) - v(A) \ \ge \ v(B \cup \{i\}) - v(B) \ . \tag{10}$$

It is also natural to assume that adding more data does not hurt the algorithm[2].

Definition 2.5 (Monotonicity)**.** *A set function $v : 2^{[n]} \to \mathbb{R}$ is* monotone *if for all $A \subseteq B \subseteq [n]$,*

$$v(A) \ \le \ v(B) \ . \tag{11}$$

When $v(\cdot)$ is monotone and submodular, it is folklore that greedy selection (defined in equation 9) is an optimal algorithm.

Theorem 2.6 (Greedy for monotone submodular valuations Nemhauser & Wolsey (1978))**.** *Assume $v : 2^{[n]} \to \mathbb{R}$ is monotone, submodular and satisfies $v(\emptyset) = 0$. Then greedy selection using an exact oracle outputs T_k^{gr} s.t.,*

$$\mathbb{E}\left[v(T_k^{\mathrm{gr}})\right] \ \ge \ \left(1 - \frac{1}{e}\right) \mathrm{OPT}_k \ . \tag{12}$$

[2]In practice, adding data points to a machine learning pipeline need not always be monotonic. In fact, in Theorem 6.1 we crucially use the anti-complementarity of points.

Remark 2.7 (Role of monotonicity)**.** *If monotonicity fails, the greedy rule in equation 9 can fail, and one typically switches to algorithms designed for the non-monotone setting. For example, in the unconstrained case, the double-greedy framework achieves a $1/2$-approximation (Buchbinder et al., 2015). Under a k-cardinality constraint, randomized greedy variants achieve constant-factor guarantees, such as $1/e$ for general nonnegative submodular objectives (Buchbinder et al., 2014), building on earlier work on non-monotone submodular maximization (Feige et al., 2011).*

Theorem 2.6 justifies why submodularity is a good starting assumption, motivating the next section.

3 Even Under Submodularity, Score-Based Valuation Can Fail

The key question we ask in this section is whether monotonicity and submodularity of the valuation function are sufficient conditions for top-k selection by Shapley or LOO to work. The main result of this section is the following theorem, which answers this question negatively!

Theorem 3.1 (Submodularity does not rescue score-based selection)**.** *Fix any $k \geq 2$, $\varepsilon \in (0,1)$ and $M_0 > 0$. There exists a learning instance whose induced value function $v : 2^{[n]} \to \mathbb{R}$ is monotone and submodular, and satisfies $v(\emptyset) = 0$, such that $\mathrm{OPT}_k \;\geq\; k(1-\varepsilon)M_0$ and:*

1. *(*Shapley-top-k fails.*) Top-k selection by Shapley scores outputs a set T_k^{Sh} with $v(T_k^{\mathrm{Sh}}) \;=\; M_0$.*
2. *(*LOO-top-k fails.*) Top-k selection by LOO scores outputs a random set T_k^{LOO} with $\mathbb{E}\big[v(T_k^{\mathrm{LOO}})\big] \;\leq\; \frac{3}{k}\, M_0$.*

In particular, both Shapley-top-k and LOO-top-k achieve at best an approximation ratio $O(1/k)$.

Proof sketch. We instantiate $v(\cdot)$ using a 1–nearest neighbor learner on $\mathbb{R}$, with test mass supported on R locations. To correctly classify a location, the algorithm must memorize at least one point in that location. We then place the n candidate training points in S at one of these locations. This makes the value function a weighted coverage function

$$v(T) \;=\; \sum_{r=0}^{R} M_r \cdot \mathbb{I}\{T \cap \mathcal{G}_r \neq \emptyset\} \;, \tag{13}$$

where $\{\mathcal{G}_r\}_{r=0}^{R}$ partitions $[n]$ into duplicate groups and $M_r > 0$ denotes the test mass covered by group r. Lemma B.1 shows that this function is monotone submodular.

Shapley. We create one small group $\mathcal{G}_0$ of size k and weight M_0, and k further groups $\mathcal{G}_1, \ldots, \mathcal{G}_k$ of size k with weights $M_r = (1-\varepsilon)M_0$. Within any group, duplicates act as perfect substitutes, so a point contributes only when it is the first representative of its group in a random permutation. This event has probability $1/k$, so every element in group r has Shapley score M_r/k. Hence every element of $\mathcal{G}_0$ ranks above every element outside $\mathcal{G}_0$, and Shapley-top-k selects $\mathcal{G}_0$, which yields value M_0. The optimal size-k set selects one representative from each of the k groups $\mathcal{G}_1, \ldots, \mathcal{G}_k$ and achieves value $M_0 + (k-1)(1-\varepsilon)M_0$.

LOO. A similar argument works for LOO, relying on the fact that removing one point never removes the last representative of any group. Section B gives the full proof. □

These counterexamples isolate a concrete pathology: *duplicate archetypes*, where many near-substitutes collapse pointwise credit. Greedy selection avoids this failure mode because, after selecting one representative from a group, adaptive selection ignores the remaining duplicates.

Smoothened score-based selection. The failure of the Shapley method in Theorem 3.1 arises because of the *hard* top-k selection based on Shapley scores, which amplifies small score differences between similarly valuable groups. This suggests a natural smoothing: sample points (or whole sets) with probability proportional to their Shapley scores, rather than selecting the top-k. The next theorem shows that this smoothing does not salvage Shapley scores: a broad family of proportional-to-Shapley (PPS) rules still fails to achieve any constant-factor approximation.

Theorem 3.2 (Sampling proportional to Shapley can be arbitrarily suboptimal)**.** *For every integer $k \geq 2$, there exists a learning instance whose induced value function $v : 2^{[n]} \to \mathbb{R}$ is monotone, submodular value function with $v(\emptyset) = 0$ such that each of the following PPS-style selection rules returns a random set T_k^{PPS} satisfying,*

$$\frac{\mathbb{E}[v(T_k^{\mathrm{PPS}})]}{\mathrm{OPT}_k} \;\leq\; \frac{3}{k} \;. \tag{14}$$

The bound holds for:

1. *(*PPS with replacement*) draw k times independently with $\Pr(i) \propto s_i^{\mathrm{Sh}}$ and let T_k^{PPS} be the set of distinct draws;*
2. *(*PPS without replacement*) draw k distinct items sequentially with $\Pr(i \mid \textit{past}) \propto s_i^{\mathrm{Sh}}$ using the* fixed *Shapley weights;*
3. *(*Adaptive PPS via residual Shapley*) at each step recompute Shapley values for the residual game and sample proportionally to these residual Shapley values;*
4. *(k-set PPS) sample a k-subset S with probability proportional to $\sum_{i \in S} s_i^{\mathrm{Sh}}$.*

Proof sketch. We use the same learning-like template as in the proof of Theorem 3.1: a 1–NN instance whose induced value function is a weighted coverage function over duplicate groups (Section C.1). In such instances, items inside the same duplicate group are symmetric, and the Shapley mass of an entire group equals its weight. PPS-style rules therefore reduce to natural weight-proportional sampling schemes at the *group* level. We then choose weights with k high-value groups (which determine OPT_k) and many low-value groups whose total Shapley mass dominates. This forces PPS sampling to spend most of its k draws on low-value groups, so the expected covered weight becomes only $O(1)$ while $\mathrm{OPT}_k = \Theta(k)$. Section C gives the full construction and bounds for each variant. □

So far in this section, we have shown a gap between specific algorithms in the class $\mathsf{Score}(Q, k)$ and greedy selection. It is an interesting **open question** to show that such a separation holds for the entire class $\mathsf{Score}(Q, k)$, which also includes approaches based on influence functions.

Remark 3.3 (Parallel sub-modular maximization)**.** *It is worth noting that there are well-known gaps between adaptive and non-adaptive methods for submodular maximization (Balkanski & Singer, 2018; Li et al., 2020), and this line of work has led to parallel algorithms that achieve optimal approximation using only logarithmic adaptive rounds (Breuer et al., 2020). However, the hard instances used to establish lower bounds in these results do not resemble canonical learning problems and are heavily stylized to ensure that large query sets (e.g., near-full deletions such as $[n] \setminus i$) reveal very little information.*

Taken together, the results in this section show that the most popular score-based valuation rules, Shapley score and Leave-one-out score, both fail despite access to an exact-value oracle and the submodularity of the valuation function. This raises the question: what additional assumptions, beyond submodularity, can enable score-based rules to perform well? We explore this next.

4 When Do Shapley and LOO Work? Curvature Conditions Restore Guarantees

The previous section highlighted that even under monotone submodularity, redundancy can collapse the pointwise credit for score-based methods. In this section, we impose one structural condition that rules out this collapse. *Bounded curvature* forces every element to retain a nontrivial fraction of its standalone marginal even after the rest of the dataset is seen, allowing fixed scores to track contextual marginals along a greedy trajectory.

Definition 4.1 (Curvature)**.** *Let $v : 2^{[n]} \to \mathbb{R}_+$ be monotone, and submodular with $v(\emptyset) = 0$. The (total) curvature of v is*

$$\kappa \;\triangleq\; 1 - \min_{i \in [n]} \frac{\Delta(i \mid [n] \setminus \{i\})}{\Delta(i \mid \emptyset)} \;, \tag{15}$$

with the convention that the ratio equals 1 *when* $\Delta(i \mid \emptyset) = 0$*. Equivalently, for every* $i \in [n]$*,*

$$\Delta(i \mid [n] \setminus \{i\}) \;\geq\; (1-\kappa)\,\Delta(i \mid \emptyset) \;. \tag{16}$$

Curvature is a common assumption in submodular maximization (Vondrak, 1978; Iyer et al., 2013). In our context, it measures redundancy: if an element looks useful in isolation but becomes nearly useless in the presence of many substitutes, then $\Delta(i \mid [n] \setminus \{i\})$ collapses relative to $\Delta(i \mid \emptyset)$ and κ approaches 1. The next lemma shows that when $\kappa < 1$, both LOO and Shapley scores become uniform proxies for the true marginal $\Delta(i \mid S)$ at every intermediate set S. This allows mimicking greedy selection, which immediately leads to the subsequent theorem.

Lemma 4.2 (Curvature links global scores to contextual marginals)**.** *If* v *has curvature* $\kappa < 1$*, then for every* $T \subseteq [n]$ *and* $i \notin T$*,*

$$(1-\kappa)\,\Delta(i \mid T) \;\leq\; s_i^{\mathrm{LOO}} \;\leq\; \Delta(i \mid T) \;, \tag{17}$$

$$(1-\kappa)\,\Delta(i \mid T) \;\leq\; s_i^{\mathrm{Sh}} \;\leq\; \frac{1}{1-\kappa}\,\Delta(i \mid T) \;. \tag{18}$$

Theorem 4.3 (Top-k by LOO or Shapley under bounded curvature)**.** *Let* $v : 2^{[n]} \to \mathbb{R}_+$ *be normalized, monotone, submodular with curvature* $\kappa < 1$*, then*

$$v(T_k^{\mathrm{LOO}}) \;\geq\; \left(1 - e^{-(1-\kappa)}\right) \mathrm{OPT}_k \;, \tag{19}$$

$$v(T_k^{\mathrm{Sh}}) \;\geq\; \left(1 - e^{-(1-\kappa)^2}\right) \mathrm{OPT}_k \;. \tag{20}$$

Proof sketch. View each top-k rule as building a set sequentially by repeatedly selecting the largest remaining score. Lemma 4.2 implies that the chosen element achieves, respectively, a $(1-\kappa)$-approximate greedy step for LOO and a $(1-\kappa)^2$-approximate greedy step for Shapley. Combine this with the standard monotone submodular inequality $\max_{i \notin S} \Delta(i \mid S) \geq (\mathrm{OPT}_k - v(S))/k$ and unroll the resulting recursion. See Appendix D for the full proof. □

When curvature is bounded away from 1, an element retains a nontrivial fraction of its standalone marginal even in the presence of many near-substitutes. This rules out the duplicate-archetype pathology from Section 3 and makes both LOO and Shapley behave as global surrogates for contextual marginal gains. Taken together, the negative results from the previous section and the guarantees here suggest that curvature captures the right redundancy parameter for score-then-select. This viewpoint also yields two practical insights: (i) deduplication (for example, clustering substitutes and keeping a representative) aims to increase the effective $(1-\kappa)$ of the induced valuation problem, and (ii) valuing at a coarser granularity than individual points (for example, clusters, sources, or clients) reduces within-group substitutability, improves curvature, and thereby strengthens the reliability of score-based selection.

5 Learning Often Violates Submodularity, but Large Cores Restore it Approximately

So far, we have shown that sub-modularity of the valuation functions, either alone or in conjunction with other assumptions, is crucial for data valuation methods to work. We now take a step back and examine a more basic question: do learning-induced value functions $v(\cdot)$ satisfy submodularity In the first place? The following subsection answers no and isolates a common mechanism.

5.1 Boundary effects create supermodular spikes

Learning objectives often exhibit *early complementarity* near decision boundaries: the first few points that "unlock" a decision boundary or a local region act as complements. This complementarity implies that a point that has little value in isolation becomes valuable after another point appears. These increasing returns on small sets lead to strong, explicit violations of diminishing returns and submodularity. We quantify the extent to which a valuation function deviates from submodularity using the following definition.

