HIGH INFLUENCE IS NOT NEW INFLUENCE: A NO-ARBITRAGE ANALYSIS OF MULTI-CAPABILITY DATA SELECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

When is one influence score per example enough to select data for several target capabilities? At rank one with a positive direction, fixed prices, task-wise max, round-robin, capability-level BIDS and greedy gap maximization select the same data, and near rank one a fixed linear price keeps a guaranteed share of every capability's best gain, tight up to a constant for two capabilities. On real payoffs, far from this regime, a fractional coverage ceiling bounds what one score loses. Beyond rank one, high influence is not new influence: to first order, a candidate is non-redundant only if no cost-matched mixture of selected data matches it on every capability. The replication program's dual prices capabilities by scarcity; with an inactive cap, the gap is the candidate's one-step, price-weighted advantage over every such mixture. On image benchmarks, the rules coincide on payoffs deformed to rank one; the one-step identity holds for small steps but stops predicting within 3% of the parameter distance training travels; and scalar influence falls below random sampling in all 32 targeted comparisons, a failure that first-order coverage flags, while gradient distance to the targets also ranks the other rules. Gradient matching is the most accurate rule that does not know the targets; our greedy rule is a probe, and we do not fine-tune language models.

1 INTRODUCTION

Influence-based data selection asks how much a training example would reduce a target loss and keeps the examples with the largest answer (Koh & Liang, 2017; Pruthi et al., 2020; Park et al., 2023); it is a standard way to choose instruction-tuning data (Xia et al., 2024; Min et al., 2026; Chen et al., 2026). Real targets have several capabilities, such as reasoning and multilingual question answering, or photographs and sketches of the same objects, and the common selection rules turn the vector of per-capability improvements of an example into one number: a mean, a maximum, a rotation over tasks, or a normalization followed by a maximum (Xia et al., 2024; Ivison et al., 2025; Dai et al., 2025). It is not known when a single score suffices, or what a balancing rule should balance.

This paper gives a geometric answer. **Influence magnitude is sufficient only when capability improvements are nearly collinear; when they are not, the relevant question is whether a candidate contributes an effect that the selected data cannot reproduce.** High influence is not new influence (Figure 1): to first order, a candidate adds something only if no cost-matched mixture of the selected examples improves every capability as much. This replication test is a linear program whose dual prices each capability by its scarcity in the selected data; its *arbitrage gap* (replication cost minus the candidate's cost) is positive exactly when no such mixture exists (with an inactive price cap), and by LP duality (Harrison & Kreps, 1979) a selected set leaves no positive gap exactly when it supports the same capability prices as the whole pool. We use these objects to explain when and why scalar influence fails, not to propose a better selector.

We report three findings, each tested by an experiment that could have come out otherwise.

- **Scalar influence has a geometric operating regime, and real payoffs lie outside it.** At rank one with a positive direction, fixed prices, task-wise max, round-robin, capability-level BIDS and greedy gap maximization select the same set; in an ε-cone around a positive unit direction v with

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

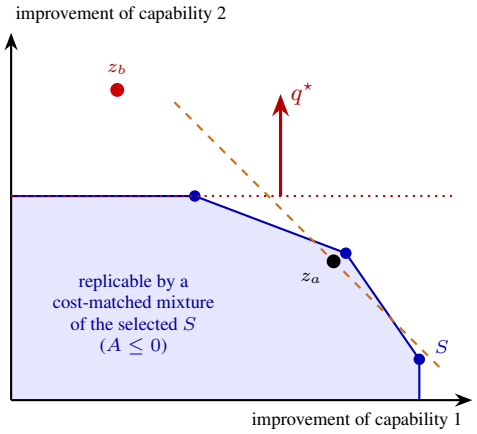

Figure 1: **High influence is not new influence.** Each example is a vector of first-order improvements on two capabilities; mixtures of the selected data S (blue) costing at most one example reproduce the shaded region. Candidate z_a has the larger mean influence (0.565 vs. 0.51; dashed: equal mean influence), yet S already delivers its improvement (arbitrage gap $A \le 0$). The scarcity price $q^\star$ (red; dotted: its supporting line) prices capability 2, which S serves worst; it values z_b, outside the region, at 1.52 ($A = 0.52 > 0$), and no-arbitrage selection picks z_b.

smallest coordinate $v_{\min}$, a fixed linear price keeps a factor $((v_{\min} - \varepsilon)/(v_{\min} + \varepsilon))^2$ on every capability, tight up to a constant for two capabilities (Theorem 1). Real matrices are far outside this cone; there a fractional coverage ceiling, a minimax over capability prices, bounds what one score loses (Proposition 1). Deforming real payoffs to rank one makes the rules coincide, and they separate as this bound grows (Section 4.1).

- **Replication separates magnitude from first-order non-redundancy, and its learning meaning is local.** The gap has an exact dual in scarcity prices, is unit-free (Theorem 2, Proposition 2), and equals the one-step, price-weighted advantage of a candidate over every cost-matched mixture of selected data (Theorem 3). Real gradient steps confirm the identity ($r = 1.000$), but it stops predicting within 3% of the parameter distance that training travels; the warm-up rankings survive training on convex heads but not on a CNN (Section 4.2).
- **Coverage diagnoses failure; gradient distance to the targets also ranks selectors.** Scalar influence falls below random sampling in all 32 targeted comparisons, and first-order coverage flags this, but on PACS and OfficeHome it barely orders the rules that avoid failure, while the gradient distance to the targets does; among them, matching target gradients directly is the most accurate (Sections 4.3 to 4.5).

Greedy gap maximization (under unit costs the greedy algorithm for regret-minimizing sets, Nanongkai et al., 2010) serves as a probe of the theory. Our evidence comes from image benchmarks; we do not fine-tune language models.

2 Related work

Influence- and gradient-based selection. Influence functions (Koh & Liang, 2017), TracIn (Pruthi et al., 2020), datamodels (Ilyas et al., 2022) and TRAK (Park et al., 2023) estimate how a training example changes a test loss; such estimates are local (Basu et al., 2021; Bae et al., 2022). LESS (Xia et al., 2024) made gradient-similarity selection practical for instruction tuning; later work improves the gradients (Min et al., 2026), the proxy model (Chen et al., 2026) or the cost (Yu et al., 2024), matches distributions (Xie et al., 2023a; Liu et al., 2024; Feng et al., 2025; Cheng et al., 2026), or selects with datamodels (Engstrom et al., 2024). Gradient matching (Mirzasoleiman et al., 2020; Killamsetty et al., 2021a;b; Deng et al., 2024; Zhang et al., 2025a), coresets (Sener & Savarese, 2018) and difficulty scores (Paul et al., 2021) target one objective; gradient matching replicates one aggregate target gradient, i.e., at a *fixed* capability weighting.

Several capabilities. BIDS (Dai et al., 2025), round-robin over tasks (Ivison et al., 2025), submodular task mixtures (Renduchintala et al., 2024), influence consensus (Wu et al., 2024), domain reweighting (Xie et al., 2023b), robust submodular selection (Krause et al., 2008), group-robust training (Sagawa et al., 2020) and removing the examples that hurt the worst group (Jain et al., 2024) serve the worst capability; influence is not additive over groups (Koh et al., 2019), top-k influence sets can be redundant (Hu et al., 2024), diversity-aware methods cluster before scoring (Zhang et al., 2025b), and multi-objective learning finds common descent directions (Sener & Koltun, 2018). Two

Table 1: **Closest rules**: what a candidate is compared with, how capabilities are weighted, and the formal statement attached to the rule.

Rule	compares a candidate with	capability weighting	formal statement
LESS mean, task-wise max (Xia et al., 2024)	nothing (ranks once)	fixed, or best coordinate	first-order influence
Round-robin, KNN-uniform (Ivison et al., 2025; Nayak et al., 2026)	nothing (per task or query)	one at a time	distance bounds (Nayak)
BIDS (Dai et al., 2025)	average selected example	z-scored, then max	none
GRAD-MATCH (Killamsetty et al., 2021b)	residual of one aggregate target	fixed	gradient-error bound
Regret-minimizing sets (Nanongkai et al., 2010)	best selected tuple	every linear preference	regret ratio
Category-wise influence LP (Nahin et al., 2025)	reweightings of the training data	Pareto	Pareto-improvement test
This work	best cost-matched mixture of S	scarcity prices	Thms. 1 to 3, Props. 1, 2

works are closest. Nahin et al. (2025) solve a linear program over *reweightings* of the training set that tests whether every class can still improve; we ask whether the *selected* data reproduce a candidate, which yields completeness and scarcity prices for budgeted selection, and our max-min LP baseline is the closest budgeted analogue of their test. Nayak et al. (2026) find on instruction tuning that gradient representations with round-robin or nearest-neighbour selection often work best and that the gradient distance of a subset to the queries predicts performance; on our selections that distance predicts accuracy better than coverage (Section 4.3). We explain why scalar rules fail.

Regret and prices. Regret-minimizing sets choose k tuples that serve every linear preference almost as well as the whole table (Nanongkai et al., 2010; Agarwal et al., 2017); with unit costs, non-negative payoffs and no cap our greedy is theirs (Proposition 3). Arbitrage-free pricing has been used to sell data and models (Koutris et al., 2015; Chen et al., 2019), and theory explains when curation beats more data (Dohmatob et al., 2026); we use no-arbitrage duality (Delbaen & Schachermayer, 1994) to choose training data. What is new is not the greedy, the LP or the duality (Table 1), but the sufficiency analysis of one score and its tests on trained models.

3 One score, several capabilities, and replication

Payoffs. Let θ be the current (warm-up) parameters, $D = \{z_1, \dots, z_N\}$ a candidate pool, and $L_1, \dots, L_K$ the validation losses of K capabilities (domains, benchmarks, or subtasks); Appendix Table 4 collects the notation. The *payoff* of a candidate is its vector of first-order capability improvements for a step direction $d(z)$,

$$r_k(z) = \big\langle \nabla L_k(\theta),\, d(z) \big\rangle, \qquad r(z) = (r_1(z), \dots, r_K(z)) \in \mathbb{R}^K, \tag{1}$$

so that a step $\theta - \eta\, d(z)$ lowers L_k by $\eta r_k(z) + O(\eta^2)$; with $d(z) = P\nabla\ell(z;\theta)$ (training loss ℓ, preconditioner P) these are influence scores. We use LESS-style cosines (Xia et al., 2024) at one warm-up checkpoint with plain gradients (LESS averages checkpoints and uses Adam): $r_k(z)$ averages $\cos(\nabla\ell(j;\theta), \nabla\ell(z;\theta))$ over the validation examples j of capability k, which is (1) for $d(z) = \nabla\ell(z;\theta)/\|\nabla\ell(z;\theta)\|$ and the loss L_k that weights each validation example by the inverse norm of its gradient at θ. We write $R = [r(z_1); \dots; r(z_N)] \in \mathbb{R}^{N\times K}$; every example costs $c(z) > 0$ (one unit here).

How scalar rules compress payoffs. Mean influence (Xia et al., 2024) ranks by the fixed price $q_0 = \frac{1}{K}\mathbf{1}$, i.e., by $q_0^\top r(z)$; task-wise max values each example at its best coordinate; round-robin (Ivison et al., 2025) and BIDS (Dai et al., 2025) move a coordinate price over the capabilities as the selection grows. None asks whether z adds anything the *already selected* data cannot reproduce.

Definition 1 (Replication cost, arbitrage gap, completeness)**.** Let $S \subseteq D$ be the selected set and $\rho \in (0, \infty]^K$ a price cap. The replication cost of z is

$$C(z \mid S) = \min_{w\in\mathbb{R}^S_{\geq 0},\, s\in\mathbb{R}^K_{\geq 0}} \sum_{i\in S} c(i)\, w_i + \rho^\top s \quad \text{s.t.} \quad \sum_{i\in S} w_i\, r(i) + s \;\geq\; r(z), \tag{2}$$

and its *arbitrage gap* is $A(z \mid S) = C(z \mid S) - c(z)$. S is δ-*complete* for D if $\max_{z\in D} A(z \mid S) \leq \delta$.

Here w_i is the (fractional) number of steps spent on example i, and s buys improvement from an outside option at price ρ (absent for $\rho = \infty$; a finite ρ keeps (2) bounded). If the optimal replication does not use the outside option (an *inactive* cap), $A(z \mid S) > 0$ exactly when no cost-matched mixture of selected examples reproduces the improvement of z on every capability; with an active cap, a positive gap lower-bounds the unreplicated improvement. We use $\rho_k = M/\max_{z\in D} r_k(z)^+$, $x^+ = \max(x, 0)$, $M = 10$. The finance terms are an analogy; prices are LP multipliers.

3.1 When is one influence score enough?

Theorem 1 (Rank-one collapse, cone sufficiency and tightness)**.** *Let all costs equal one.*

(a) Collapse. *If $r(z) = a(z)\, v$ for a fixed $v \in \mathbb{R}^K$ with positive entries, then for every budget $B \le |\{z : a(z) > 0\}|$, top-B selection by $q_0^\top r$ for any price $q_0 \ge 0$, $q_0 \ne 0$, task-wise max, round-robin, BIDS with capability-level columns and greedy gap maximization with $\rho < \infty$ all return the B examples with the largest a (up to ties in a).*

(b) Sufficiency. *Suppose $r(z) = a(z)\, u(z)$ with $a(z) > 0$ and unit vectors $u(z)$ satisfying $\|u(z) - v\|_2 \le \varepsilon$ (the ε-cone around v) for a unit $v \ge 0$ with $v_{\min} = \min_k v_k > \varepsilon$. Let S_{q_0} be the top-B set of a fixed linear price $q_0 \ge 0$, $q_0 \ne 0$. Then for every set S of size B and every capability k,*

$$\textstyle\sum_{z\in S_{q_0}} r_k(z) \;\ge\; \kappa(\varepsilon) \sum_{z\in S} r_k(z), \qquad \kappa(\varepsilon) = \left(\frac{v_{\min}-\varepsilon}{v_{\min}+\varepsilon}\right)^2 = 1 - \tfrac{4\varepsilon}{v_{\min}} + O\big((\varepsilon/v_{\min})^2\big), \quad (3)$$

and S_{q_0} is δ-complete with $\delta = 4v_{\min}\varepsilon/(v_{\min} - \varepsilon)^2$.

(c) Tightness. *For $K = 2$, every $v_{\min} \in (0, 1/\sqrt{2}\,]$ and $\varepsilon \in (0, v_{\min})$, some pools in the ε-cone make the top-B set of the uniform price retain only a factor $1 - c\,\varepsilon/v_{\min}$, $c \in [1/\sqrt{2}, 1]$, on some capability against a set of the same size; for $\varepsilon > v_{\min}$ the loss can be total.*

Part (a) is exact, (b) bounds what any rule, ours included, can gain over one linear price when payoff directions are concentrated, and (c) shows that unequal capability scales, which shrink $v_{\min}$, can make a narrow cloud defeat the uniform price. Part (b) is a certificate near rank one, not a model of the observed regime: its condition fails on every real matrix we measure ($\varepsilon/v_{\min}$ from 1.2 to 34, Appendix Table 6). To measure what one score loses on any matrix, let $b_k = \max_{|S|=B} \sum_{z\in S} r_k(z)$ (capabilities with $b_k \le 0$ are dropped) and $\mathrm{cov}(S) = \min_k \sum_{z\in S} r_k(z)/b_k$ for $|S| = B$, the smallest fraction of any capability's best first-order gain that S secures (*worst-capability coverage*); $c_{\mathrm{disc}} = \max_{|S|=B} \mathrm{cov}(S)$.

Proposition 1 (Fractional coverage ceiling and scalarization gap)**.** *Let $c^\star = \max\{\min_k \sum_z x_z r_k(z)/b_k \;:\; x \in [0,1]^N,\ \sum_z x_z = B\}$ be the fractional ceiling and $\tilde{r}_k = r_k/b_k$. (i) $\mathrm{cov}(S) \le c_{\mathrm{disc}} \le c^\star$ for every set S of size B, whatever rule produced it. (ii) $c^\star = \min_{q\in\Delta_K} \mathrm{Top}_B(q^\top \tilde{r})$, where Δ_K is the probability simplex and $\mathrm{Top}_B(f)$ the sum of the B largest values of f over the pool: every price certifies an upper bound on the coverage any selection can guarantee to all capabilities, and the best certificate is exact for the fractional problem. (iii) The* scalarization gap *$G(q_0) = c^\star - \mathrm{cov}(S_{q_0})$ of a fixed price is non-negative and exceeds its loss against the best set, $c_{\mathrm{disc}} - \mathrm{cov}(S_{q_0})$, by the integrality gap $c^\star - c_{\mathrm{disc}}$; at rank one with a positive direction $c^\star = c_{\mathrm{disc}} = 1$ and $G(q_0) = 0$, and in the ε-cone of Theorem 1(b), $G(q_0) \le 1 - \kappa(\varepsilon)$.*

On real payoff matrices with class-level capabilities, $c^\star \le 0.16$ on every benchmark and target pair (rounding the fractional solution attains 93% of it on average on PACS and OfficeHome and about half on RotatedMNIST; Appendix A.7), and every selection we evaluate, random sampling included, has negative coverage: to first order, training on it raises the loss of some class. What separates the rules that fail is how negative it is (Section 4).

3.2 Replication and scarcity prices

Theorem 2 (Data completeness)**.** *Let $Q(S) = \{q \in \mathbb{R}^K :\ 0 \le q \le \rho,\ q^\top r(i) \le c(i)\ \forall i \in S\}$ be the set of* supporting capability prices *of S.*

(i) $C(z \mid S) = \max_{q\in Q(S)} q^\top r(z)$, and both optima are attained when $\rho < \infty$, or when $\rho = \infty$ and a replication exists.

(ii) $A(z \mid S) \le 0 \iff$ *a replication* (w, s) *of cost at most* $c(z)$ *exists* $\iff$ *every supporting price values* z *at most at its cost.*

(iii) S *is* 0*-complete for* $D \iff Q(S) = Q(D)$*: the selected data support exactly the same capability prices as the whole pool.*

Part (iii) is LP duality, the finite-dimensional core of the fundamental theorem of asset pricing (Harrison & Kreps, 1979; Delbaen & Schachermayer, 1994). An optimal $q^\star$ in (i) is a *scarcity price*: with non-negative payoffs, $q_k \le c(i)/r_k(i)$ for every $q \in Q(S)$ and selected i with $r_k(i) > 0$, so $q_k^\star$ is large only if S improves capability k poorly per unit cost. With non-negative payoffs, unit costs and no cap, the largest gap is $\mathrm{mrr}(S)/(1 - \mathrm{mrr}(S))$ for the maximum regret ratio, and greedy gap maximization is the greedy algorithm for regret-minimizing sets (Proposition 3, Appendix A); our payoffs are signed, so this is a structural correspondence, and we claim no guarantee for the greedy.