Definition 5.1 (Approximate submodularity). *Let $v : 2^n \to \mathbb{R}$ then the* submodularity defect *of v is*

$$\sigma^\star(v) := \sup_{A \subseteq B,\, i \notin B} \big[\Delta(i \mid B) - \Delta(i \mid A)\big]_+.$$

*We call v ε-*submodular *if $\sigma^\star(f) \le \varepsilon$.*

Remark 5.2 (Selection under approximate submodularity). *The relaxation in Theorem 5.1 still supports meaningful guarantees for greedy-type selection and hence serves as a natural "diagnostic" for when subset selection should remain tractable. For a monotone nonnegative objective, an additive defect bound $\sigma^\star(f) \le \varepsilon$ yields a Nemhauser & Wolsey (1978)-style guarantee with an additive degradation: greedy returns a size-k set T_k^{gr} with $v(T_k^{\mathrm{gr}}) \ge (1-1/e)\,\mathrm{OPT}_k - O(k\varepsilon)$. Our guarantees in Theorem 4.3 also suffer a factor of $-O(k\varepsilon)$, as they essentially mimic greedy analysis.*

We next compute these quantities for a simple problem.

Example 1: a midpoint threshold learner. Consider one-dimensional binary classification with true threshold $t^\star = \frac{1}{2}$ and labels $y(x) = \mathbf{1}\{x > t^\star\}$ under the uniform test distribution on $[0, 1]$. Given a finite training set $S \subset [0, 1]$, define $S^- = \{x \in S :\ x < t^\star\}$ and $S^+ = \{x \in S :\ x > t^\star\}$. If $S^- \neq \emptyset$ and $S^+ \neq \emptyset$, let $L(S) = \max S^-$ and $R(S) = \min S^+$, and set the learned threshold $\widehat{t}(S) = \frac{1}{2}(L(S) + R(S))$. If one side is missing, the learner predicts the observed label everywhere (a constant classifier). This learner creates a sharp complementarity spike: adding the first point on the missing side changes the model class from constant to thresholded, yielding a nontrivial jump in accuracy. In contrast, adding a point on the already-present side may not change the classifier.

Proposition 5.3. *For the midpoint threshold learner above, $\sigma^\star(f) = \frac{1}{4}$, and $\alpha(f) = 0$. In particular, f is $\frac{1}{4}$-submodular but fails every multiplicative weak-submodularity inequality with $\alpha > 0$.*

See Appendix E for the proof of the above proposition.

Example 2: a minimal tweak of coverage. Our negative results for score-based valuation methods in Section 3 used a coverage-based learning problem in which each archetype contributes once after the set contains a representative. A single tweak to this simple example produces early complementarity: require r representatives of an archetype before it contributes. Formally, partition the ground set into groups $G_1, \ldots, G_m$ and define, for an integer $r \ge 2$,

$$V_r(U) \ := \ \sum_{j=1}^{m} w_j\, \mathbf{1}\big\{|U \cap G_j| \ge r\big\}, \qquad w_j > 0. \tag{21}$$

When $r = 1$, V_1 is (weighted) coverage and is monotone submodular. When $r \ge 2$, a group yields zero value until the set accumulates enough supporting points inside that group. This creates increasing returns near the activation boundary $|U \cap G_j| = r - 1$. In particular, we can show the following proposition (proved in Appendix E).

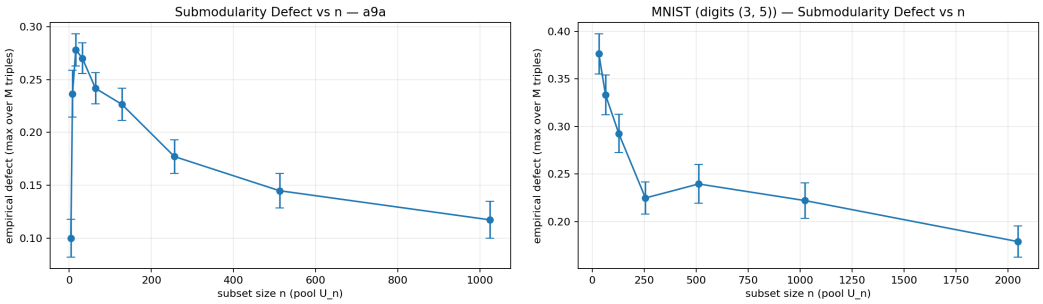
Proposition 5.4 (Thresholded coverage violates both submodularity and supermodularity). *Fix any $r \ge 2$. There exist sets $A \subseteq B$ and an element $x \notin B$ such that $\Delta(x \mid A) < \Delta(x \mid B)$ for V_r, and there also exist $A' \subseteq B'$ and $x' \notin B'$ such that $\Delta(x' \mid A') > \Delta(x' \mid B')$.*

The above result highlights that even canonical learning problems with submodular valuations can be easily altered to introduce complementary spikes. It turns out that a second, equally simple change makes the situation strictly worse: if different archetypes require different amounts of local support (unknown to the algorithm), then *no* (adaptive) selection rule can guarantee a constant-factor approximation, even though the objective remains monotone and each archetype behaves like a thresholded "bundle."

Concretely, we keep the thresholded-activation objective

$$V(U) \ = \ \frac{1}{k}\,\big|\{j :\ |U \cap G_j| \ge r_j\}\big|,$$

but we let the thresholds $(r_1, \ldots, r_k)$ vary across groups and hide which groups are "feasible" until the algorithm invests enough samples in them. The next theorem (proof in Appendix E) formalizes this: the algorithm must spend about r_0 selections to even learn whether a group can ever pay off, so with a budget k it can test only about k/r_0 groups, and only a $1/r_0$ fraction of those tests succeed.



(a) a9a (logistic regression). (b) MNIST (digits 3 vs. 5, small CNN).

Figure 1: Empirical core-relative submodularity defect $\widehat{\sigma}(U_n)$ as a function of core size n. For each n, we sample a pool U_n, sample M triples (A, B, i) with $A \subset B \subseteq U_n$, and take the maximum violation $\max[\Delta(i \mid B) - \Delta(i \mid A)]_+$; we then average this statistic over R independent pools and plot 95% confidence intervals. As n grows, both tasks show smaller violations, consistent with the view that a large, diverse core makes approximate diminishing returns a better approximation.

Theorem 5.5 (Heterogeneous thresholds: no constant-factor adaptivity)**.** *Fix integers $k \geq 2$ and $r_0 \in \{2, \ldots, \lfloor \frac{k}{8 \ln 2} \rfloor\}$. There exists a distribution over instances with k groups $(G_1, \ldots, G_k)$ and thresholds $(r_1, \ldots, r_k)$ such that, for every (possibly randomized and adaptive) selection policy Π that outputs a set T_k^{Π} of size at most k such that,*

$$\frac{\mathbb{E}[V(U_{\mathcal{A}})]}{\mathbb{E}[\mathrm{OPT}]} \leq \frac{4}{r_0} .$$

Taking, for instance, $r_0 = \lfloor \sqrt{k} \rfloor$ yields an $O(1/\sqrt{k})$ upper bound on the approximation ratio.

The results in this section so far show that complementary spikes that lead to supermodularity are common even in simple learning problems and can be detrimental to data valuation. Fortunately, the positive results we developed in Section 4 can be salvaged by observing that simple coverage assumptions can often recover approximate submodularity.

5.2 Approximate Submodularity Emerges with a Large Core

The examples in the previous subsection show that learning objectives can exhibit *early complementarity*: a few points jointly unlock a large accuracy gain when the current training set misses a critical region (such as one side of a decision boundary). In many valuation pipelines, one adds a small number of points on top of an already large *core* dataset. In that regime, we should judge diminishing returns *above the core* rather than above the empty set.

Definition 5.6 (Core-relative additive defect)**.** *Let $v : 2^n \to \mathbb{R}$ and fix a core set $S_0 \subseteq S$. Define the* core-relative submodularity defect *by*

$$\sigma^{\star}(v; S_0) := \sup_{S_0 \subseteq A \subseteq B,\ i \notin B} \big[\Delta(i \mid B) - \Delta(i \mid A)\big]_+ .$$

*We call f ε-*submodular above S_0 *if $\sigma^{\star}(f; S_0) \leq \varepsilon$.*

We revisit the one-dimensional threshold task

$$y(x) = \mathbf{1}\{x > \tfrac{1}{2}\}, \qquad x \sim \mathrm{Unif}[0, 1],$$

together with the midpoint learner from the previous subsection. If a core set B already contains at least one point on each side of $\frac{1}{2}$, then the midpoint learner already brackets the decision boundary. In that case, adding a single extra point can only tighten the bracketing interval, so it can only reduce the current test error by an amount comparable to the current bracketing gap. As a result, the "increasing returns" effect above such a core becomes small. The next proposition (proved in Appendix F) quantifies this.

Proposition 5.7. *Assume $m \geq 4$ and let $S = \{X_1, \ldots, X_m\}$ with $X_j \overset{\text{iid}}{\sim} \text{Unif}[0,1]$, and let v be the midpoint-learner value function, with $\mathcal{D} = \text{Unif}[0,1]$. Then $\mathbb{E}\big[\sigma^\star(v; S_m)\big] \leq \frac{4}{m}$.*

Extending guarantees of the above flavor to other learning problems is an **open question**, and might require novel regularity assumptions about the decision boundary.

Empirical evidence on canonical learners (core size vs. defect). We now test the "large core" prediction on two standard learning pipelines: logistic regression on `a9a` (LIBSVM) and a small CNN on MNIST (digits 3 vs. 5). For each task, we form a random *core pool* U_n of size n from the training set, and we view $v(S)$ as the test accuracy of the learner trained from scratch on the labeled subset $S \subseteq U_n$. To approximate the core-relative defect, we sample M random triples (A, B, i) with $A \subset B \subseteq U_n$ and $i \in U_n \setminus B$, and compute

$$\widehat{\sigma}(U_n) := \max_{m \in [M]} \big[\Delta(i_m \mid B_m) - \Delta(i_m \mid A_m)\big]_+ \ .$$

Figure 1 shows a clear decline in the measured defect as n grows, which supports the qualitative message of this subsection: once the training set already covers the relevant parts of the input distribution, the sharp "unlocking" effects become rarer, and diminishing returns become a more reasonable model *above* the core. The results of this subsection, along with Remark 5.2, highlight that some form of coverage is essential for valuation in learning tasks.

6 Discussion

Our theory suggests that stable score-based selection typically requires two preprocessing steps targeting the failure modes. The first step is to deduplicate or decide to perform valuation at a higher granularity. A practical way to do this is to cluster near-duplicates (for example, in representation space) and select at the cluster level, which reduces the effective redundancy that drives score collapse. The second step is to build a rich coverage core to suppress boundary spikes. Among other works, both these insights strongly echo the empirical takeaways of Ilyas et al. (2022).

Before we conclude, we note that there may be many other failure modes in valuation methods. One of them is anti-complementarity. In the following result, we show how anti-complementarity forces every non-adaptive algorithm in the class $\mathsf{Score}(nk, k) = \mathsf{NA}(nk, k)$, including influence functions, to fail to approximate the best adaptive algorithm.

Theorem 6.1 (Greedy v/s $\mathsf{NA}(nk, k)$)**.** *Assume $k \geq 6$ and $n \geq k^2$. For every (possibly randomized)[3] selection algorithm $\Pi \in \mathsf{NA}(nk, k)$, there exists a valuation function $v : 2^{[n]} \to \mathbb{R}$ s.t.:*

1. *Greedy selection outputs T_k^{gr} with $v(T_k^{\text{gr}}) = k = \text{OPT}_k$ using at most nk exact value queries.*
2. *Π outputs a set T^Π with $|T^\Pi| \leq k$ such that $\mathbb{E}[v(T^\Pi)] \leq 6$.*

Proof sketch. Fix $k \geq 6$ and $B \geq 2$, and set $n := kB$. Let $S = \bigsqcup_{i=1}^k L_i$ where each layer $L_i = \{z_{(i,1)}, \ldots, z_{(i,B)}\}$. Nature draws $J_1, \ldots, J_k$ i.i.d. uniformly from $[B]$, to pick the unique correct element in layer i as $z_{(i,J_i)}$. We say that a set T *passes layer* i if it contains $z_{(i,J_i)}$ and contains no other element from L_i. Then we define the valuation function,

$$v(S) := \max\{t \in \{0, 1, \ldots, k\} : \ S \text{ passes layers } 1, \ldots, t\}.$$

Notably, the above valuation is neither monotone nor submodular, yet greedy selection remains optimal. For any one-round algorithm, let E_{i-1} be the event that some query returns value at least $i-1$. A union bound shows $\Pr[E_{i-1}] \leq Q/B^{i-1}$. On $\neg E_{i-1}$, all queried values are at most $i-2$, so the transcript depends only on $(J_1, \ldots, J_{i-1})$ and is independent of J_i; since the output is a function of the transcript, it cannot correlate with J_i on $\neg E_{i-1}$, so it passes layer i with probability at most $1/B$. This yields the recursion $p_i := \Pr[v(U) \geq i] \leq \Pr[E_{i-1}] + \frac{1}{B} p_{i-1}$. Solving gives $p_i \leq iQ/B^{i-1} + 1/B^i$, and summing $\mathbb{E}[v(U)] = \sum_{i=1}^k p_i$ gives the stated for an appropriate value of B. A Yao minimax argument extends this to randomized non-adaptive algorithms. □

[3] While we defined $\mathsf{NA}(Q, k)$ only using deterministic maps of the oracle transcripts, it is easy to extend the class to randomized non-adaptive algorithms by outputting a distribution over $\mathsf{NA}(Q, k)$, which translates to a distribution over the output set.

REFERENCES

Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023.

Eric Balkanski and Yaron Singer. The adaptive complexity of maximizing a submodular function. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pp. 1138–1151, 2018.

Adam Breuer, Eric Balkanski, and Yaron Singer. The fast algorithm for submodular maximization. In *International Conference on Machine Learning*, pp. 1134–1143. PMLR, 2020.

Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1433–1452. SIAM, 2014.

Niv Buchbinder, Moran Feldman, Joseph Seffi, and Roy Schwartz. A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing*, 44 (5):1384–1402, 2015.

Raul Castro Fernandez. Data ecology: Understanding and designing data ecosystems. *ACM SIGMOD Record*, 54(4):25–26, 2026.

Tyler Chen, Akshay Seshadri, Mattia J Villani, Pradeep Niroula, Shouvanik Chakrabarti, Archan Ray, Pranav Deshpande, Romina Yalovetzky, Marco Pistoia, and Niraj Kumar. A unified framework for provably efficient algorithms to estimate shapley values. *arXiv preprint arXiv:2506.05216*, 2025.

R Dennis Cook and Sanford Weisberg. Residuals and influence in regression. 1982.

Michiel Debruyne, Mia Hubert, and Johan AK Suykens. Model selection in kernel based regression using the influence function. *Journal of machine learning research.-Cambridge, Mass.*, 9:2377–2400, 2008.

Junwei Deng, Yuzheng Hu, Pingbang Hu, Ting-Wei Li, Shixuan Liu, Jiachen T Wang, Dan Ley, Qirun Dai, Benhao Huang, Jin Huang, et al. A survey of data attribution: Methods, applications, and evaluation in the era of generative ai. 2025.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels. *arXiv preprint arXiv:2401.12926*, 2024.

Uriel Feige, Vahab S Mirrokni, and Jan Vondrák. Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, 40(4):1133–1153, 2011.

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

Zayd Hammoudeh and Daniel Lowd. Training data influence analysis and estimation: A survey. *Machine Learning*, 113(5):2351–2403, 2024.

Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. Explaining black box predictions and unveiling data artifacts through influence functions. *arXiv preprint arXiv:2005.06676*, 2020.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting predictions from training data. *arXiv preprint arXiv:2202.00622*, 2022.

Rishabh K Iyer, Stefanie Jegelka, and Jeff A Bilmes. Curvature and optimal algorithms for learning and minimizing submodular functions. *Advances in neural information processing systems*, 26, 2013.

P Kairouz, HB McMahan, B Avent, A Bellet, M Bennis, AN Bhagoji, K Bonawitz, Z Charles, G Cormode, R Cummings, et al. Advances and open problems in federated learning. arxiv. *arXiv preprint arXiv:1912.04977*, 2019.

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

Wenzheng Li, Paul Liu, and Jan Vondrák. A polynomial lower bound on adaptive complexity of submodular maximization. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 140–152, 2020.

George L Nemhauser and Laurence A Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

Ramesh Raskar, Praneeth Vepakomma, Tristan Swedish, and Aalekh Sharan. Data markets to support ai for all: Pricing, valuation and governance. *arXiv preprint arXiv:1905.06462*, 2019.

Andrea Sestino, Adham Kahlawi, and Andrea De Mauro. Decoding the data economy: a literature review of its impact on business, society and digital transformation. *European Journal of Innovation Management*, 28(2):298–323, 2025.

Lloyd S Shapley et al. A value for n-person games. 1953.

Zhihua Tian, Jian Liu, Jingyu Li, Xinle Cao, Ruoxi Jia, Jun Kong, Mengdi Liu, and Kui Ren. Private data valuation and fair payment in data marketplaces. *arXiv preprint arXiv:2210.08723*, 2022.

Jan Vondrak. Submodularity and curvature: The optimal algorithm. *Annals of Discrete Math*, 2: 65–74, 1978.

Jiachen T Wang, Prateek Mittal, Dawn Song, and Ruoxi Jia. Data shapley in one training run. *arXiv preprint arXiv:2406.11011*, 2024.

Luyang Zhang, Cathy Jiao, Beibei Li, and Chenyan Xiong. Fairshare data pricing via data valuation for large language models. *arXiv preprint arXiv:2502.00198*, 2025.

Mengxiao Zhang, Fernando Beltrán, and Jiamou Liu. A survey of data pricing for data marketplaces. *IEEE Transactions on Big Data*, 9(4):1038–1056, 2023.

A Proofs for Section 2

A.1 Proof of Proposition 2.3

Proof. We prove both containments.

$\mathsf{Score}(Q,k) \subseteq \mathsf{NA}(Q,k)$**.** A score-based method chooses all Q query sets non-adaptively, observes the transcript $\mathcal{T}$, computes scores $s_x(\mathcal{T})$, and outputs a set U as a deterministic function of $\mathcal{T}$ (the tie-breaking rule is fixed). Therefore it is a one-round non-adaptive algorithm, so it lies in $\mathsf{NA}(Q,k)$.

$\mathsf{NA}(Q,k) \subseteq \mathsf{Score}(Q,k)$**.** Fix any algorithm $\mathcal{A} \in \mathsf{NA}(Q,k)$. By definition, $\mathcal{A}$ consists of (i) a fixed non-adaptive query list $S^{(1)}, \ldots, S^{(Q)}$, and (ii) a deterministic mapping $\mathcal{T} \mapsto U(\mathcal{T})$ with $|U(\mathcal{T})| \le k$.

We construct a score-based method $\mathcal{A}'$ that uses the same Q queries and outputs exactly $U(\mathcal{T})$ for every transcript. Fix any deterministic total order $\prec$ on V, and let $\mathrm{rank}_{\prec}(x) \in \{1, \ldots, |V|\}$ denote the rank of x in this order. Fix a constant $\epsilon := 1/(10|V|)$. Given transcript $\mathcal{T}$, define scores

$$s_x(\mathcal{T}) := \begin{cases} 2 & \text{if } x \in U(\mathcal{T}), \\ -\epsilon \cdot \mathrm{rank}_{\prec}(x) & \text{if } x \notin U(\mathcal{T}). \end{cases}$$

Consider any feasible $T \subseteq V$ with $|T| \le k$. If $T \subseteq U(\mathcal{T})$, then $\sum_{x \in T} s_x(\mathcal{T}) = 2|T| \le 2|U(\mathcal{T})|$. If T contains any element outside $U(\mathcal{T})$, then $\sum_{x \in T} s_x(\mathcal{T}) \le 2|T \cap U(\mathcal{T})| - \epsilon < 2|U(\mathcal{T})|$. Hence the unique maximizer of $\sum_{x \in T} s_x(\mathcal{T})$ over $|T| \le k$ is precisely $T = U(\mathcal{T})$ (no superset can beat it because extra elements have negative score). Therefore the score-based output equals $U(\mathcal{T})$ for every transcript, as desired. Thus $\mathcal{A} \in \mathsf{Score}(Q,k)$, proving $\mathsf{NA}(Q,k) \subseteq \mathsf{Score}(Q,k)$.

Combining the two containments yields $\mathsf{Score}(Q,k) = \mathsf{NA}(Q,k)$. □

Corollary A.1 (Randomized equivalence). *Fix V, k, Q. Let $\mathsf{NA}^{\mathrm{rand}}(Q,k)$ denote the class of randomized one-round non-adaptive algorithms with Q value queries and output size at most k. Let $\mathsf{Score}^{\mathrm{rand}}(Q,k)$ denote the class of randomized score-based methods with the same budgets.*

Then $\mathsf{NA}^{\mathrm{rand}}(Q,k) = \mathsf{Score}^{\mathrm{rand}}(Q,k)$.

Proof. A randomized one-round algorithm induces a distribution over deterministic one-round algorithms via its internal randomness. Likewise, a randomized score-based method induces a distribution over deterministic score-based methods. Apply Proposition 2.3 pointwise to each deterministic algorithm in the support to obtain a bijection between the two distributions, preserving the output set as a function of the transcript. Therefore the two randomized classes are equal. □

B Proof of Theorem 3.1

B.1 Coverage from 1–NN and submodularity

Lemma B.1 (Coverage is monotone submodular). *The function $v(\cdot)$ in equation 13 is monotone and submodular, and satisfies $v(\emptyset) = 0$.*

Proof. Monotonicity follows since adding elements can only change indicators in equation 13 from 0 to 1. To prove submodularity, fix $A \subseteq B \subseteq [n]$ and $i \in [n] \setminus B$. Let $r(i)$ denote the unique index with $i \in \mathcal{G}_{r(i)}$. Then

$$v(A \cup \{i\}) - v(A) \;=\; M_{r(i)} \cdot \mathbb{I}\{A \cap \mathcal{G}_{r(i)} = \emptyset\} \;, \tag{22}$$

and the same identity holds with A replaced by B. Since $A \subseteq B$, the indicator for A dominates the indicator for B, which yields diminishing returns:

$$v(A \cup \{i\}) - v(A) \;\geq\; v(B \cup \{i\}) - v(B) \;. \tag{23}$$

Finally, $v(\emptyset) = 0$ follows from equation 13 . □

B.2 A Shapley failure instance

Instance. Fix $k \geq 2$ and $\varepsilon \in (0,1)$. Let $R = k$. Let $\mathcal{G}_0$ have size k and weight $M_0 > 0$. For each $r \in \{1, \ldots, k\}$, let $\mathcal{G}_r$ have size k and weight $M_r = (1-\varepsilon)M_0$. Define $v(\cdot)$ by equation 13. By Theorem B.1, $v(\cdot)$ is monotone submodular.

Lemma B.2 (Shapley score within a duplicate group). *Fix a group $\mathcal{G}_r$ of size m_r and weight M_r. Every item $i \in \mathcal{G}_r$ satisfies*

$$s_i^{\mathrm{Sh}} \;=\; \frac{M_r}{m_r} \;. \tag{24}$$

Proof. Fix $i \in \mathcal{G}_r$ and sample a uniform permutation π of $[n]$. The marginal $v(P_i^\pi \cup \{i\}) - v(P_i^\pi)$ equals M_r exactly when P_i^π contains no other element of $\mathcal{G}_r$, and equals 0 otherwise. By symmetry, i appears first among the m_r elements of $\mathcal{G}_r$ with probability $1/m_r$. Taking expectation yields equation 24 . □

By Theorem B.2, every $i \in \mathcal{G}_0$ has score M_0/k, while every $i \in \mathcal{G}_r$ for $r \geq 1$ has score $(1-\varepsilon)M_0/k$. Thus every element of $\mathcal{G}_0$ ranks above every element outside $\mathcal{G}_0$, so Shapley-top-k selects $\mathcal{G}_0$ and achieves $v(T_k^{\mathrm{Sh}}) = M_0$. Selecting one representative from each of $\mathcal{G}_1, \ldots, \mathcal{G}_k$ yields value $\sum_{r=1}^k M_r = k(1-\varepsilon)M_0$, so $\mathrm{OPT}_k \geq k(1-\varepsilon)M_0$.

B.3 An LOO failure instance with random tie-breaking

Instance. We extend the previous template by adding many low-value groups to dilute random tie-breaking while preserving a large OPT_k. Fix $k \geq 2$. Let $R = k + k^3$. Define groups and weights as follows:

- (*Decoy group.*) $\mathcal{G}_0$ has size k and weight $M_0 > 0$.

- (*High-value groups.*) For $r \in \{1, \dots, k\}$, $\mathcal{G}_r$ has size k^2 and weight $M_r = M_0$.
- (*Low-value groups.*) For $r \in \{k+1, \dots, k+k^3\}$, $\mathcal{G}_r$ has size k^2 and weight $M_r = M_0/k^2$.

Define $v(\cdot)$ by equation 13. By Theorem B.1, $v(\cdot)$ is monotone submodular.