Proposition 2 (Structure of the gap)**.** *(a)* Monotone: $S \subseteq S' \Rightarrow A(z \mid S') \le A(z \mid S)$. *(b)* Unit-free: *if every capability is rescaled,* $r_k \mapsto \lambda_k r_k$ *with* $\lambda_k > 0$*, and* $\rho_k \mapsto \rho_k/\lambda_k$ *(as for our choice of* ρ*), every gap is unchanged; the ranking of a fixed linear price with two or more positive weights changes under such a rescaling whenever two candidates trade off two capabilities. (c)* No double counting: *if* $r(z) \le r(i)$ *coordinatewise and* $c(z) \ge c(i)$ *for some* $i \in S$*, then* $A(z \mid S) \le 0$*. (d)* Frontier: *if* $\rho < \infty$ *and* $A(z^\star \mid S) = \max_z A(z \mid S) > 0$*, then* $z^\star$ *maximizes* $q^{\star\top} r(z) - c(z)$ *over the pool for its optimal price* $q^\star \neq 0$*.*

By (d), before completeness the greedy selects only frontier examples, not a representative sample. Round-robin and BIDS are unit-free too; mean influence and task-wise max are not.

3.3 The gap and one training step

Theorem 3 (Learning interpretation)**.** *Let the payoffs be defined by* (1) *for directions with* $\|d(z)\| \le \Gamma$ *(*$\Gamma = 1$ *for LESS cosines) and* β_k*-smooth losses* L_k*. Compare one step on* z*,* $\theta_z^+ = \theta - \eta\, d(z)$*, with a mixed step on* S*,* $\theta_w^+ = \theta - \eta \sum_{i \in S} w_i\, d(i)$*, where* $w \ge 0$ *has cost* $\sum_i c(i) w_i \le c(z)$ *(so* $\|w\|_1 \le c(z)/\min_i c(i)$*).*

(a) If w *replicates* z *without the outside option,* $\sum_{i \in S} w_i r(i) \ge r(z)$ *(such a* w *exists when* $A(z \mid S) \le 0$ *and* $\rho = \infty$*), then for every capability* k*,* $L_k(\theta_w^+) \le L_k(\theta_z^+) + \frac{\beta_k}{2}\eta^2\Gamma^2\big(1 + \|w\|_1^2\big)$*.*

(b) For any cap ρ*, if* $A(z \mid S) = \gamma > 0$ *with optimal price* $q^\star$*, then for* every *cost-matched* w *the price-weighted loss* $L_{q^\star} = \sum_k q_k^\star L_k$ *satisfies* $L_{q^\star}(\theta_z^+) \le L_{q^\star}(\theta_w^+) - \eta\gamma + \frac{1}{2}\eta^2\Gamma^2\big(\sum_k q_k^\star \beta_k\big)\big(1 + \|w\|_1^2\big)$*. If an optimal replication uses no outside option, rescaled to cost* $c(z)$ *it attains the first-order bound: its* $q^\star$*-weighted first-order improvement is exactly* γ *below that of* z*.*

The theorem compares one update from θ and says nothing about longer training. With an inactive cap, a non-positive gap means that some cost-matched mixture of selected data improves *every* capability at least as much as z, to first order, and $\gamma > 0$ equals the first-order advantage of z, at the scarcity prices, over every such mixture; with an active cap, γ lower-bounds it.

3.4 The selection rule as a probe

NoArb adds $\arg\max_z A(z \mid S)$ until S is complete (Algorithm 1, Appendix B); since gaps only shrink as S grows (Proposition 2(a)), lazy evaluation is exact. Past the completeness point it starts a new layer on the residual pool, for which our results hold with respect to the layer only; on the benchmarks of Table 3 and untargeted PACS, all picks up to 5% lie in the first layer, and at most 22% at 10% in later ones. NoArb targets completeness over all prices, not worst-capability accuracy: it is a probe of the theory. Table 2 maps each result to its prediction and test.

4 Experiments: testing the predictions

Setup (Appendix B). All influence-based rules read the *same* payoff matrix R: LESS-style cosines at a warm-up model θ_0 trained on a random 5% of the pool, with (domain, class) pairs as capabilities (domain-level capabilities: Section 4.4). NoArb uses unit costs and $M = 10$ (a 50-fold range

Table 2: **Theory to experiment**: prediction, hypotheses checked on our payoffs, and test.

Result	Prediction	Holds on our R	Test and outcome
Thm. 1(a)	all rules coincide	only on R_0	identical selections in every seed (Sec. 4.1)
Thm. 1(b,c)	one price keeps $\kappa(\varepsilon)$	no ($\varepsilon/v_{\min} \geq 1.2$)	synthetic cones: holds in all 100 runs (App. D)
Prop. 1	fractional ceiling; $G(q_0)$	yes	rules separate as G opens along R_α; weak across matrices (Sec. 4.1)
Thm. 2	duality, completeness	yes	dual equals primal on real matrices (App. A.7)
Prop. 2(b,c)	unit-free; copies worthless	yes	units test (Sec. 4.4); near-copies (App. D)
Thm. 3	one-step advantage (exact if cap inactive)	yes (one step)	exact for one step; lasts $< 3\%$ of the parameter distance (Sec. 4.2)
Prop. 3 (App.)	regret-set greedy	no (signed payoffs)	random non-negative pools (App. A.7)

of M changes its selections by at most 5.4% on average, Appendix Table 19). Baselines: random and class-stratified random sampling, and two references that know the target domains (random and class-stratified sampling within them); embedding similarity with round-robin over capabilities (RDS-style, Ivison et al., 2025); KNN-uniform, where each target validation example in turn adds its most gradient-similar candidate (Nayak et al., 2026); GRAD-MATCH (Killamsetty et al., 2021b) on the mean target gradient and with normalized target gradients; and, on the shared payoffs, LESS mean and task-wise max (Xia et al., 2024), a max-min LP relaxation of the worst normalized capability gain, round-robin and BIDS (Dai et al., 2025) with capability- and instance-level columns. Ablations: *static* and *fixed price* rank once, by $A(z \mid \emptyset)$ and by a uniform price on normalized payoffs; *centroid* replicates with the average selected example. Models continue training from θ_0 on the warm-up data plus the selected data (the *incremental protocol*). We report worst-target test accuracy (5 seeds, 95% t-intervals); significance marks are paired t-tests against NoArb, Holm-corrected over all rules of a table column (other families are stated).

4.1 Q1: When does one score fail? (Theorem 1, Proposition 1)

If scalar failure comes from capability geometry rather than from the training protocol, the payoff rules must coincide at rank one and separate only as the scalarization gap opens. On targeted RotatedMNIST (Section 4.3) we replace the class-level payoffs R by $R_\alpha = (1-\alpha)R_0 + \alpha R$, where R_0 gives every capability the candidate's mean payoff, keeping data, warm-up model, training and budget fixed. At $\alpha = 0$ all five rules select the *identical* 600 examples in every seed. As α grows, $c^\star(R_\alpha)$ falls from 1 to 0.15 at $\alpha = 0.25$ and 0.005 at $\alpha = 1$, and $G(q_0)$ of mean influence grows from 0 to 0.38 (Figure 2a); Theorem 1(b) is vacuous for every $\alpha > 0$. On the real R, the common selection at $\alpha = 0$ has coverage -0.38, and the capability-aware rules reach -0.05 to -0.10 from $\alpha = 0.25$ on while mean influence stays at -0.38 (Figure 2b). Accuracy follows (Figure 2c): the best capability-aware rule is 0, 4.8, 14.8 and 18.7 points above mean influence at $\alpha = 0$, 0.05, 0.1 and 0.25 and 20 to 21 points beyond, and over the 35 (rule, α) cells coverage on R ranks accuracy with Spearman correlation 0.86 (0.69 over the 175 runs); round-robin is ahead of NoArb at the two smallest spreads. Across real matrices the gap is a weaker guide: over 18 cells (benchmarks, target pairs, budgets), $G(q_0)$ and the gain of the best capability-aware rule over mean influence have Spearman correlation only 0.42 ($p = 0.08$). *Units:* in synthetic cones with equal units, mean influence covers the worst capability best at wide angles (43% at 90°, NoArb 35%); unequal units collapse the unnormalized scalar rules but not the unit-free ones (Appendix Figure 4).

4.2 Q2: Does the gap measure unreplicable learning, and for how long? (Theorem 3)

We test Theorem 3 where selection operates: targeted PACS, cosine payoffs with $M = 10$, and S the first 60 picks of NoArb. For 400 random candidates per seed (2,000 in all, 283 with $A > 0$; the cap is inactive for all) we compare a step on z with a cost-matched step on its replicating mixture, along the directions $d(z)$ and on the weighted losses for which cosines are exact payoffs; their price-weighted loss difference over η should equal $\max(A, 0)$. For one step of size $\eta = 10^{-4}$ it does (correlation 1.000, slope 0.999, median relative error 0.19% for $A > 0$; Figure 3a), far more tightly than the worst-case bound guarantees (Appendix A.7). A first-order identity must hold for small steps, so the informative question is its horizon. Over H steps of size 10^{-2}, the correlation is 0.97, 0.71, -0.04 and -0.25 at path lengths $H\eta = 0.01$, 0.03, 0.1 and 0.3, and the share of positive-gap

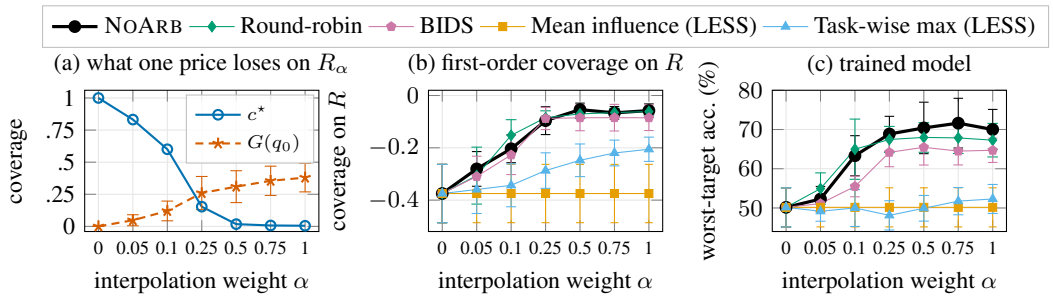

Figure 2: **Rank one makes the payoff rules equal; they separate as the scalarization gap opens** (targeted RotatedMNIST, payoffs interpolated from rank one, $\alpha = 0$, to the observed R, $\alpha = 1$; 10% budget, 5 seeds, 95% t-intervals). (a) Ceiling $c^\star(R_\alpha)$ and gap $G(q_0)$ of mean influence. (b) Coverage of each selection on the real R. (c) Worst-target accuracy.

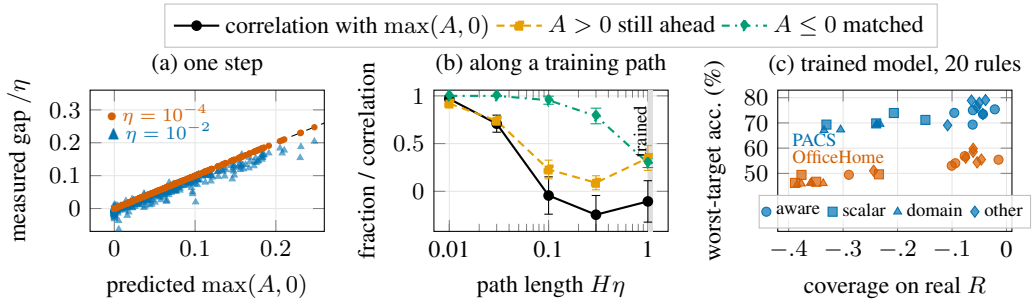

Figure 3: **From one step to a trained model.** (a,b) Targeted PACS, 5 seeds, 2,000 candidates (plotted: the 283 with $A > 0$ and 150 others). (a) Price-weighted loss difference between a step on the candidate and one on its replicating mixture. (b) Over H steps of size 10^{-2}: correlation with $\max(A, 0)$, share of $A > 0$ candidates still ahead, share of $A \leq 0$ candidates matched on every capability; grey: distance training moves the head. (c) Coverage on class-level payoffs against worst-target accuracy, 20 rules (Appendix Table 7), 10% budget.

candidates still ahead of their mixture falls from 0.92 to 0.09 (Figure 3b). Training moves the head by 1.01 to 1.13, so the one-step prediction stays informative for less than 3% of the parameter-space distance it travels. Recomputing the payoffs at checkpoints of the actual run (Appendix Table 5): on the convex PACS and OfficeHome heads the warm-up rankings survive the whole run (at the final head, Spearman 0.90 and 0.93 for the mean payoff, 0.93 for the gaps), so there the short horizon is a limit of the first-order model of learning, not of the scores; on the RotatedMNIST CNN the ranking of mean influence changes after one Adam step (Spearman 0.64, top-10% overlap 0.37, at 2.7% of the distance), while per-capability payoffs (0.93) and gaps (0.89) change more slowly.

4.3 Q3: Targeted selection: what does first-order coverage predict?

As in instruction tuning, the pool is broader than the target: targeted RotatedMNIST pools digits at 24 rotation angles and targets $\{0, 30°, 60°, 90°\}$; targeted PACS and OfficeHome pool four domains and target two (cartoon and sketch; Clipart and Product), fixed before any targeted result was seen (Table 3; paired differences: Appendix Table 16).

(i) *Coverage flags the rules that fail.* Over 20 rules (Appendix Table 7), coverage on the class-level payoffs ranks worst-target accuracy with Spearman correlation 0.71 and 0.74 on targeted PACS, 0.83 and 0.88 on targeted OfficeHome and 0.92 and 0.91 on targeted RotatedMNIST at 5 and 10% (Figure 3c). Scalar influence and the rules that read domain-level payoffs have coverage between -0.14 and -0.39 and are among the least accurate. On PACS and OfficeHome the correlation rests on this separation: over our 16 cells it is 0.43 to 0.84 without the five domain-level variants and 0.03 to 0.60 among the 11 rules that avoid scalar scores and domain-level capabilities (95% intervals include zero in 10 cells), and the gradient distance of the selection to the targets (Nayak

Table 3: **Targeted selection**: worst-target test accuracy (%, mean ± 95% t-interval, 5 seeds). Target-domain rules know the targets. Bold: best rule that does not; underline: best on the shared payoffs; †/‡: below/above NOARB (paired t-test, $p < 0.05$, Holm over all rules of the column, incl. Appendix Tables 15, 17, 18).

	RotatedMNIST		PACS		OfficeHome	
Rule	5%	10%	5%	10%	5%	10%
no payoffs						
Random	58.1 ±4.0†	61.3 ±2.7	74.9 ±2.4	78.0 ±3.0‡	52.7 ±2.9	56.2 ±3.0
Class-stratified random	60.7 ±2.9†	62.8 ±3.6	73.6 ±3.1	75.3 ±3.0	51.7 ±2.6	55.8 ±1.8
Target-domain random	69.7 ±2.5	73.2 ±3.1	74.2 ±2.0	76.8 ±1.9	53.6 ±2.8	58.5 ±1.7
Target-domain random, class-stratified	69.9 ±3.4	75.5 ±1.9	76.2 ±2.0	78.2 ±2.4	55.7 ±2.2‡	59.3 ±0.9‡
features or full gradients						
Embedding similarity (RDS)	61.6 ±4.5†	63.1 ±3.6	74.4 ±3.6	75.9 ±3.2	52.8 ±1.7	55.5 ±1.9
KNN-uniform (gradient cosines)	60.9 ±5.8†	70.0 ±5.0	74.5 ±3.1	78.9 ±2.4‡	56.4 ±2.7‡	58.6 ±3.1‡
GRAD-MATCH	**71.1** ±2.9	72.4 ±3.7	77.1 ±1.0	**79.1** ±1.9	**56.6** ±2.9‡	**59.8** ±2.4‡
GRAD-MATCH, normalized targets	69.8 ±5.8	**73.5** ±4.5	**77.8** ±1.3	78.8 ±0.8	56.4 ±1.9‡	59.7 ±1.9‡
influence payoffs R (shared input)						
LESS, mean	49.2 ±3.6†	50.2 ±5.0†	70.6 ±1.9	69.8 ±3.1	46.2 ±2.8†	46.2 ±2.9†
LESS, task-wise max	50.0 ±3.1†	52.3 ±3.7†	70.3 ±2.2	69.3 ±3.1	46.0 ±3.0†	46.7 ±3.2†
Max-min LP	63.6 ±2.5†	63.6 ±6.7	72.6 ±2.2	72.7 ±2.1†	51.2 ±2.9	53.2 ±2.5
Round-robin	63.7 ±3.8†	67.3 ±4.3	73.5 ±3.8	73.5 ±3.2	52.9 ±1.8	55.4 ±1.5
BIDS	61.6 ±4.8	64.7 ±3.1	70.2 ±2.9	69.3 ±3.1	50.3 ±1.6†	53.0 ±1.4
BIDS, instance-level	60.8 ±4.3†	65.7 ±4.3	72.7 ±3.2	74.0 ±3.2	<u>53.6</u> ±1.3	<u>56.8</u> ±1.6
NOARB (ours)	<u>69.4</u> ±1.9	<u>70.0</u> ±5.1	<u>74.6</u> ±3.8	<u>75.4</u> ±2.9	52.5 ±1.9	54.0 ±1.9
w/o replication (static)	51.1 ±3.0†	50.9 ±3.5†	72.2 ±2.5	73.9 ±3.4	47.3 ±2.0†	49.4 ±2.0†
w/ fixed price	49.7 ±2.8†	50.3 ±3.2†	71.1 ±2.0	71.2 ±3.5	48.2 ±2.0†	49.6 ±1.8
w/ centroid reference	53.0 ±3.0†	56.3 ±3.4†	73.8 ±3.5	75.1 ±3.7	48.0 ±2.1†	49.4 ±2.0†

et al., 2026) ranks accuracy better in all 16, over the 20 rules (Spearman 0.73 to 0.96) and among the 11 (0.47 to 0.87, intervals above zero; Appendix Table 8); only on RotatedMNIST does coverage also order the 11 (0.88 and 0.91). Coverage is a diagnostic of failure, not a ranking surrogate. (ii) *Scalar influence fails*: LESS mean and task-wise max fall 4 to 11 points below random sampling in all 12 cells of Table 3 (paired $p \leq 0.032$, also after Holm correction over these 12). (iii) *On the shared payoffs*, NOARB has the highest mean on RotatedMNIST and PACS, instance-level BIDS on OfficeHome; round-robin, class-balanced influence over the same cells, differs from NOARB after Holm correction only on RotatedMNIST at 5%. (iv) *Among the rules that do not know the targets, gradient matching*, with or without normalized targets, is the most accurate in every cell of Table 3; KNN-uniform is within 1.2 points of it at 10% on PACS and OfficeHome. NOARB beats random sampling only on RotatedMNIST (by 11.3 and 8.7 points), whose four targets are a sixth of the pool, and falls below it on PACS and OfficeHome, significantly on PACS at 10% (-2.6 ± 0.9).

4.4 Q4: DOES THE GAP CARRY INFORMATION BEYOND SCALAR INFLUENCE?

Same influence, different replicability. On untargeted PACS we pair influential positive-gap candidates with replicable ones of nearly equal mean influence. Adding the positive-gap members gives 0.9 and 1.2 more points of macro accuracy when matched on influence alone or also on domain (Holm-adjusted $p = 0.011$ and 0.003) and 0.75 (not significant) when also matched on class; on worst-domain accuracy none of the three is significant after Holm correction ($p = 0.052$, 0.072, 0.21). Part of what the gap detects is uncovered classes, and random candidates from the same half beat both members (82.5 against 80.9 and 80.0, $p \leq 0.02$): the gap carries information beyond influence, but its frontier examples are not a representative sample (Appendix Table 13).

Replication, units and granularity. Ranking once (static, fixed price) costs 18 to 20 worst-target points on RotatedMNIST and 1.5 to 5.2 on PACS and OfficeHome, and replicating with the average selected example (centroid) 14 to 16 and 4.5 points on RotatedMNIST and OfficeHome; k-center on the payoff vectors is 7 to 13 points below NOARB on RotatedMNIST (Table 3, Appendix Table 15).

Rescaling one target's payoffs and gradients by $\lambda \in \{0.1, 10\}$ leaves NOARB (Proposition 2(b)), round-robin and BIDS unchanged and costs GRAD-MATCH 7.0 ± 3.5 and 8.6 ± 4.2 points ($p = 0.005$), unless its targets are normalized (Appendix Table 21). With domain-level capabilities every capability-aware rule falls to scalar influence (NOARB 70.0 to 50.6 on RotatedMNIST at 10%; Appendix Table 9), as domain-level payoffs average over classes and hide those a target misses; on instruction-tuning payoffs, NOARB has the best worst-benchmark coverage with benchmark-level capabilities but the lowest (0.52, mean influence 0.77) with 93 subtasks (Appendix Table 22).

4.5 Q5: HELD-OUT TARGET PAIRS AND THE UNTARGETED POOL

Held-out target pairs are our confirmatory test: after all other results we ran the five other pairs of PACS domains as targets, protocol unchanged (Appendix Table 10). LESS mean and task-wise max fall 2.7 to 8.1 points below random sampling in all 20 comparisons (15 significant after Holm correction over the 20), coverage ranks accuracy with Spearman correlation 0.57 to 0.80, and in every cell the most accurate rule that does not know the targets matches target gradients directly (GRAD-MATCH 5, KNN-uniform 5). NOARB is above random sampling in 7 of 10 cells, but pooled over 25 runs by only $+1.5 \pm 0.8$ points at 5% and $+0.1 \pm 1.0$ at 10%; instance-level BIDS is the best payoff rule in 6 cells. *When the pool is the target* (untargeted PACS), no rule beats random sampling after Holm correction over the rules (GRAD-MATCH comes closest, $+2.0$ points at 5%, adjusted $p = 0.051$), and NOARB falls 2.2 and 2.8 points below it at 10 and 20% (unadjusted $p \leq 0.019$; Appendix Table 11).

5 DISCUSSION, LIMITATIONS AND CONCLUSION

Discussion. Scalar influence measures how strong an improvement is, not whether the selected data already provide it, and real payoff clouds are far from the collinear regime where the two coincide. Replication measures first-order non-redundancy, not utility: its positive-gap examples are frontier examples, not a representative sample. In practice, a selection whose coverage is far below that of random sampling neglects some capability; among the rules that avoid this, the gradient distance to the targets predicted accuracy, and matching target gradients directly was the most accurate.

Limitations. (i) The sufficiency certificate and the regret-set correspondence do not apply to our real matrices, $c^\star$ is a fractional bound, the link to trained accuracy is an empirical rank correlation that weakens among the rules that do not fail, and no guarantee is proved for the greedy. (ii) Payoffs are LESS-style cosines at one checkpoint with plain gradients. (iii) Replication reweights the selected data freely; training weights them uniformly. (iv) NOARB targets completeness, not worst-capability accuracy, and is heuristic beyond its first layer. (v) Capabilities must resolve what the target misses; which definition carries over to instruction tuning, where subtask-level capabilities reversed the ranking in our geometry check, is open. (vi) The protocol and the targeted design were adopted after pilots (Appendix D), held-out pairs cover PACS only, five seeds leave many pairwise differences unresolved, we do not fine-tune language models, and the cost of NOARB grows with the number of capabilities (a median 80 to 2,269 s for $K = 14$ to 130; GRAD-MATCH 394 to 1,401 s) and is untested at instruction-tuning scale.

Conclusion. Scalar influence answers the wrong question once capability directions separate. Replication asks the right first-order question, whether a candidate adds an effect that the selected data cannot reproduce, but it is not by itself a selection criterion, and the first-order account of learning behind it is local. The strongest selectors in our study match target gradients directly, which suggests using scarcity and replication as diagnostics or constraints; whether scarcity prices can guide gradient matching is open.

AI USE STATEMENT

We used generative AI tools (large language models, including a coding agent) extensively. They aided and polished the writing (drafting and editing text), reviewed drafts as simulated reviewers, acted as coding agents that implemented the method, the baselines, the experiments and the tests,

supported discussion throughout the project (including the conceptual framing, hypotheses, mathematical claims, first drafts of the proofs and the experimental methodology, among them the protocol decisions reported in Appendix D), generated the plotting code for all figures, and created slides for internal meetings; they also searched and summarized the related literature and helped format and verify the references. Every reported number is computed by code from logged results, every formal statement is also checked numerically (Appendix A.7), and every reference was checked against a publisher, proceedings or arXiv record. The authors take full responsibility for the content of this paper, including text, claims, proofs and code.

REPRODUCIBILITY STATEMENT

Section 3 defines the payoffs, the replication program and the gap; Section 3.4 and Algorithm 1 (Appendix B) specify the selection procedure; Appendix A gives complete proofs of all formal results with numerical checks. Section 4 and Appendix B list the datasets, splits, warm-up protocol, payoff construction, every baseline, hyper-parameters, seeds and compute. All datasets are public: PACS, OfficeHome, MNIST, and the processed instruction data and evaluation sets released with LESS. Protocol decisions that were changed after a pilot (the incremental training protocol and the targeted design) are reported with the pilot results in Appendix D; the held-out target pairs of Section 4.5 were run afterwards without any change to the protocol.

REFERENCES

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894, 2017.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems*, volume 33, pp. 19920–19930, 2020.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. TRAK: Attributing model behavior at scale. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 27074–27113, 2023.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54104–54132, 2024.

Guanghui Min, Tianhao Huang, Ke Wan, and Chen Chen. GIST: Targeted data selection for instruction tuning via coupled optimization geometry. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Sirui Chen, Yunzhe Qi, Mengting Ai, Yifan Sun, Ruizhong Qiu, Jiaru Zou, and Jingrui He. Influence-preserving proxies for gradient-based data selection in LLM FineTuning. In *International Conference on Learning Representations*, pp. 14913–14941, 2026.

Hamish Ivison, Muru Zhang, Faeze Brahman, Pang Wei Koh, and Pradeep Dasigi. Large-scale data selection for instruction tuning. *arXiv preprint arXiv:2503.01807*, 2025.

Qirun Dai, Dylan Zhang, Jiaqi W. Ma, and Hao Peng. Improving influence-based instruction tuning data selection for balanced learning of diverse capabilities. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 7079–7102, 2025.

J. Michael Harrison and David M. Kreps. Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*, 20(3):381–408, 1979.

Danupon Nanongkai, Atish Das Sarma, Ashwin Lall, Richard J. Lipton, and Jun Xu. Regret-minimizing representative databases. *Proceedings of the VLDB Endowment*, 3(1–2):1114–1124, 2010.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9525–9587, 2022.

Samyadeep Basu, Philip Pope, and Soheil Feizi. Influence functions in deep learning are fragile. In *International Conference on Learning Representations*, 2021.

Juhan Bae, Nathan Ng, Alston Lo, Marzyeh Ghassemi, and Roger B. Grosse. If influence functions are the answer, then what is the question? In *Advances in Neural Information Processing Systems*, volume 35, pp. 17953–17967, 2022.

Zichun Yu, Spandan Das, and Chenyan Xiong. MATES: Model-aware data selection for efficient pretraining with data influence models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 108735–108759, 2024.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34201–34227, 2023a.

Zifan Liu, Amin Karbasi, and Theodoros Rekatsinas. TSDS: Data selection for task-specific model finetuning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 10117–10147, 2024.

Lan Feng, Fan Nie, Yuejiang Liu, and Alexandre Alahi. TAROT: Targeted data selection via optimal transport. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 16837–16852, 2025.

Hao Cheng, Rui Zhang, Ling Li, Na Di, Jiaheng Wei, Zhaowei Zhu, and Bo Han. Task-aware data selection via proxy-label enhanced distribution matching for LLM finetuning. In *International Conference on Learning Representations*, pp. 42789–42812, 2026.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. DsDm: Model-aware dataset selection with datamodels. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12491–12526, 2024.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6950–6960, 2020.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. GLISTER: Generalization based data subset selection for efficient and robust learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9):8110–8118, 2021a.

Krishnateja Killamsetty, Durga S, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. GRADMATCH: Gradient matching based data subset selection for efficient deep model training. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5464–5474, 2021b.

Zhiwei Deng, Tao Li, and Yang Li. Influential language data selection via gradient trajectory pursuit. *arXiv preprint arXiv:2410.16710*, 2024.

Jipeng Zhang, Yaxuan Qin, Renjie Pi, Weizhong Zhang, Rui Pan, and Tong Zhang. TAGCOS: Task-agnostic gradient clustered coreset selection for instruction tuning data. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 4686–4701, 2025a.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20596–20607, 2021.

H S V N S Kowndinya Renduchintala, Sumit Bhatia, and Ganesh Ramakrishnan. SMART: Submodular data mixture strategy for instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 12916–12934, 2024.

Xindi Wu, Mengzhou Xia, Rulin Shao, Zhiwei Deng, Pang Wei Koh, and Olga Russakovsky. ICONS: Influence consensus for vision-language data selection. *arXiv preprint arXiv:2501.00654*, 2024.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, pp. 69798–69818, 2023b.

Andreas Krause, H. Brendan McMahan, Carlos Guestrin, and Anupam Gupta. Robust submodular observation selection. *Journal of Machine Learning Research*, 9(93):2761–2801, 2008.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020.

Saachi Jain, Kimia Hamidieh, Kristian Georgiev, Andrew Ilyas, Marzyeh Ghassemi, and Aleksander Madry. Improving subgroup robustness via data selection. In *Advances in Neural Information Processing Systems*, volume 37, pp. 94490–94511, 2024.

Pang Wei Koh, Kai-Siang Ang, Hubert H. K. Teo, and Percy Liang. On the accuracy of influence functions for measuring group effects. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yuzheng Hu, Pingbang Hu, Han Zhao, and Jiaqi W. Ma. Most influential subset selection: Challenges, promises, and beyond. In *Advances in Neural Information Processing Systems*, volume 37, pp. 119778–119810, 2024.

Chi Zhang, Huaping Zhong, Kuan Zhang, Chengliang Chai, Rui Wang, Xinlin Zhuang, Tianyi Bai, Jiantao Qiu, Lei Cao, Ju Fan, Ye Yuan, Guoren Wang, and Conghui He. Harnessing diversity for important data selection in pretraining large language models. In *International Conference on Learning Representations*, pp. 72980–73003, 2025b.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Shahriar Kabir Nahin, Wenxiao Xiao, Joshua Liu, Anshuman Chhabra, and Hongfu Liu. What is the performance ceiling of my classifier? utilizing category-wise influence functions for pareto frontier analysis. *arXiv preprint arXiv:2510.03950*, 2025.

Nihal V. Nayak, Paula Rodriguez-Diaz, Neha Hulkund, Sara Beery, and David Alvarez-Melis. A critical look at targeted instruction selection: Disentangling what matters (and what doesn't). In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Pankaj K. Agarwal, Nirman Kumar, Stavros Sintos, and Subhash Suri. Efficient algorithms for k-regret minimizing sets. In *16th International Symposium on Experimental Algorithms (SEA 2017)*, volume 75 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 7:1–7:23, 2017.

Paraschos Koutris, Prasang Upadhyaya, Magdalena Balazinska, Bill Howe, and Dan Suciu. Query-based data pricing. *Journal of the ACM*, 62(5):1–44, 2015.

Lingjiao Chen, Paraschos Koutris, and Arun Kumar. Towards model-based pricing for machine learning in a data marketplace. In *Proceedings of the 2019 International Conference on Management of Data (SIGMOD)*, pp. 1535–1552, 2019.

Elvis Dohmatob, Mohammad Pezeshki, and Reyhane Askari Hemmat. Why less is more (sometimes): A theory of data curation. In *International Conference on Learning Representations*, pp. 55454–55490, 2026.

Freddy Delbaen and Walter Schachermayer. A general version of the fundamental theorem of asset pricing. *Mathematische Annalen*, 300(1):463–520, 1994.

Qi Huangfu and J. A. Julian Hall. Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, 10(1):119–142, 2018.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 5543–5551, 2017.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5385–5394, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2551–2559, 2015.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470, 2020.

A PROOFS

Throughout, $R_S \in \mathbb{R}^{|S|\times K}$ stacks the payoffs of the selected examples and c_S their costs; inequalities between vectors are coordinatewise.

A.1 PROOF OF THEOREM 1

(a) With $r(z) = a(z)\,v$ and $v > 0$: (1) any price $q_0 \geq 0$, $q_0 \neq 0$, has $q_0^\top v > 0$ and scores z by $a(z)\,q_0^\top v$, increasing in a; (2) the task-wise max equals $a(z)\max_k v_k$ for $a(z) > 0$ and is non-positive otherwise; (3) every column $r_k = a\,v_k$ orders the candidates by a, so round-robin takes the largest remaining a at every step; (4) BIDS standardizes each column of R to Z and adds the candidate that maximizes $\max_j(Z_{zj} - \bar{Z}_{S,j})$, where $\bar{Z}_{S,j}$ is the mean of column j over the selected set; if a is not constant, all columns of Z equal $(a-\bar{a})/\mathrm{sd}(a)$, so this utility is increasing in $a(z)$, and if a is constant every set of B candidates consists of largest values of a; (5) for greedy gap maximization with $\rho < \infty$, $A(z \mid \emptyset) = a(z)\rho^\top v - c$ for $a(z) > 0$ (and $-c$ otherwise) is increasing in $a(z)$, and after picking $z^{(1)}$ with the largest a, every z with $0 < a(z) \leq a(z^{(1)})$ is replicated by $w = a(z)/a(z^{(1)}) \leq 1$ units of $z^{(1)}$, so $A(z \mid \{z^{(1)}\}) \leq 0$; the layer is then complete and the next layer starts from the largest remaining a (in the non-layered variant, $A(z \mid S) = a(z)/a(z^{(1)})\,c - c$ is again increasing in a). All rules therefore return the B largest values of a among positive candidates.

(b) Write $u(z) = v + \Delta(z)$ with $\|\Delta(z)\| \leq \varepsilon$, so $|\Delta_k(z)| \leq \varepsilon$ and $r_k(z) \in a(z)[v_k - \varepsilon,\, v_k + \varepsilon]$ with $v_k - \varepsilon \geq v_{\min} - \varepsilon > 0$. Since $q_0 \neq 0$ we may normalize $\|q_0\|_2 = 1$; then $\mu_0 := q_0^\top v \geq v_{\min}\|q_0\|_1 \geq v_{\min}$ and $q_0^\top r(z) \in a(z)[\mu_0 - \varepsilon, \mu_0 + \varepsilon]$. Let S^* collect the B largest values of a. Optimality of S_{q_0} for the scalar score gives $(\mu_0+\varepsilon)\sum_{S_{q_0}} a \geq \sum_{S_{q_0}} q_0^\top r \geq \sum_{S^*} q_0^\top r \geq (\mu_0-\varepsilon)\sum_{S^*} a$, so $\sum_{S_{q_0}} a \geq \frac{\mu_0-\varepsilon}{\mu_0+\varepsilon}\sum_{S^*} a \geq \frac{v_{\min}-\varepsilon}{v_{\min}+\varepsilon}\sum_{S^*} a$ (the ratio increases in μ_0). For any S with $|S| = B$ and any k, $\sum_S r_k \leq (v_k+\varepsilon)\sum_S a \leq (v_k+\varepsilon)\sum_{S^*} a$, while $\sum_{S_{q_0}} r_k \geq (v_k-\varepsilon)\sum_{S_{q_0}} a$. Combining, $\sum_{S_{q_0}} r_k \big/ \sum_S r_k \geq \frac{v_k-\varepsilon}{v_k+\varepsilon}\cdot\frac{v_{\min}-\varepsilon}{v_{\min}+\varepsilon} \geq \kappa(\varepsilon)$. For completeness, let z_0 be the first element of S_{q_0}; the same argument with $B = 1$ gives $a(z_0) \geq \frac{v_{\min}-\varepsilon}{v_{\min}+\varepsilon}a(z)$ for every z. For any $q \geq 0$, $q^\top r(z_0) \geq a(z_0)(q^\top v - \varepsilon\|q\|)$ and $q^\top r(z) \leq a(z)(q^\top v + \varepsilon\|q\|)$, so $q^\top r(z) \leq q^\top r(z_0)/\kappa(\varepsilon)$ and $C(z \mid \{z_0\}) \leq 1/\kappa(\varepsilon)$ with unit costs; hence $A(z \mid S_{q_0}) \leq A(z \mid \{z_0\}) \leq 1/\kappa(\varepsilon) - 1 = 4v_{\min}\varepsilon/(v_{\min}-\varepsilon)^2$ by Proposition 2(a).