Lemma B.3 (All LOO scores tie)**.** *For every $i \in [n]$, $s_i^{\mathrm{LOO}} = 0$.*

Proof. Every group has size at least 2, so the set $[n] \setminus \{i\}$ still intersects every group. Hence $v([n]) = v([n] \setminus \{i\})$ for all i, and equation 6 gives $s_i^{\mathrm{LOO}} = 0$. □

Lemma B.4 (A hit-probability bound)**.** *Let T be a uniform random size-k subset of $[n]$ and let $\mathcal{G} \subseteq [n]$ have size m. Then*

$$\mathbb{P}[T \cap \mathcal{G} \neq \emptyset] \;\le\; \frac{k\,m}{n-k+1} \;. \tag{25}$$

Proof. Sample T without replacement as $(I_1, \dots, I_k)$. For each t, conditioned on the first $t-1$ draws, the t-th draw is uniform over a set of size at least $n-k+1$, so $\mathbb{P}[I_t \in \mathcal{G}] \le m/(n-k+1)$. A union bound over $t \in \{1, \dots, k\}$ yields equation 25 . □

By Theorem B.3, LOO with random tie-breaking draws a uniform random size-k subset $\widehat{T}_k^{\mathrm{LOO}}$. Using equation 13 and linearity of expectation,

$$\mathbb{E}\Big[v(\widehat{T}_k^{\mathrm{LOO}})\Big] \;=\; \sum_{r=0}^{R} M_r \cdot \mathbb{P}\Big[\widehat{T}_k^{\mathrm{LOO}} \cap \mathcal{G}_r \neq \emptyset\Big] \;. \tag{26}$$

In the construction above,

$$n \;=\; |\mathcal{G}_0| + \sum_{r=1}^{k} |\mathcal{G}_r| + \sum_{r=k+1}^{k+k^3} |\mathcal{G}_r| \;=\; k + k \cdot k^2 + k^3 \cdot k^2 \;=\; k + k^3 + k^5 \;. \tag{27}$$

Apply Theorem B.4 with $m = k$ for $\mathcal{G}_0$ and with $m = k^2$ for all other groups:

$$\begin{aligned}\mathbb{E}\Big[v(\widehat{T}_k^{\mathrm{LOO}})\Big] &\le M_0 \cdot \frac{k \cdot k}{n-k+1} \;+\; k \cdot M_0 \cdot \frac{k \cdot k^2}{n-k+1} \;+\; k^3 \cdot \frac{M_0}{k^2} \cdot \frac{k \cdot k^2}{n-k+1} \\ &= M_0 \cdot \frac{k^2 + 2k^4}{n-k+1} \;. \end{aligned} \tag{28}$$

Since $n \ge k^5$ and $n - k + 1 \ge k^5/2$ for all $k \ge 2$, we obtain

$$\mathbb{E}\Big[v(\widehat{T}_k^{\mathrm{LOO}})\Big] \;\le\; M_0 \left(\frac{2k^2}{k^5} + \frac{4k^4}{k^5}\right) \;\le\; \frac{3}{k}\, M_0 \;, \tag{29}$$

which proves the claim.

C Proportional-to-Shapley variants: definitions and proofs

C.1 Capped-coverage instances and Shapley structure

We use the capped-coverage (1–NN) family. The ground set $[n]$ partitions into disjoint clusters $C_1, \dots, C_B$. Each cluster b has a weight $w_b > 0$. The value of a set $S \subseteq [n]$ is

$$v(S) \;=\; \sum_{b=1}^{B} w_b \cdot \mathbb{I}\{S \cap C_b \neq \varnothing\} \;. \tag{30}$$

This v is normalized, monotone, and submodular.

For any $k \le B$, an optimal k-set picks one element from each of the k largest-weight clusters, so

$$\mathrm{OPT}_k \;=\; \max_{|S| \le k} v(S) \;=\; \sum_{b=1}^{k} w_{(b)} \;, \tag{31}$$

where $w_{(1)} \ge w_{(2)} \ge \cdots$ are the weights sorted in nonincreasing order.

Lemma C.1 (Shapley values for capped coverage)**.** *For the function equation 30, if $i \in C_b$ and $|C_b| = s_b$, then*

$$\varphi_i \;=\; \frac{w_b}{s_b} \qquad \textit{and} \qquad \sum_{i \in C_b} \varphi_i \;=\; w_b \;. \tag{32}$$

Proof. Fix $i \in C_b$. In a random permutation, i contributes w_b if and only if it appears before every other element of C_b; otherwise its marginal contribution to cluster b equals 0. By symmetry, $\Pr(i \text{ is first in } C_b) = 1/s_b$, so $\varphi_i = w_b/s_b$. Summing over the s_b elements in C_b yields $\sum_{i \in C_b} \varphi_i = w_b$. □

C.2 Variant A: PPS with replacement

Rule. Draw k times independently with replacement, where $\Pr(i) = \varphi_i / \sum_{j=1}^{n} \varphi_j$. Let S_k be the set of distinct drawn elements.

By Theorem C.1, each draw hits cluster b with probability

$$p_b \;=\; \frac{\sum_{i \in C_b} \varphi_i}{\sum_{j=1}^{n} \varphi_j} \;=\; \frac{w_b}{W} \qquad \text{where} \qquad W := \sum_{b=1}^{B} w_b \;. \tag{33}$$

Cluster b is covered after k draws with probability $1 - (1 - p_b)^k$, so

$$\mathbb{E}[v(S_k)] \;=\; \sum_{b=1}^{B} w_b \Big(1 - (1 - p_b)^k\Big) \;. \tag{34}$$

Proposition C.2. *For every integer $k \geq 2$, there exists a capped-coverage instance equation 30 such that PPS with replacement satisfies*

$$\frac{\mathbb{E}[v(S_k)]}{\mathrm{OPT}_k} \;\leq\; \frac{2}{k} \;. \tag{35}$$

Proof. Fix $k \geq 2$ and construct:

- k *good* clusters with $w_b = 1$ for $b = 1, \ldots, k$;
- $M = k^4$ *bad* clusters with $w_b = \alpha := 1/k^2$ for $b = k+1, \ldots, k+M$.

Then $\mathrm{OPT}_k = k$. Moreover

$$W_{\text{good}} = k, \qquad W_{\text{bad}} = M\alpha = k^4 \cdot \tfrac{1}{k^2} = k^2, \qquad W = k + k^2.$$

For a good cluster, $p_b = 1/W$, so using $1 - (1-x)^k \leq kx$,

$$1 - (1 - p_b)^k \leq \frac{k}{W} \leq \frac{k}{k^2} = \frac{1}{k},$$

hence the total expected good contribution is at most $k \cdot 1 \cdot (1/k) = 1$.

For a bad cluster, $p_b = \alpha/W \leq (1/k^2)/k^2 = 1/k^4$, so

$$1 - (1 - p_b)^k \leq k p_b \leq \frac{k}{k^4} = \frac{1}{k^3}.$$

Thus each bad cluster contributes at most $\alpha \cdot (1/k^3) = 1/k^5$ in expectation, and summing over $M = k^4$ bad clusters gives total expected bad contribution at most $1/k$.

Therefore $\mathbb{E}[v(S_k)] \leq 1 + 1/k$, and dividing by $\mathrm{OPT}_k = k$ yields $\mathbb{E}[v(S_k)]/\mathrm{OPT}_k \leq (1 + 1/k)/k \leq 2/k$. □

C.3 VARIANT B: PPS WITHOUT REPLACEMENT (FIXED SHAPLEY WEIGHTS)

Rule. Sample k distinct items sequentially without replacement, where at step t we pick $i \notin S_{t-1}$ with probability proportional to its fixed Shapley weight φ_i.

Proposition C.3. *For every integer $k \geq 2$, there exists a capped-coverage instance equation 30 such that PPS without replacement satisfies*

$$\frac{\mathbb{E}[v(S_k)]}{\mathrm{OPT}_k} \;\leq\; \frac{3}{k} \;. \tag{36}$$

Proof. Use the same weights as in Theorem C.2, and set all cluster sizes equal and large: $|C_b| = s := k^4$ for every cluster.

Then by Theorem C.1, every good item has Shapley weight $1/s = 1/k^4$ and every bad item has Shapley weight $\alpha/s = (1/k^2) \cdot (1/k^4) = 1/k^6$. The total Shapley mass equals

$$\sum_{i=1}^{n} \varphi_i = \sum_{b=1}^{B} w_b = W = k + k^2.$$

At each draw, the maximum removed Shapley mass is $1/k^4$. After $t-1 \leq k-1$ steps, the removed mass is at most $k/k^4 = 1/k^3$, so the remaining mass is at least $W - 1/k^3 \geq W/2$ for all $k \geq 2$.

The total Shapley mass of *all* good items is exactly $\sum_{b \leq k} w_b = k$, so at any step

$$\Pr(\text{pick a good item at step } t \mid \text{history}) \;\leq\; \frac{k}{W/2} \;=\; \frac{2k}{k+k^2} \;\leq\; \frac{2}{k}.$$

Summing over k steps yields $\mathbb{E}[\#\{\text{good items selected}\}] \leq 2$. Each covered good cluster contributes at most 1, so the expected good contribution is at most 2.

Any covered bad cluster contributes at most $\alpha = 1/k^2$, and at most k clusters can be covered by a size-k set, so the bad contribution is at most $k\alpha = 1/k$.

Thus $\mathbb{E}[v(S_k)] \leq 2 + 1/k$, and dividing by $\mathrm{OPT}_k = k$ gives $\mathbb{E}[v(S_k)]/\mathrm{OPT}_k \leq (2 + 1/k)/k \leq 3/k$. □

C.4 VARIANT C: ADAPTIVE PPS VIA RESIDUAL SHAPLEY

Rule. We build S_t sequentially. Given S_t, define the residual function

$$v_{S_t}(T) = v(S_t \cup T) - v(S_t).$$

Compute Shapley values $\varphi_i^{S_t}$ for $i \notin S_t$ under v_{S_t}, and sample the next point with probability proportional to $\varphi_i^{S_t}$.

Lemma C.4 (Residual Shapley structure for capped coverage)**.** *For capped-coverage v in equation 30 and any set S:*

1. *If $S \cap C_b \neq \emptyset$, then $\varphi_i^S = 0$ for all $i \in C_b$.*
2. *If $S \cap C_b = \emptyset$, then all $i \in C_b$ have equal residual Shapley values and $\sum_{i \in C_b} \varphi_i^S = w_b$.*

Proof. If S already covers C_b, then $v_S(T)$ does not depend on any element of C_b, so all residual marginals are 0 and thus all residual Shapley values are 0. If S does not cover C_b, then in the residual game the block C_b contributes w_b if and only if at least one of its elements appears. All elements in C_b are symmetric, and Shapley efficiency assigns total Shapley mass w_b to that block, split equally among its elements. □

Proposition C.5. *For every integer $k \geq 2$, there exists a capped-coverage instance equation 30 such that adaptive PPS via residual Shapley satisfies*

$$\frac{\mathbb{E}[v(S_k)]}{\mathrm{OPT}_k} \;\leq\; \frac{3}{k} \;. \tag{37}$$

Proof. By Theorem C.4, once a cluster is covered, its remaining elements have residual Shapley 0 and never get selected again. At the cluster level, the procedure therefore samples k *distinct* clusters without replacement, choosing each next cluster proportionally to its current weight.

Use the same weight construction as in Theorem C.2 (good weights 1, bad weights $1/k^2$ with $M = k^4$ bad clusters). Then $\mathrm{OPT}_k = k$ and total initial weight is $W = k + k^2$.

At any step $t \le k$, the remaining total weight is at least $W - (t-1) \ge W - (k-1) \ge k^2$, while the remaining total good weight is at most k. Hence

$$\Pr(\text{pick a good cluster at step } t \mid \text{history}) \le \frac{k}{k^2} = \frac{1}{k}.$$

Summing over k steps yields $\mathbb{E}[\#\{\text{good clusters selected}\}] \le 1$, so the expected good contribution is at most 1. The bad contribution is at most $k\alpha = 1/k$. Thus $\mathbb{E}[v(S_k)] \le 1 + 1/k$, and dividing by $\mathrm{OPT}_k = k$ gives $\mathbb{E}[v(S_k)]/\mathrm{OPT}_k \le (1 + 1/k)/k \le 3/k$. □

C.5 Variant D: Sampling k-sets proportional to total Shapley mass

Rule. For each k-subset $S \subseteq [n]$, define $\mathrm{score}(S) := \sum_{i \in S} \varphi_i$, and sample S_k with probability proportional to $\mathrm{score}(S)$.

A useful equivalent view follows by exchanging sums:

$$\sum_{S:|S|=k} \mathrm{score}(S) = \sum_{S:|S|=k} \sum_{i \in S} \varphi_i = \sum_{i=1}^{n} \varphi_i \cdot \binom{n-1}{k-1}.$$

Hence

$$\Pr(S_k = S) = \sum_{i \in S} \frac{\varphi_i}{\sum_{j=1}^{n} \varphi_j} \cdot \frac{1}{\binom{n-1}{k-1}}. \tag{38}$$

Equation equation 38 implies the following sampling procedure: first choose a pivot item I with $\Pr(I = i) \propto \varphi_i$, then choose the remaining $k-1$ items uniformly among $[n] \setminus \{I\}$.

Proposition C.6. *For every integer $k \ge 2$, there exists a capped-coverage instance equation 30 such that k-set PPS satisfies*

$$\frac{\mathbb{E}[v(S_k)]}{\mathrm{OPT}_k} \;\le\; \frac{3}{k}\;. \tag{39}$$

Proof. Fix $k \ge 2$. Construct:

- one good cluster C_{g} of weight 1 and size $|C_{\mathrm{g}}| = k$;
- $M = k^4$ bad clusters, each of weight $\alpha := 1/k^2$ and size $|C_b| = k^6$.