(c) Take $K = 2$ and $v = (\cos\omega, \sin\omega)$ with $0 < \omega \leq \pi/4$, so $v_{\min} = \sin\omega$. Given $\varepsilon \in (0, v_{\min})$, let $\phi = 2\arcsin(\varepsilon/2)$; then $\phi < \omega$ because $\varepsilon < \sin\omega < 2\sin(\omega/2)$, and the unit directions $u_\pm$ at angles $\omega \mp \phi$ from the first axis satisfy $\|u_\pm - v\| = \varepsilon$; u_+ favors capability 1. The uniform price $q_0 = (1,1)/\sqrt{2}$ scores a unit direction at angle ϑ by $\cos(\vartheta - \pi/4)$. Let $\zeta = \cos(\pi/4-\omega-\phi)/\cos(\pi/4-\omega+\phi) \geq 1$, put two or more copies of $\zeta(1+\xi)u_+$ with $\xi > 0$ and one example u_-, and let $B = 2$. Each copy scores $(1+\xi)\cos(\pi/4-\omega-\phi)$, more than u_-, so the top-2 set consists of two copies and $\sum_{S_{q_0}} r_2 = 2\zeta(1+\xi)\sin(\omega-\phi)$, while one copy and u_- give $\sum_S r_2 = \zeta(1+\xi)\sin(\omega-\phi)+\sin(\omega+\phi)$. As $\xi \to 0$, product-to-sum identities give the loss on capability 2 as $1 - \frac{2\zeta\sin(\omega-\phi)}{\zeta\sin(\omega-\phi)+\sin(\omega+\phi)} = \frac{\sin 2\phi}{\cos 2\phi + \sin 2\omega - \cos 2\omega}$. With $\sin\phi = \varepsilon\cos(\phi/2)$, $\cos 2\phi - \cos 2\omega = 2(\sin^2\omega - \sin^2\phi)$ and $v_{\min} = \sin\omega$, the loss equals $c\,\varepsilon/v_{\min}$ with $c = \cos(\phi/2)\cos\phi\sin\omega/(\sin^2\omega - \sin^2\phi + \sin\omega\cos\omega)$. Since $\sin x + \cos x$ increases on $[0, \pi/4]$ and $\phi < \omega$, $\sin\omega + \cos\omega - \cos(\phi/2)\cos\phi \geq \sin\phi$, so the denominator minus the numerator is at least $\sin\phi(\sin\omega - \sin\phi) \geq 0$ and $c \leq 1$. Since $\sin\omega + \cos\omega \leq \sqrt{2}$, $\sqrt{2}\sin\omega \leq 1$ and $\cos(\phi/2) \geq \cos\phi \geq 0$, $\sqrt{2}\sin\omega\,(1 - \cos(\phi/2)\cos\phi) \leq 1 - \cos^2\phi = \sin^2\phi$, which rearranges to $c \geq 1/\sqrt{2}$. The loss is therefore $\Theta(\varepsilon/v_{\min})$ uniformly in $v_{\min}$, and the factor $1 - 4\varepsilon/v_{\min}$ of (b) is tight up to a constant: the constants 4 and $[1/\sqrt{2}, 1]$ differ by at most a factor $4\sqrt{2} < 5.7$; closing this gap is open. For the coordinate price e_1, the same pool with $\zeta = 1$ gives the loss $\tan\phi/\tan\omega$, between $\varepsilon/(\sqrt{2}\,v_{\min})$ and $\varepsilon/v_{\min}$. Total loss: a unit direction u with $u_k = 0$ satisfies $\|u - v\| \geq |u_k - v_k| = v_k$, with equality impossible because $v - v_k e_k$ is not a unit vector, so a total loss requires $\varepsilon > v_{\min}$. Once $\varepsilon \geq \|e_1 - v\| = 2\sin(\omega/2)$, put two or more copies of $\sqrt{2}\cos(\pi/4-\omega)(1+\xi)e_1$ and one example v: the uniform price prefers the copies, whose top-2 set has $\sum_{S_{q_0}} r_2 = 0$, while one copy and v give $\sum_S r_2 = \sin\omega > 0$. □

A.2 Proof of Proposition 1

Let $X = \{x \in [0,1]^N : \sum_z x_z = B\}$ and $f(x,q) = \sum_k q_k \sum_z x_z \tilde{r}_k(z)$ for $q \in \Delta_K$. For $x \in X$, a linear function of x over X is maximized at the indicator of a B-set, so $\sum_z x_z r_k(z) \le b_k$ and every coverage is at most one. *(i)* The indicator of S lies in X, and its objective $\min_k \sum_{z\in S} \tilde{r}_k(z)$ is $\mathrm{cov}(S)$; maximizing over indicators gives $c_{\mathrm{disc}} \le c^\star$. *(ii)* For fixed x, a linear function of q over the simplex is minimized at a vertex, so $\min_{q\in\Delta_K} f(x,q) = \min_k \sum_z x_z \tilde{r}_k(z)$ and $c^\star = \max_{x\in X} \min_{q\in\Delta_K} f(x,q)$. Both X and Δ_K are nonempty compact convex polytopes and f is bilinear, so by the minimax theorem (equivalently, LP duality) $c^\star = \min_{q\in\Delta_K} \max_{x\in X} f(x,q)$. For fixed q, $\max_{x\in X} \sum_z x_z\, q^\top \tilde{r}(z)$ is a fractional knapsack with unit weights and capacity B, solved by the indicator of the B largest values of $q^\top \tilde{r}$, with value $\mathrm{Top}_B(q^\top \tilde{r})$. *(iii)* $G(q_0) \ge 0$ by (i), and $G(q_0) - (c_{\mathrm{disc}} - \mathrm{cov}(S_{q_0})) = c^\star - c_{\mathrm{disc}}$ by definition. If $r(z) = a(z)v$ with $v > 0$ and $B \le |\{z : a(z) > 0\}|$, then $b_k = v_k A$ with A the sum of the B largest values of a, so $\tilde{r}_k(z) = a(z)/A$ for every k and $\sum_z x_z \tilde{r}_k(z) = \sum_z x_z a(z)/A \le 1$, with equality at the top-B set of a; hence $c^\star = 1$, and by Theorem 1(a) S_{q_0} is that set, so $\mathrm{cov}(S_{q_0}) = c_{\mathrm{disc}} = 1$. In the ε-cone, Theorem 1(b) applied with S the best B-set of each capability gives $\sum_{S_{q_0}} r_k \ge \kappa(\varepsilon) b_k$, so $\mathrm{cov}(S_{q_0}) \ge \kappa(\varepsilon)$, and $c^\star \le 1$; hence $G(q_0) \le 1 - \kappa(\varepsilon)$. □

A.3 Proof of Theorem 2

(i) Problem (2) is the linear program $\min\{c_S^\top w + \rho^\top s :\ R_S^\top w + s \ge r(z),\ w \ge 0,\ s \ge 0\}$. For $\rho < \infty$ it is feasible ($w = 0$, $s = r(z)^+$) and bounded below by 0, so its optimum is finite and attained. Its Lagrange dual with multipliers $q \ge 0$ on the K covering constraints is $\max\{q^\top r(z) : R_S q \le c_S,\ q \le \rho,\ q \ge 0\}$: the constraint $R_S q \le c_S$ comes from $w \ge 0$ and $q \le \rho$ from $s \ge 0$. The dual feasible set is $Q(S)$, which contains $q = 0$ and is compact, so strong duality holds and the dual optimum is attained. For $\rho = \infty$ (s absent) the same argument applies whenever the primal is feasible, and both sides equal $+\infty$ otherwise.

(ii) $A(z \mid S) \le 0$ iff $C(z \mid S) \le c(z)$; since C is the minimum replication cost, this holds iff some feasible (w, s) has cost at most $c(z)$. By (i), $C(z \mid S) \le c(z)$ iff $\max_{q\in Q(S)} q^\top r(z) \le c(z)$, i.e., iff $q^\top r(z) \le c(z)$ for every $q \in Q(S)$.

(iii) Since $S \subseteq D$, $Q(D) \subseteq Q(S)$ always. If S is 0-complete, then by (ii) every $q \in Q(S)$ satisfies $q^\top r(z) \le c(z)$ for all $z \in D$, which together with $0 \le q \le \rho$ is the definition of $Q(D)$; hence $Q(S) \subseteq Q(D)$. Conversely, if $Q(S) = Q(D)$, then for any $z \in D$ and $q \in Q(S) = Q(D)$ we have $q^\top r(z) \le c(z)$, so $A(z \mid S) \le 0$ by (ii). □

Proposition 3 (Regret-minimizing sets)**.** *If $R \ge 0$, $R \ne 0$, $S \ne \emptyset$, all costs equal one and $\rho = \infty$, then $\max_{z\in D} A(z \mid S) = \mathrm{mrr}(S)/(1 - \mathrm{mrr}(S))$ (read as $+\infty$ when $\mathrm{mrr}(S) = 1$), where $\mathrm{mrr}(S) = \sup_q \big(1 - \max_{i\in S} q^\top r(i) / \max_{z\in D} q^\top r(z)\big)$, over $q \ge 0$ with $\max_{z\in D} q^\top r(z) > 0$, is the maximum regret ratio of S over linear capability preferences (Nanongkai et al., 2010). Greedy gap maximization is then the greedy algorithm for regret-minimizing sets, and a finite cap acts as K extra selected examples with payoffs e_k/ρ_k and unit cost.*

Proof. Write $h_S(q) = \max_{i\in S} q^\top r(i) \ge 0$ and $h_D(q) = \max_{z\in D} q^\top r(z) \ge h_S(q)$; since $R \ne 0$, some $q \ge 0$ has $h_D(q) > 0$. Then $Q(S) = \{q \ge 0 :\ h_S(q) \le 1\}$. If $h_S(q) > 0$, the rescaled price $q/h_S(q)$ lies in $Q(S)$, so $C(z \mid S) \ge q^\top r(z)/h_S(q)$; if $h_S(q) = 0 < q^\top r(z)$, every multiple of q lies in $Q(S)$ and $C(z \mid S) = \infty$. Conversely, every $q \in Q(S)$ with $q^\top r(z) > 0$ has $h_S(q) \le 1$ and hence $q^\top r(z) \le q^\top r(z)/h_S(q)$. With $x/0 = \infty$ for $x > 0$, this gives $C(z \mid S) = \max\{0, \sup_{q\ge 0} q^\top r(z)/h_S(q)\}$, the supremum over q with $q^\top r(z) > 0$. Taking the maximum over $z \in D$, and using $h_D \ge h_S$, $\max_z C(z \mid S) = \sup\{h_D(q)/h_S(q) :\ q \ge 0,\ h_D(q) > 0\} = 1/\big(1 - \mathrm{mrr}(S)\big)$, with both sides $+\infty$ exactly when $\mathrm{mrr}(S) = 1$. Hence $\max_z A(z \mid S) = 1/(1 - \mathrm{mrr}(S)) - 1 = \mathrm{mrr}(S)/(1 - \mathrm{mrr}(S))$. With a finite cap, the outside option buys one unit of capability k for ρ_k, exactly as a selected example with payoff e_k/ρ_k and unit cost would. □

A.4 Proof of Proposition 2

(a) Any replication (w, s) feasible for S is feasible for $S' \supseteq S$ after padding w with zeros, so $C(z \mid S') \le C(z \mid S)$. *(b)* Let $\Lambda = \mathrm{diag}(\lambda)$ with $\lambda > 0$ and $\tilde{r} = \Lambda r$, $\tilde{\rho} = \Lambda^{-1}\rho$. The substitution $\tilde{q} = \Lambda^{-1}q$ maps $Q(S)$ bijectively onto $\tilde{Q}(S) = \{\tilde{q} : 0 \le \tilde{q} \le \tilde{\rho},\ \tilde{q}^\top \tilde{r}(i) \le c(i)\}$ and preserves the objective, $\tilde{q}^\top \tilde{r}(z) = q^\top r(z)$; by Theorem 2(i) every replication cost, hence every gap, is unchanged. For $\rho_k = M/\max_z r_k(z)^+$ we have $\tilde{\rho}_k = M/\max_z \lambda_k r_k(z)^+ = \rho_k/\lambda_k$ as required. A fixed price $q_0^\top \Lambda r(z)$ ranks candidates by $\sum_k q_{0k}\lambda_k r_k(z)$, whose order changes with λ whenever two candidates trade off two capabilities. *(c)* Take $w = e_i$ and $s = 0$: $r(i) \ge r(z)$ makes it feasible, with cost $c(i) \le c(z)$. *(d)* Let $q^\star$ attain $C(z^\star \mid S) < \infty$ (always possible for $\rho < \infty$). For every $z \in D$, $q^{\star\top} r(z) - c(z) \le C(z \mid S) - c(z) = A(z \mid S) \le A(z^\star \mid S) = q^{\star\top} r(z^\star) - c(z^\star)$, so $z^\star$ maximizes $q^{\star\top} r(z) - c(z)$ over D; and $q^\star \ne 0$ because $q^{\star\top} r(z^\star) - c(z^\star) > 0 > -c(z^\star)$. In later layers of Algorithm 1 the same argument applies to the residual pool. □

A.5 Proof of Theorem 3

Write $d_w = \sum_{i \in S} w_i\, d(i)$; $\|d(z)\| \le \Gamma$ and $\|d_w\| \le \Gamma \|w\|_1$, where $\|w\|_1 \le c(z)/\min_i c(i)$ for a cost-matched w. By β_k-smoothness, for any direction d, $-\eta\langle \nabla L_k, d\rangle - \frac{\beta_k}{2}\eta^2\|d\|^2 \le L_k(\theta - \eta d) - L_k(\theta) \le -\eta\langle \nabla L_k, d\rangle + \frac{\beta_k}{2}\eta^2\|d\|^2$, and, by (1), $\langle \nabla L_k, d(z)\rangle = r_k(z)$ and $\langle \nabla L_k, d_w\rangle = (R_S^\top w)_k$. For LESS cosines, $d(z) = \nabla\ell(z;\theta)/\|\nabla\ell(z;\theta)\|$ and $L_k(\theta') = \frac{1}{|V_k|}\sum_{j \in V_k} \ell(j;\theta')/\|\nabla\ell(j;\theta)\|$ with the weights frozen at θ, so $\langle \nabla L_k(\theta), d(z)\rangle$ is the average cosine and $\Gamma = 1$.

(a) A replication without the outside option has $R_S^\top w \ge r(z)$. Hence $L_k(\theta_w^+) - L_k(\theta) \le -\eta(R_S^\top w)_k + \frac{\beta_k}{2}\eta^2\Gamma^2\|w\|_1^2 \le -\eta r_k(z) + \frac{\beta_k}{2}\eta^2\Gamma^2\|w\|_1^2$ and $L_k(\theta_z^+) - L_k(\theta) \ge -\eta r_k(z) - \frac{\beta_k}{2}\eta^2\Gamma^2$. Subtracting gives the claim. For $\rho = \infty$, $A(z \mid S) \le 0$ means $C(z \mid S) \le c(z) < \infty$, and an optimal replication, which uses no outside option, is such a w.