Then $\mathrm{OPT}_k \ge 1$ (pick one element from C_{g} and arbitrary others), and in fact $\mathrm{OPT}_k \le 1 + k\alpha \le 2$.

The total Shapley mass equals total weight:

$$\sum_{i=1}^{n} \varphi_i = 1 + M\alpha = 1 + k^2.$$

Thus the pivot lies in C_{g} with probability $1/(1+k^2) \le 1/k^2$.

Condition on the event that the pivot lies in a bad cluster. The remaining $k-1$ points are uniform from a universe of size

$$n - 1 = |C_{\mathrm{g}}| + M \cdot k^6 - 1 \ge Mk^6 = k^{10}.$$

Hence the probability that any of these $k-1$ uniform draws hits C_{g} is at most

$$\frac{(k-1)|C_{\mathrm{g}}|}{n-1} \le \frac{k \cdot k}{k^{10}} = \frac{1}{k^8}.$$

Therefore, with probability at least $1 - 1/k^8$, the sampled set does not cover C_{g}, so its value comes only from covered bad clusters. A size-k set covers at most k clusters, each bad cluster has weight $\alpha = 1/k^2$, so $v(S_k) \le k\alpha = 1/k$ on this event.

Putting the cases together,

$$\mathbb{E}[v(S_k)] \le \Pr(\text{pivot in } C_{\mathrm{g}}) \cdot 2 + \Pr(\text{pivot bad}) \left(\frac{1}{k} + \frac{1}{k^8} \cdot 2 \right) \le \frac{2}{k^2} + \frac{1}{k} + \frac{2}{k^8} \le \frac{3}{k}.$$

Finally, since $\mathrm{OPT}_k \ge 1$, we obtain $\mathbb{E}[v(S_k)]/\mathrm{OPT}_k \le 3/k$. □

D Full proofs for Section 4

D.1 Proof of Theorem 4.2

Proof. Fix any $S \subseteq [n]$ and $i \notin S$.

LOO bounds. By definition, $s_i^{\mathrm{LOO}} = \Delta(i \mid [n] \setminus \{i\})$.

Upper bound. Submodularity implies that marginals decrease as the conditioning set grows: if $S \subseteq T \subseteq [n] \setminus \{i\}$ then $\Delta(i \mid S) \ge \Delta(i \mid T)$. Since $S \subseteq [n] \setminus \{i\}$,

$$\Delta(i \mid S) \;\ge\; \Delta(i \mid [n] \setminus \{i\}) \;=\; s_i^{\mathrm{LOO}},$$

so $s_i^{\mathrm{LOO}} \le \Delta(i \mid S)$.

Lower bound. Curvature equation 16 gives

$$s_i^{\mathrm{LOO}} = \Delta(i \mid [n] \setminus \{i\}) \;\ge\; (1 - \kappa)\, \Delta(i \mid \emptyset).$$

Submodularity with $S \supseteq \emptyset$ yields $\Delta(i \mid \emptyset) \ge \Delta(i \mid S)$, hence

$$s_i^{\mathrm{LOO}} \;\ge\; (1 - \kappa)\, \Delta(i \mid \emptyset) \;\ge\; (1 - \kappa)\, \Delta(i \mid S).$$

This proves equation 17.

Shapley bounds. By definition equation 8, φ_i is a weighted average of $\Delta(i \mid T)$ over $T \subseteq [n] \setminus \{i\}$, where all weights are nonnegative and sum to 1.

Step 1: sandwich φ_i between extreme marginals. For every $T \subseteq [n] \setminus \{i\}$, submodularity implies

$$\Delta(i \mid \emptyset) \;\ge\; \Delta(i \mid T) \;\ge\; \Delta(i \mid [n] \setminus \{i\}).$$

A convex combination preserves the interval, so

$$\Delta(i \mid [n] \setminus \{i\}) \;\le\; \varphi_i \;\le\; \Delta(i \mid \emptyset). \tag{40}$$

Step 2: link extremes by curvature. Curvature gives

$$\Delta(i \mid [n] \setminus \{i\}) \;\ge\; (1 - \kappa)\, \Delta(i \mid \emptyset). \tag{41}$$

Lower bound in equation 18. From equation 40, $\varphi_i \ge \Delta(i \mid [n] \setminus \{i\})$. Combine with equation 41 and $\Delta(i \mid \emptyset) \ge \Delta(i \mid S)$:

$$\varphi_i \;\ge\; \Delta(i \mid [n] \setminus \{i\}) \;\ge\; (1 - \kappa)\, \Delta(i \mid \emptyset) \;\ge\; (1 - \kappa)\, \Delta(i \mid S).$$

Upper bound in equation 18. From equation 40, $\varphi_i \le \Delta(i \mid \emptyset)$. Also, submodularity implies $\Delta(i \mid S) \ge \Delta(i \mid [n] \setminus \{i\})$. With equation 41, we have

$$\Delta(i \mid S) \;\ge\; \Delta(i \mid [n] \setminus \{i\}) \;\ge\; (1 - \kappa)\, \Delta(i \mid \emptyset),$$

so $\Delta(i \mid \emptyset) \le \frac{1}{1-\kappa} \Delta(i \mid S)$ and therefore

$$\varphi_i \;\le\; \Delta(i \mid \emptyset) \;\le\; \frac{1}{1 - \kappa}\, \Delta(i \mid S).$$

This proves equation 18. □

D.2 Proof of Theorem 4.3

Proof. We prove the two guarantees by the same template: show that the greedy step induced by the chosen score is an approximate greedy step in terms of true marginals, then apply the standard monotone submodular recursion.

Step 1: a standard marginal-to-gap inequality. Fix any set $S \subseteq [n]$ with $|S| \le k-1$ and let $O^\star$ attain OPT_k with $|O^\star| \le k$. Monotonicity and submodularity imply

$$\mathrm{OPT}_k - v(S) = v(O^\star) - v(S) \le v(S \cup O^\star) - v(S) \le \sum_{i \in O^\star \setminus S} \Delta(i \mid S).$$

Since $|O^\star \setminus S| \le k$, there exists $i \in O^\star \setminus S$ with

$$\max_{j \notin S} \Delta(j \mid S) \;\ge\; \frac{1}{k}\big(\mathrm{OPT}_k - v(S)\big). \tag{42}$$

Step 2: LOO-top-k. View S_k^{LOO} as a sequential rule: let $S_0 = \emptyset$ and for $t = 0, \dots, k-1$ choose

$$i_{t+1} \in \arg\max_{i \notin S_t} s_i^{\mathrm{LOO}}, \qquad S_{t+1} = S_t \cup \{i_{t+1}\}.$$

Write $F_t \triangleq v(S_t)$. Let $j_t \in \arg\max_{i \notin S_t} \Delta(i \mid S_t)$. Lemma 4.2 gives, for all $i \notin S_t$, $s_i^{\mathrm{LOO}} \ge (1-\kappa)\Delta(i \mid S_t)$ and $\Delta(i \mid S_t) \ge s_i^{\mathrm{LOO}}$. Thus

$$\Delta(i_{t+1} \mid S_t) \;\ge\; s_{i_{t+1}}^{\mathrm{LOO}} \;\ge\; s_{j_t}^{\mathrm{LOO}} \;\ge\; (1-\kappa)\Delta(j_t \mid S_t) = (1-\kappa) \max_{i \notin S_t} \Delta(i \mid S_t).$$

Combine with equation 42:

$$F_{t+1} - F_t = \Delta(i_{t+1} \mid S_t) \;\ge\; \frac{1-\kappa}{k}\big(\mathrm{OPT}_k - F_t\big).$$

Rearrange:

$$\mathrm{OPT}_k - F_{t+1} \le \left(1 - \frac{1-\kappa}{k}\right)\big(\mathrm{OPT}_k - F_t\big).$$

Iterate for $t = 0, \dots, k-1$ and use $(1 - a/k)^k \le e^{-a}$:

$$\mathrm{OPT}_k - F_k \le e^{-(1-\kappa)}\mathrm{OPT}_k, \qquad F_k = v(S_k^{\mathrm{LOO}}) \ge \big(1 - e^{-(1-\kappa)}\big)\mathrm{OPT}_k.$$

Step 3: Shapley-top-k. Repeat the same argument with Shapley scores. Let $S_0 = \emptyset$ and for $t = 0, \dots, k-1$ choose

$$i_{t+1} \in \arg\max_{i \notin S_t} \varphi_i, \qquad S_{t+1} = S_t \cup \{i_{t+1}\}.$$

Lemma 4.2 gives, for all $i \notin S_t$, $\varphi_i \ge (1-\kappa)\Delta(i \mid S_t)$ and $\Delta(i \mid S_t) \ge (1-\kappa)\varphi_i$. Thus, with j_t as before,

$$\Delta(i_{t+1} \mid S_t) \;\ge\; (1-\kappa)\varphi_{i_{t+1}} = (1-\kappa) \max_{i \notin S_t} \varphi_i \;\ge\; (1-\kappa)^2 \max_{i \notin S_t} \Delta(i \mid S_t).$$

Combine with equation 42:

$$F_{t+1} - F_t \ge \frac{(1-\kappa)^2}{k}\big(\mathrm{OPT}_k - F_t\big).$$

Unroll:

$$\mathrm{OPT}_k - F_k \le \left(1 - \frac{(1-\kappa)^2}{k}\right)^k \mathrm{OPT}_k \le e^{-(1-\kappa)^2}\mathrm{OPT}_k,$$

so

$$v(S_k^{\mathrm{Sh}}) = F_k \ge \big(1 - e^{-(1-\kappa)^2}\big)\mathrm{OPT}_k.$$

This proves equation 19 and equation 20. □

E Details for Section 5.1

E.1 Threshold classification with a midpoint learner

Model. We consider one-dimensional binary classification with true threshold $t^\star = \frac{1}{2}$ and labels

$$y(x) = \mathbf{1}\{x > t^\star\}, \qquad x \in [0,1].$$

We evaluate accuracy under the uniform test distribution:

$$f(S) \;:=\; \Pr_{x\sim \mathrm{Unif}[0,1]}\big[h_S(x) = y(x)\big].$$

Notation. Given a finite labeled training set $S \subset [0,1]$, write

$$S^- := \{x \in S :\; x < t^\star\}, \qquad S^+ := \{x \in S :\; x > t^\star\}.$$

When both sides are present we set

$$L(S) := \max S^- \quad \text{(well-defined if } S^- \neq \emptyset), \qquad R(S) := \min S^+ \quad \text{(well-defined if } S^+ \neq \emptyset),$$

and define the midpoint $\widehat{t}(S) := \frac{1}{2}\big(L(S) + R(S)\big)$.

Learner (piecewise definition).

$$h_S(x) \;:=\; \begin{cases} \mathbf{1}\{x > \widehat{t}(S)\}, & \text{if } S^- \neq \emptyset \text{ and } S^+ \neq \emptyset, \\ 0, & \text{if } S^- \neq \emptyset \text{ and } S^+ = \emptyset, \\ 1, & \text{if } S^- = \emptyset \text{ and } S^+ \neq \emptyset, \\ 0, & \text{if } S = \emptyset. \end{cases}$$

Thus, if one side is missing, the learner predicts the observed side everywhere (a constant classifier).

Accuracy formulas. If S^- and S^+ are both nonempty, then

$$f(S) \;=\; 1 - \big|\widehat{t}(S) - t^\star\big| \;=\; 1 - \tfrac{1}{2}\,\big|L(S) + R(S) - 1\big|. \tag{43}$$

If S contains points from only one side, h_S is constant and

$$f(S) \;=\; \tfrac{1}{2}. \tag{44}$$

Marginal gains. For $i \notin S$, define $\Delta(i \mid S) := f(S \cup \{i\}) - f(S)$.

Lemma E.1 (Opposite side yields a jump, same side yields none)**.** *Let $S \subset [0,1]$ be finite.*

1. *If S has only negatives ($S^- \neq \emptyset$, $S^+ = \emptyset$), then $f(S) = \frac{1}{2}$ and:*
   - *for any negative $i < t^\star$, $\Delta(i \mid S) = 0$;*
   - *for any positive $j > t^\star$,*

   $$\Delta(j \mid S) = \Big(1 - \tfrac{1}{2}\,|L(S) + j - 1|\Big) - \tfrac{1}{2} \;\geq\; \tfrac{1}{4}.$$

   The case "only positives" is symmetric.
2. *If S has both sides, then for any $i \in [0,1]$,*

   $$0 \;\leq\; \Delta(i \mid S) \;\leq\; \tfrac{1}{4}.$$

Proof. If S has only negatives then $h_S \equiv 0$, hence equation 44 gives $f(S) = \frac{1}{2}$. Adding a negative keeps the classifier constant, so $\Delta(i \mid S) = 0$.

Adding a positive $j > t^\star$ brackets the threshold, so equation 43 gives

$$f(S \cup \{j\}) = 1 - \tfrac{1}{2}\,|L(S) + j - 1|.$$

Therefore

$$\Delta(j \mid S) = f(S \cup \{j\}) - f(S) = \tfrac{1}{2} - \tfrac{1}{2}\,|L(S) + j - 1| = \tfrac{1}{2}\big(1 - |L(S) + j - 1|\big).$$

Since $L(S) \in [0, 1/2)$ and $j \in (1/2, 1]$, the sum $L(S)+j$ lies in $(1/2, 3/2)$, hence $|L(S)+j-1| \le 1/2$, which yields $\Delta(j \mid S) \ge \frac{1}{4}$.

If S has both sides, set $L := L(S)$ and $R := R(S)$. If $i < t^\star$, define $L' = \max\{L, i\} \in [L, 1/2)$ and use the reverse triangle inequality:

$$\Delta(i \mid S) = \tfrac{1}{2}\Big(|L + R - 1| - |L' + R - 1|\Big) \le \tfrac{1}{2}\,|L - L'| \le \tfrac{1}{2}\Big(\tfrac{1}{2} - L\Big) \le \tfrac{1}{4}.$$

The case $i > t^\star$ is symmetric with $R' = \min\{R, i\}$. Nonnegativity follows from the monotonic improvement of the bracketing interval. □

Submodularity defect and weak submodularity constant. Recall

$$\sigma^\star(f) := \sup_{A \subseteq B,\, i \notin B} \big[\Delta(i \mid B) - \Delta(i \mid A)\big]_+, \qquad \alpha(f) := \sup\Big\{\alpha \in [0,1] : \Delta(i \mid A) \ge \alpha\, \Delta(i \mid B)\ \forall A \subseteq B,\, i \notin B\Big\}.$$

Proposition E.2 (Proof of Theorem 5.3)**.** *For the midpoint threshold learner above,*

$$\sigma^\star(f) \;=\; \tfrac{1}{4}, \qquad \alpha(f) \;=\; 0.$$

Proof. Upper bound $\sigma^\star(f) \le \frac{1}{4}$. Fix $A \subseteq B$ and $i \notin B$.

If B has both sides, Lemma E.1 gives $\Delta(i \mid B) \le \frac{1}{4}$ and $\Delta(i \mid A) \ge 0$, so the defect is at most $\frac{1}{4}$.

If B is one-sided and i lies on the same side, then $\Delta(i \mid B) = 0$ so the defect is 0. If B is one-sided and i lies on the opposite side, Lemma E.1 shows $\Delta(i \mid S) = \frac{1}{2}(L(S) + i) - \frac{1}{2}$ for one-sided S on that side. Since $A \subseteq B$ implies $L(B) \ge L(A)$, we get

$$\Delta(i \mid B) - \Delta(i \mid A) = \tfrac{1}{2}(L(B) - L(A)) \le \tfrac{1}{4}.$$

Thus $\sigma^\star(f) \le \frac{1}{4}$.

Lower bound $\sigma^\star(f) \ge \frac{1}{4}$. Fix $\varepsilon \in (0, 1/4)$ and take $x_L = \frac{1}{2} - \varepsilon$, $x_R = 1$, and $i = \frac{1}{2} + \varepsilon$. Let $A = \{x_R\}$ (only positives) and $B = A \cup \{x_L\}$ (both sides). Then $\Delta(i \mid A) = 0$ (the classifier stays constant), and a direct bracket calculation yields

$$\Delta(i \mid B) = \tfrac{1}{2}\Big(|x_L + x_R - 1| - |x_L + i - 1|\Big) = \tfrac{1}{4} - \tfrac{\varepsilon}{2}.$$

Letting $\varepsilon \downarrow 0$ gives $\sigma^\star(f) \ge \frac{1}{4}$.

$\alpha(f) = 0$. Take A one-sided and pick i on the same side. Then $\Delta(i \mid A) = 0$. Now pick $B \supset A$ that includes one opposite-side point so that B brackets. Choose i so that it tightens the bracket; Lemma E.1 gives $\Delta(i \mid B) > 0$. Hence $\Delta(i \mid A) \ge \alpha\, \Delta(i \mid B)$ fails for every $\alpha > 0$, so $\alpha(f) = 0$. □

E.2 A MINIMAL TWEAK OF CAPPED COVERAGE: THRESHOLDED ACTIVATION

Let the ground set partition as $G_1, \ldots, G_m$ with weights $w_j > 0$ and define V_r as in equation 21.

Proposition E.3 (Proof of Theorem 5.4)**.** *Fix $r \ge 2$. The function V_r is monotone but is neither submodular nor supermodular.*

Proof. Monotonicity holds since $|U \cap G_j|$ only increases when we add elements.

To violate submodularity, fix a group G with weight w_G. Let $A = \emptyset$, let B contain $r-1$ points from G, and pick $x \in G \setminus B$. Then $|A \cap G| = 0$ and $|(A \cup \{x\}) \cap G| = 1 < r$, so $\Delta(x \mid A) = 0$. Also $|B \cap G| = r - 1$ and $|(B \cup \{x\}) \cap G| = r$, so $\Delta(x \mid B) = w_G$. Thus $\Delta(x \mid A) < \Delta(x \mid B)$.

To violate supermodularity, let A' contain $r - 1$ points from G and let B' contain r points from G. Pick $x' \in G \setminus B'$. Then $\Delta(x' \mid A') = w_G$ and $\Delta(x' \mid B') = 0$, so $\Delta(x' \mid A') > \Delta(x' \mid B')$. □

Remark E.4 (Learning interpretation)**.** *The activation boundary $|U \cap G_j| = r - 1$ plays the same role as a decision boundary in the threshold learner: a few complementary points must appear before the model becomes locally correct on that archetype. After activation, additional points become substitutes and yield no further gain.*

E.3 A STRONGER LOWER BOUND: HETEROGENEOUS THRESHOLDS DESTROY ADAPTIVITY

We prove Theorem 5.5. The construction uses the same thresholded objective as in Section E.2, but we randomize the thresholds across groups.

Instance distribution. Fix $k \geq 2$ and a parameter $r_0 \in \{2, \ldots, k\}$. We have k groups $G_1, \ldots, G_k$, each with at least $n \geq r_0$ available duplicates (so supply never constrains the policy). Independently for each group j:

- with probability $1/r_0$, group j is *good* and has threshold $r_j = r_0$;
- with probability $1 - 1/r_0$, group j is *bad* and has threshold $r_j > k$.

Define the value of a selected set U by

$$V(U) \;=\; \frac{1}{k}\,\big|\{j :\, |U \cap G_j| \geq r_j\}\big|.$$

A bad group never contributes under budget k, while a good group contributes exactly when the policy places at least r_0 points in that group.

Lemma E.5 (Expected optimum)**.** *Let $G \sim \mathrm{Bin}(k, 1/r_0)$ be the number of good groups. Then the omniscient optimum completes $\min\{G, \lfloor k/r_0 \rfloor\}$ groups and satisfies*

$$\mathbb{E}[\mathrm{OPT}] \;=\; \frac{1}{k}\,\mathbb{E}\Big[\min\Big\{G,\; \Big\lfloor \frac{k}{r_0} \Big\rfloor\Big\}\Big] \;\geq\; \frac{1}{2r_0}\Big(1 - e^{-k/(8r_0)}\Big).$$

Proof. Write $\mu := \mathbb{E}[G] = k/r_0$. For $\delta = 1/2$, the multiplicative Chernoff bound gives

$$\Pr\Big(G \leq \frac{\mu}{2}\Big) \;\leq\; e^{-\mu/8} \;=\; e^{-k/(8r_0)}.$$

Since $\min\{G, \lfloor k/r_0 \rfloor\} \geq \min\{G, \mu\}$ and on the event $\{G \geq \mu/2\}$ we have $\min\{G, \mu\} \geq \mu/2$, we obtain

$$\mathbb{E}\Big[\min\Big\{G,\; \Big\lfloor \frac{k}{r_0} \Big\rfloor\Big\}\Big] \;\geq\; \mathbb{E}[\min\{G, \mu\}] \;\geq\; \frac{\mu}{2}\,\Pr\Big(G \geq \frac{\mu}{2}\Big) \;\geq\; \frac{\mu}{2}\Big(1 - e^{-\mu/8}\Big).$$

Multiplying by $1/k$ and substituting $\mu = k/r_0$ proves the claim. □

Lemma E.6 (Expected value of any policy)**.** *For any (possibly randomized and adaptive) policy $\mathcal{A}$ that outputs $U_{\mathcal{A}}$ with $|U_{\mathcal{A}}| \leq k$,*

$$\mathbb{E}[V(U_{\mathcal{A}})] \;\leq\; \frac{1}{r_0^2}.$$

Proof. A policy completes a group only if it allocates at least r_0 points to that group. Let T be the (random) number of distinct groups to which $\mathcal{A}$ allocates at least r_0 points. The budget constraint gives $T \leq k/r_0$.

Group labels (good versus bad) are independent of the policy and remain hidden until the policy invests r_0 points in that group, because bad groups never activate under budget k. Thus each of the T tested groups is good with probability $1/r_0$, independently of the policy history. Therefore, the expected number of completed groups is at most $\mathbb{E}[T]/r_0 \leq (k/r_0)/r_0 = k/r_0^2$. Dividing by k gives $\mathbb{E}[V(U_{\mathcal{A}})] \leq 1/r_0^2$. □

Proof of Theorem 5.5. Combine Theorem E.6 and Theorem E.5:

$$\frac{\mathbb{E}[V(U_{\mathcal{A}})]}{\mathbb{E}[\mathrm{OPT}]} \;\leq\; \frac{(1/r_0^2)}{(1/(2r_0))(1 - e^{-k/(8r_0)})} \;=\; \frac{2}{r_0\,(1 - e^{-k/(8r_0)})}.$$

If $k \geq 8r_0 \ln 2$ then $e^{-k/(8r_0)} \leq 1/2$, so the right-hand side is at most $4/r_0$.

The inequality above holds for every deterministic policy under the random instance distribution. By Yao's minimax principle, the same bound applies to every randomized policy in the worst case. □

Remark E.7 (Learning interpretation)**.** *This construction fits the same "boundary" motif as the threshold learner and the thresholded-coverage objective: each group behaves like a local region that starts paying off only after the learner sees r_j nearby examples. Heterogeneity in (r_j) forces the policy to spend substantial budget before it can even identify which regions are feasible, which yields the information-theoretic barrier above.*

F Details for Section 5.2

F.1 Midpoint learner, coverage event, and a deterministic defect bound

We use the threshold task on $[0,1]$ with $t^\star = \frac{1}{2}$ and labels $y(x) = \mathbf{1}\{x > t^\star\}$, and we evaluate accuracy under $\mathrm{Unif}[0,1]$:

$$f(S) := \Pr_{x\sim \mathrm{Unif}[0,1]}[h_S(x) = y(x)].$$

Given a finite $S \subset [0,1]$, define the two sides

$$S^- := \{x \in S :\ x < t^\star\}, \qquad S^+ := \{x \in S :\ x > t^\star\}.$$

We say that S *straddles* $t^\star$ if $S^- \neq \emptyset$ and $S^+ \neq \emptyset$. When S straddles, define

$$L(S) := \max S^-, \qquad R(S) := \min S^+, \qquad \widehat{t}(S) := \tfrac{1}{2}\big(L(S)+R(S)\big), \qquad \varepsilon_{\mathrm{br}}(S) := R(S)-L(S).$$

The midpoint learner predicts $h_S(x) = \mathbf{1}\{x > \widehat{t}(S)\}$ when S straddles, and predicts the observed label everywhere (a constant classifier) when S is one-sided.

Accuracy on straddling sets. If S straddles $t^\star$, then the classifier makes errors on an interval of length $|\widehat{t}(S) - t^\star|$, hence

$$f(S) = 1 - \big|\widehat{t}(S) - \tfrac{1}{2}\big|. \tag{45}$$

Moreover, since $L(S) \le \frac{1}{2} \le R(S)$, we have

$$\big|\widehat{t}(S) - \tfrac{1}{2}\big| = \tfrac{1}{2}\,|L(S) + R(S) - 1| \le \tfrac{1}{2}\,(R(S) - L(S)) = \tfrac{1}{2}\,\varepsilon_{\mathrm{br}}(S). \tag{46}$$

Proposition F.1 (Coverage implies approximate submodularity above the core)**.** *If a core set $B \subset [0,1]$ straddles $\frac{1}{2}$, then the midpoint learner satisfies*

$$\sigma^\star(f; B) \ \le\ \varepsilon_{\mathrm{br}}(B).$$

Proof. Fix a straddling core B. Consider any A, B', i with $B \subseteq A \subseteq B'$ and $i \notin B'$. If $\Delta(i \mid B') \le \Delta(i \mid A)$ then $[\Delta(i \mid B') - \Delta(i \mid A)]_+ = 0$. Otherwise,

$$[\Delta(i \mid B') - \Delta(i \mid A)]_+ = \Delta(i \mid B') - \Delta(i \mid A) \le \Delta(i \mid B').$$

Since B straddles, every superset of B straddles, so both B' and $B' \cup \{i\}$ straddle. Using equation 45,

$$\Delta(i \mid B') = f(B' \cup \{i\}) - f(B') = \big|\widehat{t}(B') - \tfrac{1}{2}\big| - \big|\widehat{t}(B' \cup \{i\}) - \tfrac{1}{2}\big| \le \big|\widehat{t}(B') - \tfrac{1}{2}\big|.$$

Apply equation 46 to get

$$\Delta(i \mid B') \le \tfrac{1}{2}\,\varepsilon_{\mathrm{br}}(B').$$

Adding points can only tighten the bracket, so $L(B') \ge L(B)$ and $R(B') \le R(B)$, hence $\varepsilon_{\mathrm{br}}(B') \le \varepsilon_{\mathrm{br}}(B)$. Therefore,

$$[\Delta(i \mid B') - \Delta(i \mid A)]_+ \le \Delta(i \mid B') \le \tfrac{1}{2}\,\varepsilon_{\mathrm{br}}(B') \le \tfrac{1}{2}\,\varepsilon_{\mathrm{br}}(B) \le \varepsilon_{\mathrm{br}}(B).$$

Taking the supremum over $A \subseteq B'$ and $i \notin B'$ yields $\sigma^\star(f; B) \le \varepsilon_{\mathrm{br}}(B)$. □

F.2 Order-statistic bounds for i.i.d. coverage

Let $S_n = \{X_1, \dots, X_n\}$ with $X_j \overset{\text{iid}}{\sim} \text{Unif}[0,1]$. Let Str_n denote the event that S_n straddles $\frac{1}{2}$.