(b) $L_{q^\star}$ is $(\sum_k q_k^\star \beta_k)$-smooth and $\langle \nabla L_{q^\star}, d\rangle = \sum_k q_k^\star \langle \nabla L_k, d\rangle$. Because $q^\star \in Q(S)$ (for any cap ρ), $q^{\star\top} R_S^\top w = \sum_i w_i\, q^{\star\top} r(i) \le \sum_i c(i) w_i \le c(z)$, while $q^{\star\top} r(z) = C(z \mid S) = c(z) + \gamma$. Therefore $L_{q^\star}(\theta_z^+) - L_{q^\star}(\theta) \le -\eta(c(z) + \gamma) + \frac{1}{2}\eta^2\Gamma^2 \sum_k q_k^\star \beta_k$ and $L_{q^\star}(\theta_w^+) - L_{q^\star}(\theta) \ge -\eta\, c(z) - \frac{1}{2}\eta^2\Gamma^2\|w\|_1^2 \sum_k q_k^\star \beta_k$; subtracting gives the claim. If $(w^*, 0)$ is an optimal replication, complementary slackness gives $q^{\star\top} r(i) = c(i)$ whenever $w_i^* > 0$, so $q^{\star\top} R_S^\top w^* = c_S^\top w^* = C(z \mid S)$; rescaled to cost $c(z)$, $\tilde{w} = w^* c(z)/C(z \mid S)$ has $q^{\star\top} R_S^\top \tilde{w} = c(z) = q^{\star\top} r(z) - \gamma$, so the first-order bound is attained. □

A.6 Sketch stability

Proposition 4. *Fix a finite cap ρ. If $\|\hat{R} - R\|_{\max} \le \tau$ and the optimal replications for R and $\hat{R}$ have ℓ_1 masses $\|w\|_1$ at most W, then $|\hat{A}(z \mid S) - A(z \mid S)| \le \tau\,\|\rho\|_1(1 + W)$ for all z simultaneously. In particular, all arbitrage decisions with $|A(z \mid S)| > \tau\|\rho\|_1(1 + W)$ are preserved. With cosine payoffs computed from Gaussian Johnson-Lindenstrauss projections of dimension $O(\tau^{-2}\log(N|V|/p_{\mathrm{fail}}))$, where V is the set of validation examples, $\|\hat{R} - R\|_{\max} = O(\tau)$ with probability at least $1 - p_{\mathrm{fail}}$, uniformly over all candidates.*

Proof. Let (w, s) be optimal for R. Then $\hat{R}_S^\top w + s \ge r(z) - \tau\|w\|_1 \mathbf{1} \ge \hat{r}(z) - \tau(1 + \|w\|_1)\mathbf{1}$, so $(w, s + \tau(1 + \|w\|_1)\mathbf{1})$ is feasible for $\hat{R}$ and $\hat{C} \le C + \tau(1 + W)\|\rho\|_1$. The symmetric argument bounds $C - \hat{C}$. The projection statement is the standard inner-product form of the Johnson-Lindenstrauss lemma applied to the $N|V|$ pairs of unit gradients with a union bound, averaged over each capability. When ρ is recomputed from $\hat{R}$ (as $\hat{\rho}_k = M/\hat{m}_k$ with $m_k = \max_z r_k(z)^+$ and $|\hat{m}_k - m_k| \le \tau < m_k$), then $|\hat{\rho}_k - \rho_k| \le M\tau/(m_k(m_k - \tau))$, and the bound increases by at most $\|s\|_1 \max_k M\tau/(m_k(m_k - \tau))$, where s is the outside-option purchase of the optimal replication. □

A.7 NUMERICAL CHECKS

Besides the proofs, every statement was checked numerically. Proposition 1(ii): on the 30 stored payoff matrices of the PACS and OfficeHome settings of Table 6 (three settings, five seeds, class-level and domain-level capabilities, 5% budget), the price recovered from the multipliers of the max-min program attains $c^\star$ to within $5 \cdot 10^{-11}$, and on 200 random signed matrices to within 10^{-15}; no random price ever certified less than $c^\star$. Proposition 1(iii): $c^\star = \mathrm{cov}(S_{q_0}) = 1$ on 200 random rank-one pools, and $G(q_0) \le 1 - \kappa(\varepsilon)$ on all 50 random cone pools; on the real class-level matrices, rounding the fractional solution to its B largest weights attains on average 93% of $c^\star$ on PACS and OfficeHome (at least 62%), but only 48% and 54% on RotatedMNIST at 5 and 10%, where $c^\star \approx 0.005$ is close to zero. Theorem 1(b): the guarantee held in all 100 synthetic runs in which it is non-vacuous (Appendix D); (c): over $4 \cdot 10^5$ random (ω, ε) the constant c of the construction lies in $[0.7071, 1.0000]$, the range proved. Theorem 2(i): on the targeted PACS and OfficeHome matrices (S the first 60 picks of NOARB, 200 random candidates per seed, 2,000 pairs), the dual value equals the value of the primal program solved with HiGHS to a relative $6 \cdot 10^{-7}$. Proposition 3: on 50 random non-negative pools, regret ratios over 20,000 random prices never exceeded the value given by the replication program and came within $4 \cdot 10^{-4}$ of it (median). Theorem 3: Section 4.2. In that configuration the worst-case bound is conservative: bounding β_k by the local curvature of the weighted losses at θ_0, its second-order term is a median 1.8 times γ at $\eta = 10^{-4}$ (32 times with the global smoothness constant), while the measured deviation from the first-order identity is 0.19%. Unit tests check the replication program against a primal solver, the rank-one collapse, unit invariance, the treatment of duplicates, and the lazy and layered greedy against a brute-force greedy.

B EXPERIMENTAL DETAILS

Algorithm. Greedy gap maximization adds $\arg\max_z A(z \mid S)$ until S is complete (Algorithm 1). Gaps only shrink as S grows (Proposition 2(a)), so stale values are valid upper bounds and the lazy evaluation returns exactly the greedy choice; bounds are also tightened after each pick by replication with the new example alone (a closed-form upper bound), and the K-variable dual of the current layer stays in a warm-started simplex solver (Huangfu & Hall, 2018). A budget beyond the completeness point (tolerance 10^{-9}) is spent by starting a new layer on the residual pool (hull peeling), with bounds reset to $A(z \mid \emptyset)$ and gaps computed against that layer only.

Algorithm 1: No-arbitrage selection (NOARB)

1: $S \leftarrow \emptyset, T \leftarrow \emptyset$ (current layer)
2: $b(z) \leftarrow A(z \mid \emptyset) = \rho^\top r(z)^+ - c(z)$ for all z
3: **while** $|S| < B$:
4: pop z with the largest bound $b(z)$
5: **if** $T \neq \emptyset$ and $b(z) \le 0$: $T \leftarrow \emptyset$, reset b; **continue**
6: **if** $b(z)$ is stale: $b(z) \leftarrow A(z \mid T)$ (LP); push back
7: **else**: $S \leftarrow S \cup \{z\}$, $T \leftarrow T \cup \{z\}$; tighten b by the one-example replication bound
8: **return** S (in selection order)

Data. *PACS* (Li et al., 2017) has 9,991 images of 7 classes in 4 domains (art painting, cartoon, photo, sketch). *OfficeHome* (Venkateswara et al., 2017) has 15,588 images of 65 classes in 4 domains (Art, Clipart, Product, Real World). Each domain is split at random, per seed, into 70% candidate pool, 10% capability validation and 20% test; for PACS this gives 6,994 / 998 / 1,999 images. Images are resized to 256 pixels, center-cropped to 224 and encoded once by a frozen ImageNet ResNet-50 (He et al., 2016) (torchvision weights `IMAGENET1K_V2`); features are standardized with pool statistics. *Targeted RotatedMNIST* is built from MNIST (LeCun et al., 1998): the pool contains 250 training digits at each of 24 rotation angles $\{0°, 15°, \ldots, 345°\}$ (6,000 examples), and the $K = 4$ target capabilities are the angles $\{0, t, 2t, 3t\}$ with 100 validation and 1,000 test digits each, drawn from disjoint images; $t = 0$ makes the four targets identical. We follow the rotated-digit construction of Ghifary et al. (2015) and Gulrajani & Lopez-Paz (2021).

Models and training. Vision heads are multinomial logistic regressions on the frozen features with an ℓ_2 penalty of weight 0.1 for PACS and 0.03 for OfficeHome (chosen on validation data with random subsets, never on test data), fitted to convergence with L-BFGS. RotatedMNIST uses a CNN (two 5×5 convolutions with 8 and 16 channels, max pooling, a 64-unit hidden layer; 20,522 parameters) trained with Adam. The warm-up model θ_0 is trained on a random 5% of the pool (200 Adam steps for the CNN). In the *incremental* protocol (main text), the final model is trained on warm-up data plus the selected data, starting from θ_0 (300 Adam steps, learning rate $5 \cdot 10^{-4}$, batch 64, weight decay $5 \cdot 10^{-4}$ for the CNN; the convex head is refitted to convergence). In the *selected-only* protocol (Appendix D), the model is trained on the selected data alone.

Payoffs. All influence-based rules consume the same payoff matrix, so they differ only in how they use it. Following LESS (Xia et al., 2024), $r_k(z)$ is the average over validation examples j of capability k of the cosine between the gradients of z and j at θ_0: head gradients $\tilde{x}(\hat{y} - y)^\top$ for the vision heads, with the features $\tilde{x}$ extended by a constant 1 and predicted class probabilities $\hat{y}$ (computed in closed form) and full-network gradients for the CNN (computed with per-example automatic differentiation). *Fine* capabilities are (domain, class) pairs of the target domains that have validation examples (untargeted PACS 28; targeted PACS 14; targeted OfficeHome 128 to 130, depending on the split) and (target angle, digit) pairs for RotatedMNIST (40); *coarse* capabilities are the target domains or angles. The main text uses fine capabilities, as LESS scores subtasks.

Rules. *Random*; *domain-balanced random* (equal quota per domain, interleaved); *class-stratified random* (equal quota per (domain or angle, class) cell of the pool); in the targeted settings *target-domain random*, which samples among pool examples of the target domains or angles, plainly or with an equal quota per (target, class) cell; *max-min LP*: the LP relaxation $\max_{x \in [0,1]^N,\, \sum x = B} \min_k \sum_z x_z r_k(z) / \max_{z'} r_k(z')^+$ on the fine payoffs, rounded to the B largest x_z (solved separately for each budget); *embedding similarity* (RDS-style): cosine between the candidate's feature vector (frozen ResNet features; penultimate CNN features at θ_0 for RotatedMNIST) and the normalized mean feature of each capability's validation examples, with round-robin over capabilities; *KNN-uniform* (Nayak et al., 2026): round-robin over the individual validation examples of the targets, each adding its most gradient-similar remaining candidate (the instance-level cosines that instance-level BIDS reads); *EL2N* (Paul et al., 2021) (largest $\|\hat{y} - y\|$ at the warm-up model; the original averages early-training checkpoints); *k-center* (Sener & Savarese, 2018) on normalized features (penultimate CNN features for RotatedMNIST); *GRAD-MATCH* (Killamsetty et al., 2021b) (orthogonal matching pursuit with non-negative least-squares refits every 50 atoms, matching the mean validation gradient; head gradients are sketched with a Kronecker Gaussian projection of the features to 512 dimensions for PACS and 128 for OfficeHome, and the CNN uses its full per-example gradients; as for every rule, the selected examples are then trained with uniform weights, and the sketch dimensions and refit interval were set once, not tuned); *GRAD-MATCH with normalized targets*, identical except that the target is the sum over the target domains or angles of each target's mean validation gradient, normalized to unit length; *k-center on payoffs* (targeted RotatedMNIST only): farthest-first traversal on the fine payoff vectors, normalized to unit length or raw; LESS mean and task-wise max (Xia et al., 2024); round-robin over capabilities (Ivison et al., 2025); BIDS (Dai et al., 2025) with task-level columns and, as in the original, instance-level columns. Ablations of our rule: *static gap* ranks once by $A(z \mid \emptyset) = \rho^\top r(z)^+ - c(z)$ and *fixed price* ranks once by $q_0^\top (\rho \odot r)$ with uniform q_0 (both remove replication and iteration and differ only in the positive part), and *centroid reference* iterates but replaces the best cost-matched mixture by the uniform mixture of S (so that the gap is $\sum_k \rho_k (r_k(z) - \bar{r}_k(S))^+$). Our rule uses unit costs and $\rho_k = 10 / \max_z r_k(z)^+$, fixed before any result; its main design choice is the capability granularity.

Mechanism test in the selection configuration (Section 4.2). Per seed we refit the targeted-PACS warm-up head (the stored payoffs are reproduced to a relative 10^{-3}), take S as the first 60 picks of NoArb and 400 random other candidates, and compute $A(z \mid S)$, $q^\star$ and an optimal replication w with the cap of the selection runs ($M = 10$). Steps move the head along the normalized direction $d(z)$ of the candidate, or along $\sum_i w_i d(i)$ with w rescaled to cost $c(z)$ when $C(z \mid S) > c(z)$, and the losses L_k weight each validation example of capability k by the inverse norm of its gradient at θ_0, so that the cosine payoffs are exact first-order improvements (Section 3). One step uses $\eta = 10^{-4}$; the horizon sweep repeats steps of size 10^{-2}, recomputing the directions at the current point, for $H \in \{1, 3, 10, 30, 100\}$. A replicable candidate counts as matched if the mixture's loss decrease

is at least that of the candidate on every capability, up to 5% of the candidate's largest first-order improvement. The training distance is $\|\theta - \theta_0\|$ after refitting the head on warm-up plus the 10% selection of NOARB.

Coverage, the scalarization gap and the rule sets. For a budget B, b_k is the sum of the B largest payoffs on capability k, the coverage of a set is $\min_k \sum_{z \in S} r_k(z)/b_k$ over the capabilities with $b_k > 0$, and $c^\star$ is the value of the max-min program of Proposition 1, solved with HiGHS; its rounding keeps the B largest fractional weights. The coverage analysis of Section 4.3 uses the 20 rules whose selection orders were stored with the trained runs (Table 7): random and domain-balanced random sampling, EL2N, k-center, GRAD-MATCH and KNN-uniform (*other*); LESS mean and task-wise max and the static and fixed-price ablations (*scalar scores*); round-robin, BIDS with capability- and instance-level columns, NOARB and its centroid ablation (*capability-aware*); and the domain-level variants of LESS mean and max, round-robin, BIDS and NOARB. Max-min LP, class-stratified random sampling and embedding similarity, whose orders were not stored, are reproduced from the stored payoffs, splits and seeds where they were trained (targeted PACS and OfficeHome, untargeted PACS); normalized GRAD-MATCH is left out because its sketched gradients were not stored. Spearman correlations are computed on the means of the rules over seeds; their 95% intervals resample the 5 seeds with replacement (2,000 draws). The two gradient distances in the spirit of Nayak et al. (2026) use the instance-level cosines at θ_0: *NN* averages over the target validation examples the largest cosine to the selected set, and *MMD* is the squared distance between the mean normalized gradients of the selected set and of the target validation examples (lower is closer; we correlate its negative). For targeted RotatedMNIST, whose orders were not stored, every rule's order is reproduced from the same data, warm-up model, payoffs and seeds and checked against the fingerprints stored with the trained runs (class and target-angle counts, the on-target share, the stored KNN order): all 52 fingerprinted selections match in every seed (normalized GRAD-MATCH, not among the 20 rules, has no fingerprint). Along the interpolation of Section 4.1 the selections are recomputed from the same data, warm-up model and R_α; their overlaps with the selection of mean influence agree with those of the stored training runs in all 175 (seed, α, rule) cases.

Payoff drift along the training run. The drift analysis of Section 4.2 takes the run that trains on warm-up data plus the 10% selection of NOARB. For the convex heads the checkpoints are the points $\theta_0 + t(\theta_{\text{final}} - \theta_0)$, $t \in \{0.01, 0.03, 0.1, 0.3, 1\}$, on the segment to the refitted head; for the CNN they are the Adam iterates of the stored run itself (replayed with the same seeds and batches; the replay reproduces the stored final weights bit for bit) after 1, 3, 10, 30, 100 and 300 steps. At each checkpoint the LESS-style payoffs are recomputed with the same capabilities, and we report the Spearman correlation with the θ_0 payoffs of the mean payoff over the pool, the overlap of the top-10% sets of mean influence, the average per-capability Spearman correlation, and the Spearman correlation of the gaps $A(z \mid S_0)$ of 400 random candidates, S_0 the first 60 picks of NOARB.

Statistics and compute. A seed controls the split, the warm-up subset, the random baselines and the initialization, and all rules share them within a seed. Every setting uses seeds 0 to 4, except the angle sweep of Figure 6 (seeds 0 and 1, and 0 to 4 at $t = 30°$), the synthetic cones (10 seeds) and the instruction-tuning geometry (one sample). We report means with 95% t-intervals and two-sided paired t-tests across seeds, Holm-corrected over all rules of a column (the same family in the main and the appendix tables); with five seeds and up to 22 comparisons per column, differences of a few points between the stronger rules are rarely significant after correction, so Table 16 also gives unadjusted p-values. All selection and training runs used a laptop CPU (Intel Core i9-9880H, 8 cores, shared with other jobs); feature extraction used a 4 GB AMD Radeon Pro 5500M through DirectML. *Layers of Algorithm 1.* The first layer (the completeness point) is reached after 543 to 667 picks on targeted PACS and after 1,018 to 1,062 picks on targeted OfficeHome in three of five seeds (in the other two not within its 1,091 picks), and not within the largest budget on untargeted PACS (1,398 picks) or targeted RotatedMNIST (600 picks), in every seed. Hence all picks at budgets up to 5% lie in the first layer, where Proposition 2 applies to the gap with respect to the whole selected set; at a 10% budget 5 to 22% of the picks on targeted PACS and 0 to 7% on OfficeHome, and at 20% 52 to 61% on targeted PACS, come from later layers, whose gaps are computed against their own layer. A run solves 16,000 to 31,000 small linear programs. The cost of NOARB grows with the number of capabilities: the median run over seeds takes 80 s on targeted PACS ($K = 14$), 343 s on

untargeted PACS ($K = 28$) and 2,269 s on targeted OfficeHome ($K \approx 130$), against 700, 394 and 1,401 s for GRAD-MATCH on the same pools (the machine was shared, so timings are indicative).

C NOTATION

Table 4 collects the notation of the paper.

Table 4: Notation.