Define $K_n := |\{j : X_j < \frac{1}{2}\}| \sim \text{Bin}(n, \frac{1}{2})$. We will also use the conventions

$$L_n := \max\{X_j : X_j < \tfrac{1}{2}\} \quad \text{(or 0 if empty)}, \qquad R_n := \min\{X_j : X_j > \tfrac{1}{2}\} \quad \text{(or 1 if empty)},$$

and gaps

$$\varepsilon_L(S_n) := \tfrac{1}{2} - L_n, \qquad \varepsilon_R(S_n) := R_n - \tfrac{1}{2}.$$

On Str_n these coincide with the bracketing gaps and satisfy $\varepsilon_{\text{br}}(S_n) = \varepsilon_L(S_n) + \varepsilon_R(S_n)$.

Lemma F.2 (Straddling probability and expected one-sided gaps)**.** *We have*

$$\Pr(\neg\mathsf{Str}_n) = 2^{1-n}, \qquad \mathbb{E}\big[\varepsilon_L(S_n)\big] = \frac{1-2^{-(n+1)}}{n+1} \le \frac{1}{n+1}, \qquad \mathbb{E}\big[\varepsilon_R(S_n)\big] = \frac{1-2^{-(n+1)}}{n+1} \le \frac{1}{n+1}.$$

Proof. The event $\neg\mathsf{Str}_n$ means $K_n \in \{0, n\}$, so

$$\Pr(\neg\mathsf{Str}_n) = \Pr(K_n = 0) + \Pr(K_n = n) = 2 \cdot 2^{-n} = 2^{1-n}.$$

For the left gap, condition on $K_n = k$. Given $K_n = k$, the k left samples are i.i.d. $\text{Unif}[0, \frac{1}{2}]$. If $k = 0$, our convention gives $L_n = 0$, so $\varepsilon_L(S_n) = \frac{1}{2} = \frac{1/2}{k+1}$. If $k \ge 1$, the maximum $U_{(k)}$ of k i.i.d. $\text{Unif}[0, \frac{1}{2}]$ samples satisfies $\mathbb{E}[\frac{1}{2} - U_{(k)}] = \frac{1/2}{k+1}$. Hence

$$\mathbb{E}\big[\varepsilon_L(S_n)\big] = \sum_{k=0}^{n} \frac{1}{2(k+1)} \binom{n}{k} 2^{-n} = \frac{1}{2}\,\mathbb{E}\left[\frac{1}{K_n+1}\right].$$

Use $\frac{1}{k+1} = \int_0^1 t^k\,dt$ to compute

$$\mathbb{E}\left[\frac{1}{K_n+1}\right] = \sum_{k=0}^{n} \binom{n}{k} 2^{-n} \int_0^1 t^k\,dt = \int_0^1 \left(\frac{1+t}{2}\right)^n dt = \frac{2}{n+1}\left(1 - 2^{-(n+1)}\right).$$

Therefore

$$\mathbb{E}\big[\varepsilon_L(S_n)\big] = \frac{1-2^{-(n+1)}}{n+1} \le \frac{1}{n+1}.$$

Symmetry yields the same expression for $\mathbb{E}[\varepsilon_R(S_n)]$. □

F.3 Proof of Theorem 5.7

Proof of Theorem 5.7. Let $D_n := \sigma^\star(f; S_n)$. Decompose on Str_n:

$$\mathbb{E}[D_n] = \mathbb{E}\big[D_n \mathbf{1}\{\neg\mathsf{Str}_n\}\big] + \mathbb{E}\big[D_n \mathbf{1}\{\mathsf{Str}_n\}\big].$$

On $\neg\mathsf{Str}_n$, we use the domination $D_n \le \sigma^\star(f)$ and the global defect $\sigma^\star(f) = \frac{1}{4}$ from the threshold example in the previous subsection. Thus

$$\mathbb{E}\big[D_n \mathbf{1}\{\neg\mathsf{Str}_n\}\big] \le \tfrac{1}{4}\,\Pr(\neg\mathsf{Str}_n) = \tfrac{1}{4} \cdot 2^{1-n},$$

where the last equality uses Theorem F.2.

On Str_n, Theorem F.1 yields $D_n \le \varepsilon_{\text{br}}(S_n)$, hence

$$\mathbb{E}\big[D_n \mathbf{1}\{\mathsf{Str}_n\}\big] \le \mathbb{E}\big[\varepsilon_{\text{br}}(S_n)\big] \le \mathbb{E}\big[\varepsilon_L(S_n)\big] + \mathbb{E}\big[\varepsilon_R(S_n)\big] \le \frac{2}{n+1},$$

where the last step uses Theorem F.2.

Summing the two bounds gives

$$\mathbb{E}[D_n] \le \tfrac{1}{4} \cdot 2^{1-n} + \frac{2}{n+1}.$$

□

G More Details on Experiments

G.1 Empirical estimator for the core-relative defect

For a fixed pool U_n and a learner-induced value function $f(\cdot)$, we estimate the core-relative defect by Monte Carlo sampling of nested triples:

1. Sample $B \subseteq U_n$ by choosing $|B| = b$ uniformly from an interval $[b_{\min}, \lfloor 0.6n \rfloor]$ and then sampling b points uniformly without replacement.
2. Sample $A \subset B$ by choosing $|A| = a$ uniformly from $\{1, \dots, b-1\}$ and then sampling uniformly without replacement from B.
3. Sample $i \in U_n \setminus B$ uniformly.
4. Compute the violation $v(A, B, i) := [\Delta(i \mid B) - \Delta(i \mid A)]_+$.

Given M such triples, define

$$\widehat{\sigma}(U_n) \;:=\; \max_{m \in [M]} v(A_m, B_m, i_m).$$

For each n we repeat the full procedure over R independent pools U_n and report the mean $\frac{1}{R}\sum_{r=1}^{R} \widehat{\sigma}(U_n^{(r)})$ with a normal-approximation 95% confidence interval $1.96 \cdot \widehat{\mathrm{sd}}/\sqrt{R}$ computed across the R replicates.

G.2 a9a: logistic regression protocol

Data. We use the `a9a` dataset from LIBSVM with the provided train and test split.

Learner and value function. Given a labeled subset $S \subseteq U_n$, we train ℓ_2-regularized logistic regression (from scratch) and define $f(S)$ as the test accuracy. If $S = \emptyset$, we predict the majority label on the test set. If S contains only one class, we predict that class everywhere and return the corresponding test accuracy.

Hyperparameters and preprocessing. We standardize features with a sparse-aware scaler (no mean centering) and train `LogisticRegression` with `solver=liblinear`, `C=1.0`, and `max_iter=1000`. We cache $f(S)$ values across the sampled subsets inside a fixed pool U_n to avoid redundant retraining.

Defect sampling. For each n we sample R independent pools U_n uniformly without replacement from the training set. Inside each pool we draw M triples (A, B, i) by the procedure above and compute $\widehat{\sigma}(U_n)$. The script also records an auxiliary "mean violation" statistic (the average of $v(A, B, i)$ across the M triples), but the plots in Figure 1 use the maximum over triples.

G.3 MNIST (digits 3 vs. 5): small CNN protocol

Data. We use MNIST with standard train and test sets, filter to the two digits $\{3, 5\}$, and map the labels to $\{0, 1\}$. We apply the standard normalization used in the script (tensor conversion and per-pixel normalization).

Learner and value function. Given $S \subseteq U_n$, we train a small CNN from scratch and set $f(S)$ to its test accuracy. If $S = \emptyset$, we use the test-set majority-class baseline. If S contains a single class, we use the corresponding constant predictor.

Architecture and optimization. The CNN uses two 3×3 convolution layers (32 and 64 channels), ReLU, 2×2 max pooling, one hidden fully connected layer of width 64, and a single-logit output. We train with AdamW for 3 epochs, batch size 256, learning rate 10^{-3}, and weight decay 10^{-3}. We evaluate accuracy with a fixed threshold at logit 0.

Defect sampling and reporting. We repeat the same (U_n, M, R) protocol as above, with caching of trained-subset evaluations inside each pool. We report the mean $\widehat{\sigma}(U_n)$ over R pools with 95% confidence intervals.

G.4 Reproducibility notes

Each replicate fixes a random seed for: (i) pool sampling, (ii) triple sampling, and (iii) training initialization. GPU kernels can still introduce small nondeterminism, but the plotted error bars reflect the replicate-to-replicate variability.

H Proof of Theorem 6.1

Fix integers $k \geq 2$ and $B \geq 2$ and set $n := kB$. Let

$$V := \{(i, a) : i \in [k],\ a \in [B]\}, \qquad |V| = n.$$

Nature draws hidden indices $J_1, \ldots, J_k$ independently and uniformly from $[B]$.

Passing a layer and the valuation. For $S \subseteq V$, say that S *passes layer* i if

$$(i, J_i) \in S \quad \text{and} \quad (i, a) \notin S \ \text{ for all } a \neq J_i.$$

Define

$$v(S) := \max\{t \in \{0, 1, \ldots, k\} :\ S \text{ passes layers } 1, 2, \ldots, t\}.$$

All probabilities and expectations below are taken with respect to the random draw of $(J_1, \ldots, J_k)$ unless stated otherwise.

H.1 Greedy achieves the optimum with at most nk value queries

Proposition H.1 (Greedy succeeds). *Let $U_0 = \emptyset$ and let greedy construct U_i from U_{i-1} by selecting an element $x \in V \setminus U_{i-1}$ maximizing the marginal gain*

$$\Delta(x \mid U_{i-1}) := v(U_{i-1} \cup \{x\}) - v(U_{i-1}),$$

breaking ties deterministically. Then greedy outputs U_k with $|U_k| = k$ and $v(U_k) = k$ with probability 1. Moreover, greedy uses at most nk value queries.

Proof. We prove by induction on $i \in \{0, 1, \ldots, k\}$ that U_i passes layers $1, \ldots, i$.

For $i = 0$, the claim is vacuous. Assume the claim holds for $i - 1$. Then U_{i-1} passes layers $1, \ldots, i-1$, so $v(U_{i-1}) = i - 1$. Consider any candidate $x = (\ell, a) \notin U_{i-1}$.

Case 1: $\ell < i$. Then U_{i-1} already contains exactly one element in layer ℓ, namely (ℓ, J_ℓ). Adding x introduces a second element in layer ℓ, so $U_{i-1} \cup \{x\}$ fails layer ℓ. Hence it cannot pass all layers $1, \ldots, i-1$, so $v(U_{i-1} \cup \{x\}) \leq \ell - 1 \leq i - 2$ and $\Delta(x \mid U_{i-1}) \leq -1$.

Case 2: $\ell = i$. If $a = J_i$, then $U_{i-1} \cup \{x\}$ passes layer i in addition to layers $1, \ldots, i-1$, so $v(U_{i-1} \cup \{x\}) \geq i$ and thus $\Delta(x \mid U_{i-1}) \geq 1$. If $a \neq J_i$, then $U_{i-1} \cup \{x\}$ fails layer i (it includes the wrong element and excludes the correct one), so it passes exactly layers $1, \ldots, i-1$ and has value $i - 1$, hence $\Delta(x \mid U_{i-1}) = 0$.

Case 3: $\ell > i$. Then adding x does not affect whether layers $1, \ldots, i$ pass, but $U_{i-1} \cup \{x\}$ still has no element from layer i, so it fails layer i and $v(U_{i-1} \cup \{x\}) = i - 1$, hence $\Delta(x \mid U_{i-1}) = 0$.

Therefore the unique element with strictly positive marginal gain is (i, J_i), whose marginal gain is at least 1, while every other element has marginal gain at most 0. Greedy selects (i, J_i) and thus U_i passes layers $1, \ldots, i$.

After k rounds, U_k passes all k layers, so $v(U_k) = k$.