Symbol	Meaning
$D = \{z_1, \dots, z_N\}$, N	candidate pool and its size
$S \subseteq D$, B	selected set and budget (number of selected examples)
K, L_k	number of capabilities; validation loss of capability k
θ, θ_0, η, P	parameters, warm-up parameters, step size, optimizer preconditioner
$\ell(z;\theta)$, $d(z)$	training loss of example z; its step direction (normalized gradient for LESS cosines)
$r(z) \in \mathbb{R}^K$, $R \in \mathbb{R}^{N\times K}$	payoff (first-order capability improvements) of z; payoff matrix
$c(z)$	cost of selecting z (one unit in all experiments)
w, s	replication weights on selected examples; outside-option purchase
ρ, M	price cap per capability, $\rho_k = M/\max_z r_k(z)^+$; cap multiplier ($M = 10$)
$C(z \mid S)$, $A(z \mid S)$	replication cost; arbitrage gap $C(z \mid S) - c(z)$
$Q(S)$, q, $q^\star$	supporting capability prices of S; a price; an optimal (scarcity) price
δ	completeness tolerance (S is δ-complete if $\max_z A(z \mid S) \le \delta$)
$\mathrm{mrr}(S)$	maximum regret ratio of S over linear capability preferences
q_0, S_{q_0}	a fixed (exogenous) price used by a scalar rule; its top-B set
$a(z)$, $u(z)$	magnitude and unit direction of a payoff, $r(z) = a(z)u(z)$
v, $v_{\min}$, ε	central unit direction of the payoff cone, its smallest coordinate, cone radius
ε-cone	payoffs $r(z) = a(z)u(z)$ with $a(z) > 0$ and $\|u(z) - v\| \le \varepsilon$ (Theorem 1(b))
$\kappa(\varepsilon)$	sufficiency factor $((v_{\min} - \varepsilon)/(v_{\min} + \varepsilon))^2$ of Theorem 1
b_k, $\mathrm{cov}(S)$	sum of the B largest payoffs on capability k; worst-capability coverage $\min_k \sum_{z\in S} r_k(z)/b_k$
x, $c^\star$, c_{disc}	fractional selection in $[0,1]^N$; fractional coverage ceiling; best coverage of a set of size B (Proposition 1)
$G(q_0)$	scalarization gap $c^\star - \mathrm{cov}(S_{q_0})$ of a fixed price q_0
Δ_K, Top_B, $\tilde{r}$	probability simplex; sum of the B largest values over the pool; payoffs divided by b_k
β_k, Γ	smoothness constant of L_k; bound on the norm of the step directions
e_k, x^+	k-th standard basis vector; positive part $\max(x, 0)$, coordinatewise
L_q	price-weighted loss $\sum_k q_k L_k$
γ	value of a positive gap in Theorem 3
H, $H\eta$	number of steps and path length in the horizon test (Section 4.2)
λ_k, Λ, λ	rescaling of the units of capability k; $\mathrm{diag}(\lambda)$; factor applied to one target's loss in Section 4.4
R_α, R_0, α	interpolated payoffs $(1-\alpha)R_0 + \alpha R$; rank-one payoffs that give every capability the example's mean payoff; interpolation weight
σ_1, σ_2	two largest singular values of a payoff matrix; σ_2/σ_1 measures its spread
$b(z)$, T	upper bound on the gap of z and current layer in Algorithm 1
t	spread of the target angles $\{0, t, 2t, 3t\}$ in targeted RotatedMNIST
ψ	half-angle of the synthetic payoff cones (Appendix D)
τ, p_{fail}, W	sketch error, failure probability and replication mass in Proposition 4

D ADDITIONAL RESULTS

Protocol decisions taken after a pilot. Before any real-data result we fixed the primary capability granularity ((domain, class) pairs, with domains as the secondary granularity), the budgets, the worst-target metric, the baselines of the original plan and the constants of NOARB ($c = 1$, $M = 10$). Two decisions were changed after pilot runs and apply to every rule. (1) *Training from the warm-up model.* Training a model from scratch on the selected data alone makes every influence-based rule collapse: on untargeted RotatedMNIST ($t = 30°$, one seed) with domain-level capabilities, the

influence rules obtained 9 to 23% worst-domain accuracy at 5 and 10% budgets (BIDS with instance-level columns 44 to 54%) against 72 to 76% for random selection, because influence computed at a weak warm-up model concentrates the budget on a few classes. First-order payoffs describe improvements *from* the warm-up model, so the main text continues training from it on warm-up plus selected data; under this protocol the same pilot still favors random sampling (81.9 and 83.3% against 77.6 and 76.7% for NoArb). The selected-only results for PACS are in Table 12; there, NoArb is far above GRAD-MATCH at small budgets ($+26.2 \pm 10.1$ points at 2%, $p = 0.002$; $+12.8 \pm 10.5$ at 5%, unadjusted $p = 0.028$, not significant after Holm correction). (2) *Targeted design.* When the pool follows the target distribution, random selection is hard to beat (Section 4.5); influence-based selection is meant for broad pools and specific targets, as in instruction tuning, so we added the targeted RotatedMNIST, PACS and OfficeHome settings. The target domains were fixed before any targeted result was seen, and the held-out pairs of Section 4.5 were run after all other results without changing the protocol.

Synthetic cones. We draw 3,000 payoffs in $K = 4$ capabilities whose directions fill a cone of half-angle ψ around the diagonal, with log-normal magnitudes, and select 60 (10 seeds). Worst-capability coverage is unit-free, and Theorem 1(b) guarantees $\kappa(\varepsilon)$ to every fixed price. At $\psi = 0$ every rule returns the same set, and the guarantee holds in all 100 runs in which it is non-vacuous (Figure 4). With equal units the rules stay close up to 30°; at wider angles mean influence covers the worst capability best (43% at 90°, NoArb 35%), since in a symmetric cone the uniform price is right. Rescaling the same draws to units $(4, 1, 1, \frac{1}{4})$ shrinks $v_{\min}$ 8.5-fold and leaves the unit-free rules (normalized mean, i.e., the fixed-price ablation, round-robin, BIDS and NoArb) unchanged, while the unnormalized scalar rules collapse (at 90° mean influence covers 6% and task-wise max 1%). Normalizing each capability thus repairs units for a fixed price.

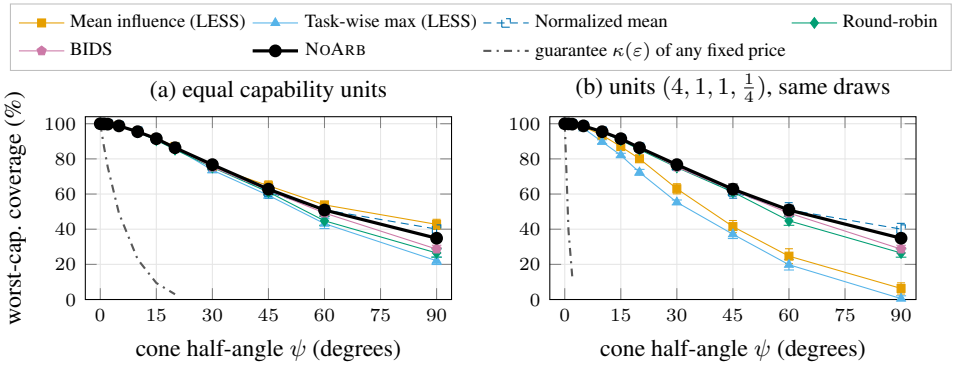

Figure 4: Synthetic payoff cones (same draws in both panels, 10 seeds, bars: 95% t-intervals): worst-capability coverage of each rule; dash-dotted: the guarantee $\kappa(\varepsilon)$ of Theorem 1(b), where it is non-vacuous.

The gap with raw payoffs. Theorem 3 also holds for raw influence payoffs. We take the untargeted PACS linear head at θ_0, a random set S of 40 examples and 400 random candidates per seed (2,000 in total), with raw first-order payoffs $r_k(z) = \langle \nabla L_k, \nabla \ell(z) \rangle$ on the four domains and $M = 10^3$, so that no computed replication uses the outside option, and compare one step on z with one on its cost-matched mixture. The two sides of the identity agree with correlation 1.000 and slope 0.996 at $\eta = 10^{-5}$ (median relative error 0.3% on the 251 candidates with a positive gap; Figure 5a), 0.81 at 10^{-3}, and not at 10^{-2}. For $\eta \leq 10^{-4}$ the replicating mixture matches every replicable candidate with an above-median payoff on all four domains, up to 5% of its largest first-order improvement (90% at 10^{-3}), while the selected example closest to dominating it matches 44% and a random mixture almost none (Figure 5b): replicability is a property of mixtures, not of nearest neighbors.

Payoff drift along the training run. Table 5 gives the drift analysis of Section 4.2 (protocol in Appendix B). On the convex heads every ranking stays above 0.96 up to 30% of the distance and between 0.71 and 0.93 at the refitted head. On the CNN the ranking of mean influence is the least stable quantity: after one Adam step its top-10% set keeps 37% of its examples (18 to 55% across

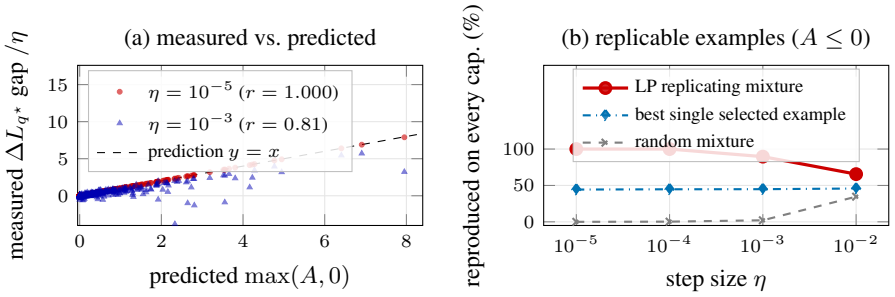

Figure 5: The gap with raw payoffs (untargeted PACS head, 5 seeds, 2,000 candidates, all used in the statistics; plotted: the 251 with a positive gap and 150 replicable ones). (a) Measured price-weighted loss difference between a step on the candidate and on its cost-matched replicating mixture, against $\max(A, 0)$. (b) Share of replicable candidates (above-median payoff) matched on every capability, up to 5% of their largest first-order improvement, by a step of the same cost on each comparator.

seeds), while the per-capability payoffs keep a Spearman correlation of 0.93 and the gaps 0.89; at the end of the run the gaps keep 0.35.

Table 5: Payoff drift along the actual training run on warm-up data plus the 10% selection of NoArb (5 seeds, means): Spearman correlation with the payoffs at θ_0 of the mean payoff over the pool, overlap of the top-10% sets of mean influence, average per-capability Spearman correlation, and Spearman correlation of the gaps of 400 random candidates. Convex heads: points $\theta_0 + t(\theta_{\text{final}} - \theta_0)$; CNN: Adam iterates of the stored run.

Model	checkpoint	distance share	mean payoff	top-10% set	per capability	gaps
PACS head	$t = 0.01$	0.010	1.00	1.00	1.00	1.00
	$t = 0.03$	0.030	1.00	1.00	1.00	1.00
	$t = 0.1$	0.100	1.00	0.99	1.00	1.00
	$t = 0.3$	0.300	0.99	0.98	0.98	0.99
	$t = 1$	1.000	0.90	0.93	0.86	0.93
OfficeHome head	$t = 0.01$	0.010	1.00	1.00	1.00	1.00
	$t = 0.03$	0.030	1.00	1.00	1.00	1.00
	$t = 0.1$	0.100	1.00	0.99	0.99	1.00
	$t = 0.3$	0.300	0.99	0.96	0.96	0.99
	$t = 1$	1.000	0.93	0.88	0.71	0.93
RotatedMNIST CNN	step 1	0.027	0.64	0.37	0.93	0.89
	step 3	0.059	0.42	0.30	0.87	0.77
	step 10	0.144	0.67	0.46	0.91	0.76
	step 30	0.312	0.61	0.42	0.86	0.59
	step 100	0.573	0.50	0.40	0.75	0.39
	step 300	1.000	0.48	0.42	0.67	0.35

Payoff geometry of the real pools. Table 6 reports the quantities of Theorem 1 and Proposition 1 for the real payoff matrices. All of them are far from rank one: $\sigma_2/\sigma_1 \geq 0.50$, $\varepsilon/v_{\min}$ between 1.2 and 34, and the average payoff magnitude differs across capabilities by a factor of up to 8; on targeted RotatedMNIST, 35 to 43% of the candidates have a negative mean payoff. The guarantee $\kappa(\varepsilon)$ is therefore vacuous in every real setting. Along the interpolation of Figure 2, $\varepsilon/v_{\min}$ is 0 at $\alpha = 0$ and already 6.0 at $\alpha = 0.05$, where 63% of the candidates hurt some capability. The ceiling $c^\star$ is at most 0.11 at class level (0.16 on the held-out pairs) and 0.52 to 0.82 at domain level, and the coverage of the set chosen by mean influence is negative at class level in every seed and benchmark (at 5% as well: -0.18 to -0.45 on average).

The coverage analysis: rules and robustness. Table 7 lists the 20 rules of the coverage analysis with their coverage and accuracy on targeted PACS and OfficeHome at a 10% budget, and three rules outside the set whose selections we reproduced. Table 8 reports, for every PACS and OfficeHome

Table 6: Geometry of the real payoff matrices (mean over 5 seeds; K can differ between splits). Eff. rank: $(\sum_i \sigma_i)^2 / \sum_i \sigma_i^2$. Scale ratio: largest over smallest mean absolute payoff across capabilities. v is the normalized mean direction and ε the largest distance of a unit payoff direction from it, computed on the candidates whose payoffs are positive on every capability, or on the payoffs clipped at zero when there are none (all fine-capability pools; every such pool contains candidates that hurt some capability, which already violates the hypothesis of Theorem 1(b)). $c^\star$ and $\mathrm{cov}(S_{q_0})$: coverage ceiling and coverage of mean influence (Proposition 1) at a 10% budget, on the matrix of the row.

Pool	capabilities	K	eff. rank	σ_2/σ_1	$\varepsilon/v_{\min}$	scale ratio	$c^\star$	$\mathrm{cov}(S_{q_0})$
RotatedMNIST, targeted	fine	40	9.1	0.84	9.5	1.8	0.005	-0.37
RotatedMNIST, targeted	coarse	4	3.1	0.57	2.7	1.7	0.720	$+0.52$
PACS, all domains	fine	28	18.1	0.85	16.3	4.9	0.051	-0.16
PACS, all domains	coarse	4	3.1	0.50	2.8	2.2	0.522	$+0.11$
PACS, targeted	fine	14	10.3	0.83	8.9	3.9	0.060	-0.24
PACS, targeted	coarse	2	1.9	0.61	1.2	1.4	0.700	$+0.48$
OfficeHome, targeted	fine	129	85.8	0.82	34.3	7.7	0.108	-0.39
OfficeHome, targeted	coarse	2	1.8	0.50	1.5	1.6	0.824	$+0.73$

setting and for targeted RotatedMNIST, Spearman's correlation between coverage and accuracy on the 20 rules, without the domain-level variants, and on the 11 rules that avoid scalar scores and domain-level capabilities (95% seed-bootstrap intervals), and the correlations of the two gradient distances to the targets. On PACS and OfficeHome the correlation is high when the failing families are included and weak among the rules that avoid failure, and MMD ranks accuracy better than coverage in every setting and budget, also among the rules that avoid failure (0.47 to 0.87): first-order coverage identifies failure regimes rather than ordering good selectors. On RotatedMNIST, where the four targets are a sixth of the pool, coverage also orders the rules that avoid failure.

Table 7: The 20 rules of the coverage analysis (Figure 3c) and three reproduced rules outside the set: worst-capability coverage on the class-level payoffs and worst-target accuracy (%), targeted PACS and OfficeHome, 10% budget, means over 5 seeds.

		PACS, targeted		OfficeHome, targeted	
Family	Rule	coverage	acc.	coverage	acc.
scalar score	LESS, mean	-0.239	69.8	-0.388	46.2
	LESS, task-wise max	-0.331	69.3	-0.349	46.7
	NoArb w/o replication (static)	-0.207	73.9	-0.376	49.4
	NoArb w/ fixed price	-0.149	71.2	-0.234	49.6
domain-level	LESS, mean	-0.305	67.4	-0.385	45.6
	LESS, task-wise max	-0.335	67.1	-0.336	46.0
	Round-robin	-0.232	69.6	-0.360	46.2
	BIDS	-0.239	69.2	-0.358	46.1
	NoArb	-0.238	69.5	-0.377	45.9
capability-aware	Round-robin	-0.043	73.5	-0.015	55.4
	BIDS	-0.062	69.3	-0.101	53.0
	BIDS, instance-level	-0.096	74.0	-0.077	56.8
	NoArb	-0.022	75.4	-0.094	54.0
	NoArb w/ centroid reference	-0.062	75.1	-0.289	49.4
other	Random	-0.050	78.0	-0.075	56.2
	Domain-balanced random	-0.054	76.3	-0.054	54.4
	EL2N	-0.108	69.1	-0.244	51.0
	k-center	-0.043	73.5	-0.046	55.5
	GRAD-MATCH	-0.038	79.1	-0.061	59.8
	KNN-uniform	-0.064	78.9	-0.061	58.6
not in the 20	Max-min LP	$+0.046$	72.7	$+0.052$	53.2
	Class-stratified random	-0.016	75.3	-0.053	55.8
	Embedding similarity (RDS)	-0.082	75.9	-0.103	55.5

Table 8: Spearman correlation between first-order coverage and worst-target accuracy across rules (means over 5 seeds; 95% seed-bootstrap intervals), and between accuracy and two gradient similarities of the selection to the target validation examples (NN: mean largest cosine; MMD: negative squared distance between mean normalized gradients) on the 20 rules and, for MMD, on the 11 non-failing ones; the similarities were not computed for the reproduced RotatedMNIST selections (n/a).

		coverage			gradient distance to the targets		
Pool	budget	20 rules	w/o domain-level	11 non-failing	NN 20 rules	MMD 20 rules	MMD 11 non-failing
PACS, targeted	5%	0.71 [0.51, 0.82]	0.62 [0.35, 0.78]	0.42 [-0.01, 0.59]	0.76	0.81	0.75 [0.46, 0.83]
	10%	0.74 [0.65, 0.81]	0.56 [0.41, 0.71]	0.36 [0.01, 0.46]	0.71	0.84	0.87 [0.80, 0.92]
OfficeHome, targeted	5%	0.83 [0.71, 0.89]	0.76 [0.70, 0.84]	0.45 [0.34, 0.64]	0.86	0.94	0.81 [0.62, 0.87]
	10%	0.88 [0.74, 0.92]	0.84 [0.74, 0.89]	0.60 [0.40, 0.77]	0.78	0.96	0.83 [0.76, 0.91]
PACS, untargeted	5%	0.68 [0.41, 0.82]	0.67 [0.30, 0.78]	0.50 [0.05, 0.63]	0.79	0.73	0.55 [0.35, 0.57]
	10%	0.49 [0.35, 0.68]	0.49 [0.41, 0.69]	0.29 [0.17, 0.55]	0.79	0.75	0.75 [0.64, 0.80]
PACS A+C	5%	0.66 [0.51, 0.83]	0.55 [0.35, 0.70]	0.24 [-0.15, 0.46]	0.82	0.79	0.55 [0.35, 0.73]
	10%	0.72 [0.60, 0.83]	0.44 [0.29, 0.64]	0.03 [-0.21, 0.45]	0.77	0.82	0.70 [0.37, 0.74]
PACS A+P	5%	0.78 [0.73, 0.83]	0.64 [0.50, 0.76]	0.14 [-0.20, 0.46]	0.75	0.87	0.47 [0.32, 0.67]
	10%	0.80 [0.68, 0.88]	0.64 [0.48, 0.76]	0.12 [-0.25, 0.42]	0.69	0.86	0.55 [0.28, 0.73]
PACS A+S	5%	0.57 [0.41, 0.59]	0.43 [0.21, 0.44]	0.03 [-0.32, 0.18]	0.76	0.78	0.72 [0.51, 0.79]
	10%	0.73 [0.55, 0.76]	0.58 [0.29, 0.62]	0.25 [-0.16, 0.37]	0.87	0.88	0.83 [0.70, 0.92]
PACS C+P	5%	0.70 [0.62, 0.76]	0.62 [0.49, 0.71]	0.25 [-0.06, 0.48]	0.88	0.80	0.47 [0.33, 0.62]
	10%	0.72 [0.60, 0.78]	0.63 [0.51, 0.67]	0.29 [0.05, 0.46]	0.76	0.75	0.57 [0.43, 0.63]
PACS P+S	5%	0.60 [0.47, 0.67]	0.47 [0.25, 0.51]	0.25 [-0.16, 0.30]	0.84	0.82	0.81 [0.57, 0.83]
	10%	0.68 [0.55, 0.71]	0.45 [0.35, 0.51]	0.27 [-0.05, 0.36]	0.80	0.88	0.81 [0.68, 0.90]
RotatedMNIST, targeted	5%	0.92 [0.81, 0.94]	0.92 [0.86, 0.96]	0.88 [0.74, 0.96]	n/a	n/a	n/a
	10%	0.91 [0.79, 0.94]	0.95 [0.81, 0.96]	0.91 [0.58, 0.92]	n/a	n/a	n/a

Capability granularity. Table 9 repeats the selection benchmarks with domain-level (coarse) capabilities. Every capability-aware rule, ours included, then loses most of its advantage and falls to about the level of scalar influence: a domain-level payoff averages over classes, so a domain-level rule cannot tell which classes the selection still lacks (on targeted RotatedMNIST the coarse rules concentrate on a few digits, most often 1). Payoff-space replication can only see what the capabilities resolve, so subtask-level capabilities are necessary; the scalar rules change much less with the choice.