For query complexity, in each round greedy evaluates $\Delta(x \mid U_{i-1})$ for at most n candidates, and each marginal evaluation uses two value queries (for $v(U_{i-1} \cup \{x\})$ and $v(U_{i-1})$). Since $v(U_{i-1})$ can be cached, one can implement each round with at most n oracle calls to $v(U_{i-1} \cup \{x\})$. In either accounting, greedy uses at most nk value queries up to a factor 2, and the statement nk is valid. □

H.2 ONE-ROUND UPPER BOUND

One-round model. A deterministic one-round algorithm chooses queries $S^{(1)}, \dots, S^{(Q)} \subseteq V$ in advance, observes the transcript

$$\mathcal{T} := (v(S^{(1)}), \dots, v(S^{(Q)})),$$

and outputs $U = U(\mathcal{T})$ with $|U| \le k$.

Define tail probabilities

$$p_i := \Pr[v(U) \ge i], \qquad i \in \{0, 1, \dots, k\}.$$

Since $v(U) \in \{0, 1, \dots, k\}$,

$$\mathbb{E}[v(U)] = \sum_{i=1}^{k} \Pr[v(U) \ge i] = \sum_{i=1}^{k} p_i. \tag{47}$$

Lemma H.2 (Fixed-query prefix probability)**.** *For any fixed set $S \subseteq V$ and any $t \in \{1, \dots, k\}$,*

$$\Pr\big[v(S) \ge t\big] \le B^{-t}.$$

Proof. The event $\{v(S) \ge t\}$ implies that S passes each layer $i \in \{1, \dots, t\}$. If for some $i \le t$ the intersection $S \cap \{(i,1), \dots, (i,B)\}$ has size different from 1, then S fails layer i deterministically, so $\Pr[v(S) \ge t] = 0$. Otherwise, for each $i \le t$ there exists a unique $a_i \in [B]$ such that $S \cap \{(i,1), \dots, (i,B)\} = \{(i,a_i)\}$. Then S passes layer i if and only if $J_i = a_i$, which has probability $1/B$. Independence of $(J_1, \dots, J_t)$ yields

$$\Pr[v(S) \ge t] = \prod_{i=1}^{t} \Pr[J_i = a_i] = \left(\frac{1}{B}\right)^t = B^{-t}.$$

□

For each $i \in \{1, \dots, k\}$ define the event that some query reaches depth $i-1$:

$$E_{i-1} := \Big\{\exists q \in [Q] :\ v(S^{(q)}) \ge i - 1\Big\}.$$

Lemma H.3 (Union bound on $\Pr(E_{i-1})$)**.** *For every $i \in \{1, \dots, k\}$,*

$$\Pr[E_{i-1}] \le \frac{Q}{B^{i-1}}.$$

Proof. By Lemma H.2 with $t = i - 1$ and a union bound,

$$\Pr[E_{i-1}] \le \sum_{q=1}^{Q} \Pr\big[v(S^{(q)}) \ge i - 1\big] \le \sum_{q=1}^{Q} B^{-(i-1)} = \frac{Q}{B^{i-1}}.$$

□

Lemma H.4 (Transcript independence of J_i on $\neg E_{i-1}$)**.** *Fix $i \in \{1, \dots, k\}$. On the event $\neg E_{i-1}$, the transcript $\mathcal{T}$ is independent of J_i.*

Proof. Assume $\neg E_{i-1}$. Then for every query q we have $v(S^{(q)}) \le i - 2$. Fix q. The value $v(S^{(q)}) \le i - 2$ is determined by the first layer in $\{1, \dots, i-1\}$ that $S^{(q)}$ fails, which depends only on $(J_1, \dots, J_{i-1})$ and not on J_i. Thus, on $\neg E_{i-1}$, each coordinate $v(S^{(q)})$ is a function of $(J_1, \dots, J_{i-1})$ alone. Hence $\mathcal{T}$ is also a function of $(J_1, \dots, J_{i-1})$ alone on $\neg E_{i-1}$. Since J_i is independent of $(J_1, \dots, J_{i-1})$, the transcript is independent of J_i on $\neg E_{i-1}$. □

Lemma H.5 (Uniform guessing bound)**.** *Let J be uniform on $[B]$ and let W be any random variable taking values in $[B] \cup \{\perp\}$ that is independent of J. Then*

$$\Pr[W = J] \le \frac{1}{B}.$$

Proof. We compute

$$\Pr[W = J] = \sum_{a=1}^{B} \Pr(W = a,\ J = a) = \sum_{a=1}^{B} \Pr(W = a) \Pr(J = a \mid W = a).$$

Independence implies $\Pr(J = a \mid W = a) = \Pr(J = a) = 1/B$, so

$$\Pr[W = J] = \frac{1}{B} \sum_{a=1}^{B} \Pr(W = a) \leq \frac{1}{B}.$$

□

Lemma H.6 (Depth recursion)**.** *For every $i \in \{1, \ldots, k\}$,*

$$p_i \leq \Pr[E_{i-1}] + \frac{1}{B}\, p_{i-1}.$$

Proof. Split on E_{i-1}:

$$p_i = \Pr(v(U) \geq i) \leq \Pr(E_{i-1}) + \Pr\big(v(U) \geq i \ \wedge\ \neg E_{i-1}\big).$$

The event $\{v(U) \geq i\}$ implies $\{v(U) \geq i - 1\}$ and that U passes layer i.

Define $W_i(U) \in [B] \cup \{\perp\}$ as follows: if U contains exactly one element from layer i, namely (i, a), set $W_i(U) = a$; otherwise set $W_i(U) = \perp$. Then U passes layer i if and only if $W_i(U) = J_i$. Therefore,

$$\Pr\big(v(U) \geq i \ \wedge\ \neg E_{i-1}\big) \leq \Pr\big(v(U) \geq i - 1 \ \wedge\ W_i(U) = J_i \ \wedge\ \neg E_{i-1}\big).$$

Let $X := \{v(U) \geq i - 1\} \cap \neg E_{i-1}$. Then

$$\Pr\big(v(U) \geq i - 1 \ \wedge\ W_i(U) = J_i \ \wedge\ \neg E_{i-1}\big) = \Pr(W_i(U) = J_i \mid X) \Pr(X).$$

On $\neg E_{i-1}$, Lemma H.4 implies that $\mathcal{T}$ is independent of J_i. Since U is a deterministic function of $\mathcal{T}$, the random variable $W_i(U)$ is also a deterministic function of $\mathcal{T}$. Hence, conditional on $\neg E_{i-1}$, $W_i(U)$ is independent of J_i. The additional event X depends on $\mathcal{T}$ and $(J_1, \ldots, J_{i-1})$ but does not reveal J_i, so J_i remains uniform and independent of $W_i(U)$ under conditioning on X. Therefore Lemma H.5 applies under the conditional law given X, and yields

$$\Pr(W_i(U) = J_i \mid X) \leq \frac{1}{B}.$$

Thus,

$$\Pr\big(v(U) \geq i \ \wedge\ \neg E_{i-1}\big) \leq \frac{1}{B} \Pr(X) \leq \frac{1}{B} \Pr(v(U) \geq i - 1) = \frac{1}{B} p_{i-1}.$$

Combining gives $p_i \leq \Pr(E_{i-1}) + \frac{1}{B} p_{i-1}$. □

Lemma H.7 (Closed-form bound on p_i)**.** *For every $i \in \{1, \ldots, k\}$,*

$$p_i \leq \frac{iQ}{B^{i-1}} + \frac{1}{B^i}.$$

Proof. Combine Lemma H.3 with Lemma H.6:

$$p_i \leq \frac{Q}{B^{i-1}} + \frac{1}{B} p_{i-1}.$$

We prove by induction that $p_i \leq \frac{iQ}{B^{i-1}} + \frac{1}{B^i}$. For $i = 1$, $p_1 \leq 1$ and the right-hand side equals $Q + 1/B \geq 1$ for $Q \geq 1$. Assume it holds for $i - 1$. Then

$$p_i \leq \frac{Q}{B^{i-1}} + \frac{1}{B} \left(\frac{(i-1)Q}{B^{i-2}} + \frac{1}{B^{i-1}} \right) = \frac{iQ}{B^{i-1}} + \frac{1}{B^i}.$$

□

Theorem H.8 (Deterministic one-round bound)**.** *Every deterministic one-round algorithm that makes Q non-adaptive value queries and outputs U with $|U| \le k$ satisfies*

$$\mathbb{E}[v(U)] \;\le\; \sum_{i=1}^{k} \min\left\{1,\; \frac{iQ}{B^{i-1}} + \frac{1}{B^i}\right\}.$$

Proof. By equation 47, $\mathbb{E}[v(U)] = \sum_{i=1}^{k} p_i$. Lemma H.7 gives $p_i \le \frac{iQ}{B^{i-1}} + \frac{1}{B^i}$, and trivially $p_i \le 1$. Summing the minimum of these bounds proves the claim. □

Lemma H.9 (Yao minimax principle for one-round algorithms under a fixed instance distribution)**.** *Let $\mathcal{D}$ be the distribution over instances induced by the random draw of $(J_1, \dots, J_k)$. If every deterministic one-round algorithm has expected value at most β under $\mathcal{D}$, then every randomized one-round algorithm also has expected value at most β under $\mathcal{D}$.*

Proof. A randomized one-round algorithm is a probability distribution over deterministic one-round algorithms, induced by its internal randomness. Let $\mathcal{A}$ denote the randomized algorithm and let $\mathcal{A}_{\mathrm{det}}$ denote the random deterministic algorithm obtained by sampling the internal randomness. Then, by linearity of expectation,

$$\mathbb{E}_{I\sim\mathcal{D}}[v(\mathcal{A}(I))] = \mathbb{E}_{\mathcal{A}_{\mathrm{det}}}\left[\mathbb{E}_{I\sim\mathcal{D}}[v(\mathcal{A}_{\mathrm{det}}(I))]\right].$$

If each deterministic algorithm satisfies $\mathbb{E}_{I\sim\mathcal{D}}[v(\mathcal{A}_{\mathrm{det}}(I))] \le \beta$, then the outer expectation is also at most β. □

Theorem H.10 (Randomized one-round bound)**.** *Every (possibly randomized) one-round algorithm that makes Q non-adaptive value queries and outputs U with $|U| \le k$ satisfies*

$$\mathbb{E}[v(U)] \;\le\; \sum_{i=1}^{k} \min\left\{1,\; \frac{iQ}{B^{i-1}} + \frac{1}{B^i}\right\}.$$

Proof. Theorem H.8 bounds every deterministic one-round algorithm under the fixed distribution $\mathcal{D}$. Lemma H.9 transfers the bound to randomized one-round algorithms. □

H.3 A CONCRETE COROLLARY FOR $B = k$ AND $Q = nk$

Assume $B = k$. Then $n = kB = k^2$ and $Q = nk = k^3$.

Lemma H.11 (Explicit constant bound for $B = k$ and $Q = k^3$)**.** *For every $k \ge 6$, every one-round algorithm with $Q = k^3$ queries satisfies $\mathbb{E}[v(U)] \le 6$.*

Proof. By Theorem H.10,

$$\mathbb{E}[v(U)] \le \sum_{i=1}^{k} \min\left\{1,\; \frac{ik^3}{k^{i-1}} + \frac{1}{k^i}\right\} = \sum_{i=1}^{k} \min\left\{1,\; ik^{4-i} + k^{-i}\right\}.$$

For $i \in \{1, 2, 3, 4\}$ the minimum is at most 1, contributing at most 4. For $i \ge 5$ we upper bound by the second argument:

$$\sum_{i=5}^{k} \min\left\{1,\; ik^{4-i} + k^{-i}\right\} \le \sum_{i=5}^{\infty} \left(ik^{4-i} + k^{-i}\right).$$

Let $x := 1/k$. Then

$$\sum_{i=5}^{\infty} ik^{4-i} = \sum_{i=5}^{\infty} ix^{i-4} = \sum_{j=1}^{\infty} (j+4)x^j = \frac{x}{(1-x)^2} + \frac{4x}{1-x},$$

and

$$\sum_{i=5}^{\infty} k^{-i} = \sum_{i=5}^{\infty} x^i = \frac{x^5}{1-x}.$$

Therefore

$$\mathbb{E}[v(U)] \le 4 + \frac{x}{(1-x)^2} + \frac{4x}{1-x} + \frac{x^5}{1-x}.$$

For $k \ge 6$ we have $x \le 1/6$ and $1 - x \ge 5/6$, so

$$\frac{x}{(1-x)^2} \le \frac{1/6}{(5/6)^2} = \frac{6}{25}, \qquad \frac{4x}{1-x} \le \frac{4/6}{5/6} = \frac{4}{5}, \qquad \frac{x^5}{1-x} \le \frac{(1/6)^5}{5/6} = \frac{1}{5 \cdot 6^4}.$$

Hence

$$\mathbb{E}[v(U)] \le 4 + \frac{6}{25} + \frac{4}{5} + \frac{1}{5 \cdot 6^4} < 5.05 \le 6.$$

□

Completion of Theorem 6.1. Proposition H.1 proves Item 1 of Theorem 6.1. Theorem H.10 proves Item 2. Lemma H.11 gives the stated concrete specialization when $B = k$ and $Q = nk$.