Table 9: Capability granularity: worst-target accuracy (%) at a 10% budget with (domain or angle, class) capabilities (fine) and with domain-level capabilities (coarse); mean over seeds.

	RotatedMNIST (targeted)		PACS (targeted)		PACS		OfficeHome (targeted)	
Rule	fine	coarse	fine	coarse	fine	coarse	fine	coarse
NoArb (ours)	70.0	50.6	75.4	69.5	75.8	70.6	54.0	45.9
Round-robin	67.3	50.1	73.5	69.6	73.5	71.0	55.4	46.2
BIDS	64.7	51.1	69.3	69.2	69.7	70.7	53.0	46.1
LESS, mean	50.2	48.3	69.8	67.4	70.2	70.3	46.2	45.6
LESS, task-wise max	52.3	49.1	69.3	67.1	70.5	70.9	46.7	46.0

Held-out target pairs. Table 10 reports the five PACS target pairs that were run after all other results, with the protocol unchanged (same splits, warm-up models, payoffs, rules, budgets and training; 5 seeds), next to the original pair. Random sampling does not depend on the targets, so it coincides across pairs with the same worst target domain (sketch). At 2% and 20% budgets as well, the most accurate rule on every held-out pair is GRAD-MATCH, its normalized variant or KNN-uniform.

Same influence, different replicability. This test isolates the gap from influence (Section 4.4). From the warm-up set, we greedily pair the most influential positive-gap candidates of the more influential half of untargeted PACS (300 per seed) with replicable candidates of nearly the same mean influence, optionally of the same domain and class (Table 13). Adding the positive-gap members instead of their partners raises macro accuracy by 0.94 ± 0.37 points ($p = 0.002$, 5 seeds), 1.23 ± 0.34 when also matched on domain ($p < 0.001$) and 0.75 ± 0.80 when matched on domain and class

Table 10: Held-out PACS target pairs (A: art painting, C: cartoon, P: photo, S: sketch; C+S is the original pair of Table 3): worst-target accuracy (%, mean over 5 seeds). Bold: best rule that does not know the targets; underline: best on the shared payoffs. Last row: Spearman correlation between first-order coverage and accuracy over the 20 rules of Section 4.3.

	5% budget						10% budget					
Rule	A+C	A+P	A+S	C+P	P+S	C+S	A+C	A+P	A+S	C+P	P+S	C+S
Random	77.0	83.6	74.9	77.0	74.9	74.9	79.6	85.6	78.5	79.6	78.5	78.0
Target-domain random	81.0	86.9	74.4	79.6	74.2	74.2	83.1	89.5	76.7	82.9	76.3	76.8
KNN-uniform	80.8	87.5	75.7	**82.6**	76.6	74.5	**83.0**	**89.6**	**79.2**	**84.3**	79.0	78.9
GRAD-MATCH	80.9	**87.8**	76.5	81.3	**78.1**	77.1	82.8	88.7	78.6	82.8	79.3	**79.1**
GRAD-MATCH, normalized	**81.6**	87.4	**76.9**	81.8	77.7	**77.8**	82.6	87.9	79.0	83.2	**79.7**	78.8
LESS, mean	74.1	79.7	71.5	73.7	71.0	70.6	74.8	80.1	71.0	74.5	71.2	69.8
LESS, task-wise max	74.2	80.0	70.9	73.6	70.9	70.3	76.5	79.8	70.4	74.2	71.2	69.3
Round-robin	78.7	85.3	73.5	78.3	73.3	73.5	80.8	87.1	73.8	80.3	73.7	73.5
BIDS	78.9	83.0	69.9	77.7	70.4	70.2	81.2	84.9	70.5	79.4	69.7	69.3
BIDS, instance-level	79.6	86.2	73.1	79.8	73.2	72.7	81.8	88.6	73.6	81.4	74.3	74.0
NoArb	79.1	86.1	74.8	79.1	75.8	74.6	80.4	87.6	76.6	80.9	76.6	75.4
w/o replication (static)	76.3	81.7	71.9	76.5	73.2	72.2	78.9	83.4	73.0	79.5	74.5	73.9
w/ fixed price	75.0	81.1	71.7	74.3	71.4	71.1	76.2	82.2	72.4	75.9	71.2	71.2
w/ centroid reference	78.1	82.3	73.8	77.7	74.0	73.8	80.4	83.7	76.3	79.6	76.0	75.1
ρ_s(coverage, accuracy)	0.66	0.78	0.57	0.70	0.60	0.71	0.72	0.80	0.73	0.72	0.68	0.74

Table 11: Untargeted PACS (all four domains are pool and targets), incremental protocol: worst-domain accuracy (%, 5 seeds). Grouping, bold, underline, † and ‡ as in Table 3.

Rule	2%	5%	10%	20%
no payoffs				
Random	72.7 $\pm$1.3	74.9 $\pm$2.4	78.0 $\pm$3.0	79.5 $\pm$2.7
Domain-balanced random	71.6 $\pm$3.1	74.3 $\pm$2.1	76.3 $\pm$2.6	78.3 $\pm$2.2
Class-stratified random	72.2 $\pm$2.1	73.6 $\pm$3.1	75.3 $\pm$3.0	77.0 $\pm$3.5
EL2N	71.0 $\pm$2.0†	68.7 $\pm$4.4	69.1 $\pm$3.5	75.0 $\pm$1.2
features or full gradients				
k-center	71.0 $\pm$2.4	71.9 $\pm$2.8	73.5 $\pm$3.2	75.5 $\pm$3.2
Embedding similarity (RDS)	72.3 $\pm$2.5	73.3 $\pm$2.9	74.2 $\pm$3.8	75.5 $\pm$3.2
KNN-uniform (gradient cosines)	70.1 $\pm$2.2	70.6 $\pm$3.0	74.3 $\pm$3.2	78.9 $\pm$2.2
GRAD-MATCH	**74.5** $\pm$2.7	**76.8** $\pm$1.6	**78.8** $\pm$2.6	**80.8** $\pm$2.2
GRAD-MATCH, normalized targets	73.9 $\pm$2.9	76.4 $\pm$3.1	78.1 $\pm$1.0	80.5 $\pm$2.0
influence payoffs R (shared input)				
LESS, mean	70.1 $\pm$2.3	70.3 $\pm$2.3	70.2 $\pm$2.3	70.5 $\pm$2.5
LESS, task-wise max	70.3 $\pm$2.6	70.9 $\pm$2.4	70.5 $\pm$2.5	68.7 $\pm$3.1
Max-min LP	72.9 $\pm$1.7	72.6 $\pm$1.9	72.3 $\pm$2.7	71.7 $\pm$3.0†
Round-robin	72.4 $\pm$3.1	72.8 $\pm$3.2	73.5 $\pm$3.6	73.2 $\pm$3.5
BIDS	71.2 $\pm$2.1	70.5 $\pm$2.8	69.7 $\pm$2.2	69.0 $\pm$3.2
BIDS, instance-level	70.9 $\pm$2.8	71.7 $\pm$3.3	72.0 $\pm$4.1	73.1 $\pm$3.6
NoArb (ours)	73.5 $\pm$2.9	74.7 $\pm$3.9	75.8 $\pm$3.8	76.8 $\pm$3.9
w/o replication (static)	70.7 $\pm$3.8	71.1 $\pm$3.6	71.3 $\pm$3.0	71.9 $\pm$2.5
w/ fixed price	70.1 $\pm$2.3	70.2 $\pm$2.4	70.5 $\pm$2.7	71.2 $\pm$2.0
w/ centroid reference	71.1 $\pm$3.8	71.9 $\pm$3.1	71.7 $\pm$2.8	72.4 $\pm$3.0

($p = 0.06$), so part of what replicability detects is uncovered classes. The positive-gap members have 6 to 9% more influence, which does not favor them (the most influential candidates do worse, 80.1 against 80.9), and random candidates from the same half beat both (82.5, $p \leq 0.02$): the gap separates candidates of equal influence, but in this untargeted pool the most influential ones are not the ones to select (Section 4.5). With ten near-copies of each of the 40 most influential PACS examples added to the pool, mean influence spends 91% of its budget on them and task-wise max 44%, against 2% for NoArb, 1% for BIDS and 5% for random sampling (Table 14), as Proposition 2(c) suggests; the copies cost every rule at most 0.4 points of worst-domain accuracy, so this checks a selection property, not a learning benefit.

Table 12: Untargeted PACS, selected-only protocol (the head is trained on the selected data alone): worst-domain accuracy (%, 5 seeds). Grouping, bold, underline, † and ‡ as in Table 3. Scalar influence and BIDS collapse onto a few classes; the static ablation costs NOARB 19 to 36 points, the fixed price 47 to 50 and the centroid reference 17 to 26; random sampling is ahead of NOARB from a 5% budget on.

Rule	2%	5%	10%	20%
no payoffs				
Random	56.8 ±3.1	**68.5** ±3.7	**74.6** ±4.6	**78.2** ±2.3‡
Domain-balanced random	51.8 ±4.2†	65.4 ±2.9	70.4 ±1.8	76.2 ±1.4
EL2N	18.0 ±6.4†	22.1 ±6.9†	28.2 ±7.2†	50.0 ±14.5
features or full gradients				
k-center	43.4 ±8.0†	56.6 ±3.4	63.9 ±5.7	70.6 ±2.4
GRAD-MATCH	33.9 ±8.2†	51.0 ±7.3	64.7 ±7.4	76.4 ±3.1
influence payoffs R (shared input)				
LESS, mean	12.5 ±2.5†	14.4 ±2.1†	16.8 ±1.0†	20.0 ±2.8†
LESS, task-wise max	12.3 ±8.1†	27.5 ±7.5†	34.7 ±3.1†	39.3 ±2.7†
Round-robin	**61.2** ±3.4‡	63.1 ±1.6	64.4 ±3.4†	66.2 ±2.7†
BIDS	10.2 ±1.8†	9.9 ±1.5†	9.4 ±1.7†	20.7 ±2.6†
BIDS, instance-level	18.9 ±6.7†	30.3 ±6.1†	37.9 ±9.7†	56.3 ±10.3†
NOARB (ours)	60.1 ±3.5	63.8 ±5.2	68.0 ±3.7	72.7 ±3.5
w/o replication (static)	24.3 ±4.7†	33.5 ±4.7†	40.6 ±2.8†	53.4 ±5.1†
w/ fixed price	13.0 ±1.5†	14.7 ±1.1†	17.9 ±1.6†	23.6 ±5.1†
w/ centroid reference	34.0 ±6.1†	41.9 ±4.2†	49.6 ±2.5†	55.7 ±7.6†

Table 13: Same influence, different replicability (untargeted PACS, 5 seeds): test accuracy (%) after adding to the warm-up data the positive-gap or the replicable member of each pair. Pairs: the most influential positive-gap candidates of the more influential half of the pool, each greedily matched to the nearest replicable candidate within 5% of the influence range, 300 per seed; positive-gap members have 1.06 to 1.09 times the mean influence of their partners. Paired t-test over seeds; Holm-adjusted over the six tests, $p = 0.011$, 0.003, 0.12 (macro) and 0.052, 0.072, 0.21 (worst-domain). For reference, adding the same number of the most influential candidates of that half gives 80.13 (macro), and random candidates of that half 82.48.

Accuracy	matched on	pairs	gap > 0	gap ≤ 0	difference	p
macro	scalar influence	300	80.92	79.97	$+0.94$ ±0.37	0.002
	+ domain	300	81.17	79.94	$+1.23$ ±0.34	<0.001
	+ domain and class	300	80.76	80.01	$+0.75$ ±0.80	0.060
worst-domain	scalar influence	300	71.54	70.14	$+1.40$ ±0.91	0.013
	+ domain	300	71.78	70.09	$+1.69$ ±1.32	0.024
	+ domain and class	300	71.02	70.23	$+0.79$ ±1.45	0.206

Targeted results at every budget. Tables 15, 17 and 18 extend Table 3 to every rule and budget, and Table 16 gives the paired differences between NOARB and the main competitors. Figure 6 varies the spread t of the four target angles of targeted RotatedMNIST at a 10% budget. At $t = 0$ the four targets coincide, but the payoffs are still far from rank one: the capabilities separate the ten digits, and each target has its own 100 validation digits. In the means over two seeds (five at $t = 30°$), GRAD-MATCH is the best plotted rule that does not know the targets at every spread (at $t = 30°$ its normalized variant is slightly better, Table 3) and NOARB the best rule on the shared payoffs, ahead of round-robin and BIDS; its margin over random sampling shrinks from 13 points at $t = 0$ to about 1 point at $t = 60°$ and $90°$, as the target angles spread apart, while they remain a sixth of the pool (two-seed intervals are too wide for tests).

Sensitivity to the price cap. The only constant of NOARB is the multiplier M in $\rho_k = M/\max_z r_k(z)^+$. Table 19 varies it over a 50-fold range on untargeted PACS: the selections at

Table 14: Units and clones on PACS (5 seeds, 5% budget). Left: overlap of each selection with the one obtained before multiplying by 10 the payoffs of one domain (the one with the lowest validation accuracy at the warm-up model: sketch in three seeds), and the share of the budget spent on that domain. Random sampling uses its own draw here (74.2 without clones, 74.9 in Table 11). Right: share of the budget spent on injected near-clones of the 40 most influential examples, worst-domain accuracy with clones, and its change relative to the pool without clones.

	units of one domain $\times 10$		10 clones of 40 top examples		
Rule	overlap w/ $\times 1$	domain share	clone share	worst acc.	Δ vs. no clones
LESS, mean	0.24	0.79	0.91	70.2	-0.1
LESS, task-wise max	0.11	0.89	0.44	70.5	-0.4
Round-robin	1.00	0.25	0.09	72.8	$+0.0$
BIDS	1.00	0.21	0.01	70.5	$+0.0$
NoArb (ours)	1.00	0.37	0.02	74.7	$+0.1$
Random	1.00	0.34	0.05	74.0	-0.2

Table 15: Targeted RotatedMNIST (targets $\{0, 30°, 60°, 90°\}$), every rule: worst-target accuracy (%, mean $\pm$ 95% interval over 5 seeds). Grouping, bold, underline, † and ‡ as in Table 3.

Rule	5%	10%
no payoffs		
Random	58.1 $\pm$4.0†	61.3 $\pm$2.7
Domain-balanced random	58.7 $\pm$2.3†	62.1 $\pm$3.1
Class-stratified random	60.7 $\pm$2.9†	62.8 $\pm$3.6
Target-domain random	69.7 $\pm$2.5	73.2 $\pm$3.1
Target-domain random, class-stratified	69.9 $\pm$3.4	75.5 $\pm$1.9
EL2N	53.2 $\pm$1.3†	52.8 $\pm$1.7†
features or full gradients		
k-center	58.2 $\pm$3.4†	62.2 $\pm$1.6
Embedding similarity (RDS)	61.6 $\pm$4.5†	63.1 $\pm$3.6
KNN-uniform (gradient cosines)	60.9 $\pm$5.8†	70.0 $\pm$5.0
GRAD-MATCH	**71.1** $\pm$2.9	72.4 $\pm$3.7
GRAD-MATCH, normalized targets	69.8 $\pm$5.8	**73.5** $\pm$4.5
influence payoffs R (shared input)		
LESS, mean	49.2 $\pm$3.6†	50.2 $\pm$5.0†
LESS, task-wise max	50.0 $\pm$3.1†	52.3 $\pm$3.7†
Max-min LP	63.6 $\pm$2.5†	63.6 $\pm$6.7
k-center on payoff directions	56.0 $\pm$4.8†	58.4 $\pm$6.4†
k-center on payoffs	58.2 $\pm$3.7†	63.0 $\pm$3.1†
Round-robin	63.7 $\pm$3.8†	67.3 $\pm$4.3
BIDS	61.6 $\pm$4.8	64.7 $\pm$3.1
BIDS, instance-level	60.8 $\pm$4.3†	65.7 $\pm$4.3
NoArb (ours)	<u>69.4</u> $\pm$1.9	<u>70.0</u> $\pm$5.1
w/o replication (static)	51.1 $\pm$3.0†	50.9 $\pm$3.5†
w/ fixed price	49.7 $\pm$2.8†	50.3 $\pm$3.2†
w/ centroid reference	53.0 $\pm$3.0†	56.3 $\pm$3.4†

a 10% budget overlap with those for $M = 10$ by 94.6% or more on average, and worst-domain accuracy changes by at most 0.4 points.

Macro accuracy. Table 20 repeats Table 3 with macro (average over targets) accuracy; the ordering of the rules that do not know the targets is nearly the same as for worst-target accuracy (Spearman correlation 0.85 to 0.99 across the six columns), so gains on the worst target do not come at the expense of the average.

Table 16: Paired differences in worst-target accuracy, NoArb minus each rule (points, mean $\pm$ 95% t-interval over 5 seeds; in parentheses the unadjusted two-sided p and the Holm-adjusted p over all rules of the column of Table 3).

NoArb minus	Random	Round-robin	BIDS, inst.	GRAD-MATCH	GRAD-MATCH, norm.
RotatedMNIST 5%	$+11.3 \pm 4.8$ (0.003; 0.025)	$+5.7 \pm 3.0$ (0.006; 0.043)	$+8.6 \pm 2.6$ (0.001; 0.010)	-1.7 ± 1.9 (0.069; 0.277)	-0.4 ± 4.6 (0.831; 1.000)
RotatedMNIST 10%	$+8.7 \pm 4.4$ (0.005; 0.064)	$+2.7 \pm 3.6$ (0.105; 0.420)	$+4.3 \pm 4.2$ (0.046; 0.229)	-2.4 ± 3.6 (0.135; 0.420)	-3.5 ± 2.4 (0.015; 0.122)
PACS 5%	-0.3 ± 2.4 (0.785; 1.000)	$+1.1 \pm 1.0$ (0.030; 0.486)	$+1.9 \pm 3.5$ (0.206; 1.000)	-2.5 ± 3.2 (0.097; 0.991)	-3.2 ± 3.3 (0.055; 0.763)
PACS 10%	-2.6 ± 0.9 (0.001; 0.022)	$+1.9 \pm 1.7$ (0.034; 0.375)	$+1.4 \pm 3.1$ (0.276; 1.000)	-3.7 ± 1.7 (0.004; 0.062)	-3.4 ± 2.7 (0.027; 0.318)
OfficeHome 5%	-0.2 ± 2.0 (0.836; 1.000)	-0.4 ± 1.2 (0.420; 1.000)	-1.1 ± 1.9 (0.198; 1.000)	-4.1 ± 1.7 (0.002; 0.031)	-3.9 ± 1.6 (0.003; 0.031)
OfficeHome 10%	-2.2 ± 2.3 (0.059; 0.351)	-1.4 ± 1.1 (0.025; 0.227)	-2.8 ± 1.4 (0.006; 0.064)	-5.8 ± 1.8 (0.001; 0.015)	-5.7 ± 2.5 (0.003; 0.043)

Table 17: Targeted PACS (targets cartoon and sketch; pool: all four domains), incremental protocol: worst-target accuracy (%, mean $\pm$ 95% interval over 5 seeds). Grouping, bold, underline, † and ‡ as in Table 3.

Rule	2%	5%	10%	20%
no payoffs				
Random	72.7 $\pm$1.3	74.9 $\pm$2.4	78.0 $\pm$3.0‡	79.5 $\pm$2.7
Domain-balanced random	71.6 $\pm$3.1	74.3 $\pm$2.1	76.3 $\pm$2.6	78.3 $\pm$2.2
Class-stratified random	72.2 $\pm$2.1	73.6 $\pm$3.1	75.3 $\pm$3.0	77.0 $\pm$3.5
Target-domain random	72.2 $\pm$2.3	74.2 $\pm$2.0	76.8 $\pm$1.9	79.7 $\pm$1.7
Target-domain random, class-stratified	73.4 $\pm$3.2	76.2 $\pm$2.0	78.2 $\pm$2.4	80.3 $\pm$2.0
EL2N	71.0 $\pm$2.0	68.7 $\pm$4.4	69.1 $\pm$3.5	75.0 $\pm$1.2
features or full gradients				
k-center	71.0 $\pm$2.4	71.9 $\pm$2.8	73.5 $\pm$3.2	75.5 $\pm$3.2
Embedding similarity (RDS)	72.8 $\pm$2.9	74.4 $\pm$3.6	75.9 $\pm$3.2	78.1 $\pm$3.1
KNN-uniform (gradient cosines)	70.5 $\pm$2.8	74.5 $\pm$3.1	78.9 $\pm$2.4‡	80.8 $\pm$2.8‡
GRAD-MATCH	76.0 $\pm$2.3	77.1 $\pm$1.0	**79.1** $\pm$1.9	81.5 $\pm$2.1
GRAD-MATCH, normalized targets	**76.3** $\pm$2.1	**77.8** $\pm$1.3	78.8 $\pm$0.8	**81.6** $\pm$1.8
influence payoffs R (shared input)				
LESS, mean	70.4 $\pm$2.6	70.6 $\pm$1.9	69.8 $\pm$3.1	67.2 $\pm$2.4†
LESS, task-wise max	71.5 $\pm$2.0	70.3 $\pm$2.2	69.3 $\pm$3.1	72.1 $\pm$3.9†
Max-min LP	72.4 $\pm$1.6	72.6 $\pm$2.2	72.7 $\pm$2.1†	71.9 $\pm$3.6†
Round-robin	73.0 $\pm$3.2	73.5 $\pm$3.8	73.5 $\pm$3.2	73.9 $\pm$4.3
BIDS	70.7 $\pm$2.5	70.2 $\pm$2.9	69.3 $\pm$3.1	71.6 $\pm$3.5
BIDS, instance-level	71.6 $\pm$3.1	72.7 $\pm$3.2	74.0 $\pm$3.2	75.1 $\pm$2.6
NoArb (ours)	<u>74.2</u> $\pm$3.8	<u>74.6</u> $\pm$3.8	<u>75.4</u> $\pm$2.9	77.1 $\pm$3.8
w/o replication (static)	71.0 $\pm$3.2	72.2 $\pm$2.5	73.9 $\pm$3.4	76.7 $\pm$3.5
w/ fixed price	70.7 $\pm$2.5	71.1 $\pm$2.0	71.2 $\pm$3.5	70.6 $\pm$4.3
w/ centroid reference	72.6 $\pm$3.5	73.8 $\pm$3.5	75.1 $\pm$3.7	<u>77.6</u> $\pm$4.2

Units test. Table 21 gives the units test of Section 4.4: on targeted RotatedMNIST ($t = 30^\circ$, 10% budget) the payoff columns and the validation gradients of target 0° are multiplied by λ; overlap is the fraction of the selection shared with $\lambda = 1$. GRAD-MATCH matches the mean validation gradient; its normalized variant matches the sum of the four targets' mean gradients, each normalized to unit length. The invariance of NoArb, round-robin and BIDS holds by construction (Proposition 2(b), ranks within columns, standardized columns), so their overlap of 1.00 checks the implementation. GRAD-MATCH moves its budget toward the rescaled target as λ grows (4.6, 7.5 and 10.2% of the selection at 0° for $\lambda = 0.1$, 1 and 10; NoArb 7.7% for every λ). Its loss is significant ($p = 0.005$ at both λ), its deficit to NoArb is not ($p = 0.11$ and 0.061), and the normalized variant is above NoArb in every seed ($+3.5 \pm 2.4$ points, unadjusted $p = 0.015$; Holm-adjusted over the rules of Table 3, 0.12).

Table 18: Targeted OfficeHome (targets Clipart and Product; pool: all four domains), incremental protocol: worst-target accuracy (%, mean ± 95% interval over 5 seeds). Grouping, bold, underline, † and ‡ as in Table 3.

Rule	2%	5%	10%
no payoffs			
Random	48.3 ±2.9	52.7 ±2.9	56.2 ±3.0
Domain-balanced random	48.0 ±3.4	51.1 ±2.4	54.4 ±3.8
Class-stratified random	48.2 ±2.0	51.7 ±2.6	55.8 ±1.8
Target-domain random	48.7 ±2.6	53.6 ±2.8	58.5 ±1.7
Target-domain random, class-stratified	50.9 ±2.7	55.7 ±2.2‡	59.3 ±0.9‡
EL2N	44.8 ±4.0	48.1 ±3.5	51.0 ±2.9
features or full gradients			
k-center	47.7 ±2.4	51.8 ±2.1	55.5 ±1.6
Embedding similarity (RDS)	50.2 ±1.7	52.8 ±1.7	55.5 ±1.9
KNN-uniform (gradient cosines)	**52.2** ±2.2	56.4 ±2.7‡	58.6 ±3.1‡
GRAD-MATCH	52.1 ±1.8	**56.6** ±2.9‡	**59.8** ±2.4‡
GRAD-MATCH, normalized targets	51.7 ±2.3	56.4 ±1.9‡	59.7 ±1.9‡
influence payoffs R (shared input)			
LESS, mean	45.4 ±3.1	46.2 ±2.8†	46.2 ±2.9†
LESS, task-wise max	44.9 ±3.6	46.0 ±3.0†	46.7 ±3.2†
Max-min LP	50.6 ±2.3	51.2 ±2.9	53.2 ±2.5
Round-robin	50.8 ±1.3	52.9 ±1.8	55.4 ±1.5
BIDS	48.0 ±3.1	50.3 ±1.6†	53.0 ±1.4
BIDS, instance-level	49.3 ±3.0	<u>53.6</u> ±1.3	<u>56.8</u> ±1.6
NoArb (ours)	<u>51.2</u> ±2.0	52.5 ±1.9	54.0 ±1.9
w/o replication (static)	45.8 ±2.5	47.3 ±2.0†	49.4 ±2.0†
w/ fixed price	46.9 ±2.1†	48.2 ±2.0†	49.6 ±1.8
w/ centroid reference	46.4 ±2.1	48.0 ±2.1†	49.4 ±2.0†

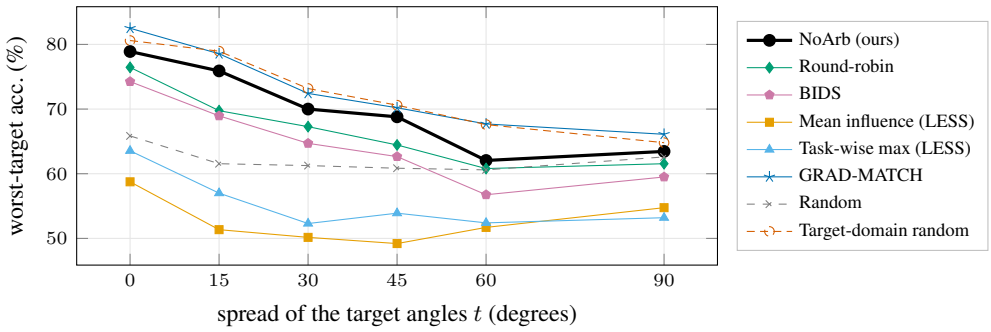

Figure 6: Targeted RotatedMNIST at a 10% budget: worst-target test accuracy as the spread t of the target angles $\{0, t, 2t, 3t\}$ grows (means over 2 seeds, and over 5 seeds at $t = 30°$, whose intervals are in Table 3).

D.1 Payoff geometry of instruction-tuning data

To check whether the regime of Theorem 1 is relevant to instruction tuning, we computed LESS-style payoffs on the processed instruction pool released with LESS (Xia et al., 2024) without fine-tuning any model. We sampled 300 examples from each of FLAN v2, CoT, Dolly and Open Assistant (1,200 in total), used Qwen2.5-0.5B (Yang et al., 2024) with rank-8 LoRA-B gradients at initialization on the query and value projections of the last four blocks (a random projection of the weight gradients), count-sketched to 8,192 dimensions (for Gaussian sketches, Proposition 4 bounds the resulting change of every gap), and took as capabilities the few-shot validation sets of MMLU (Hendrycks et al., 2021) (285 examples, 57 subjects), BBH (Suzgun et al., 2023) (81 examples, 27 tasks) and TyDiQA (Clark et al., 2020) (9 examples, 9 languages) that LESS uses; the loss is com-

Table 19: Sensitivity of NoArb to the price-cap multiplier M (untargeted PACS, 5 seeds): overlap of the 10% selection with the one for $M = 10$, and worst-domain accuracy (%).

M	2	5	10	30	100
overlap with $M = 10$	0.946	0.995	1.000	0.993	0.989
worst-domain acc., 5%	74.3 ±3.9	74.7 ±3.7	74.7 ±3.9	74.9 ±3.9	74.6 ±3.7
worst-domain acc., 10%	75.6 ±4.0	75.7 ±3.8	75.8 ±3.8	75.9 ±4.1	75.7 ±3.9

Table 20: Targeted selection, macro test accuracy over the targets (%, 5 seeds). Grouping and marks as in Table 3.

	RotatedMNIST		PACS		OfficeHome	
Rule	5%	10%	5%	10%	5%	10%
no payoffs						
Random	61.4 ±2.9†	64.4 ±3.0†	75.9 ±1.6	79.0 ±1.7	66.5 ±1.1	69.5 ±1.2
Class-stratified random	64.8 ±3.2†	66.9 ±4.3†	75.8 ±2.0	77.7 ±2.1	66.5 ±1.1	69.4 ±0.8‡
Target-domain random	73.6 ±2.5	77.5 ±2.3‡	76.8 ±2.3	79.6 ±1.4	67.5 ±2.0	71.6 ±1.3‡
Target-domain random, class-stratified	74.0 ±2.4	78.7 ±1.9‡	77.6 ±2.2	80.1 ±2.1	68.9 ±1.0‡	72.0 ±1.3‡
features or full gradients						
Embedding similarity (RDS)	65.1 ±4.4†	65.9 ±3.6†	76.4 ±2.9	78.6 ±2.4	67.0 ±1.5	69.0 ±1.0
KNN-uniform (gradient cosines)	71.1 ±1.8	**76.3** ±3.1‡	78.2 ±2.5	80.6 ±2.3‡	67.3 ±0.9	71.3 ±1.6‡
GRAD-MATCH	**74.1** ±3.2	76.1 ±3.1‡	79.0 ±1.2	**80.7** ±2.7‡	69.7 ±1.7‡	**72.2** ±1.9‡
GRAD-MATCH, normalized targets	73.9 ±4.4	76.1 ±3.7‡	**79.6** ±1.1	80.5 ±2.2	**69.9** ±1.0‡	72.2 ±1.6‡
influence payoffs R (shared input)						
LESS, mean	54.5 ±5.0†	54.3 ±5.1†	72.5 ±1.0	72.6 ±1.4	59.8 ±1.5†	60.5 ±1.4†
LESS, task-wise max	55.3 ±4.0†	57.0 ±2.5†	72.2 ±0.8	71.4 ±1.1	60.3 ±0.9†	61.4 ±1.3†
Max-min LP	67.6 ±3.5†	69.0 ±4.8†	73.6 ±2.5	73.6 ±2.3†	66.0 ±1.4	66.9 ±1.0
Round-robin	66.7 ±4.7†	69.9 ±4.1†	75.9 ±3.0	76.4 ±2.5	67.1 ±1.1	68.8 ±1.0
BIDS	65.3 ±3.4†	67.8 ±3.3†	74.2 ±1.7	75.5 ±2.2	64.6 ±1.0	66.6 ±0.9
BIDS, instance-level	65.8 ±4.8†	69.3 ±5.5†	76.6 ±1.3	77.9 ±1.6	65.9 ±0.9	69.4 ±1.3
NoArb (ours)	72.0 ±2.0	73.8 ±2.7	76.0 ±3.7	76.7 ±3.2	66.5 ±1.1	67.9 ±0.4
w/o replication (static)	54.8 ±3.4†	54.3 ±4.7†	73.8 ±1.7	75.4 ±2.1	60.5 ±1.1†	62.7 ±1.4†
w/ fixed price	53.7 ±4.9†	54.3 ±3.5†	73.1 ±1.4	73.4 ±2.4	62.2 ±1.2†	63.4 ±1.6†
w/ centroid reference	58.9 ±3.1†	60.9 ±3.1†	74.6 ±2.9	75.9 ±2.8	61.2 ±1.0†	63.1 ±1.5†

puted on answer tokens only. Unlike LESS we skip the warm-up fine-tuning, so the payoffs describe the pretrained model.

The payoff matrix is closer to rank one than the vision matrices ($\sigma_2/\sigma_1 = 0.30$ with the three benchmarks as capabilities, 0.27 with the 93 subtasks), but the benchmarks have unequal payoff scales: the mean absolute payoff is 4.5 times larger for TyDiQA than for BBH, only 70% of the candidates improve all three benchmarks to first order, and $\varepsilon/v_{\min} = 5.5$ on those candidates, so the guarantee of Theorem 1 is vacuous. Table 22 reports, for benchmark-level capabilities, the coverage of each benchmark: the mean payoff of the selected examples on it divided by the mean of its B largest payoffs. As expected for unequal scales, the scalar rules serve the benchmarks with large payoffs and neglect BBH (mean influence covers 92 to 94% of MMLU and TyDiQA but 77% of BBH at a 5% budget), while the capability-aware rules balance the three and NoArb has the best worst-benchmark coverage. With the 93 subtasks as capabilities (the granularity we use for the vision benchmarks), the capability-aware rules spread the budget over subtasks and their benchmark-level coverage is lower (0.52 to 0.60 at 5%), lowest for NoArb and below mean influence (0.77); this is not the quantity they balance, but it shows that the benchmark-level advantage depends on the granularity. These are first-order quantities computed without fine-tuning; whether they translate into benchmark accuracy after instruction tuning is not tested here.

Table 21: Units test on targeted RotatedMNIST: worst-target accuracy (%, mean ± 95% t-interval over 5 seeds) and selection overlap with $\lambda = 1$.

	worst-target acc. (%) at λ			overlap with $\lambda = 1$	
Rule	0.1	1	10	0.1	10
NoArb (ours)	70.0 ±5.1	70.0 ±5.1	70.0 ±5.1	1.00	1.00
Round-robin	67.3 ±4.3	67.3 ±4.3	67.3 ±4.3	1.00	1.00
BIDS	64.7 ±3.1	64.7 ±3.1	64.7 ±3.1	1.00	1.00
LESS, mean	50.2 ±2.0	50.2 ±5.0	46.5 ±3.6	0.88	0.70
LESS, task-wise max	52.2 ±3.0	52.3 ±3.7	48.9 ±3.6	0.80	0.47
GRAD-MATCH	65.4 ±4.4	72.4 ±3.7	63.8 ±5.5	0.61	0.50
GRAD-MATCH, normalized targets	73.5 ±4.5	73.5 ±4.5	73.5 ±4.5	1.00	1.00

Table 22: Instruction-tuning payoffs (Qwen2.5-0.5B, 1,200 candidates, no training): first-order coverage of each benchmark by the selected examples (mean payoff on the benchmark relative to its B largest payoffs); random sampling: mean over 20 draws; rows below the line use the 93 subtasks as capabilities; bold: best worst-benchmark coverage.

	5% budget				10% budget
Rule	MMLU	BBH	TyDiQA	worst	worst
Random (20 draws)	0.23	0.32	0.28	0.23	0.26
LESS, mean	0.92	0.77	0.94	0.77	0.78
LESS, task-wise max	0.87	0.72	0.93	0.72	0.72
Round-robin	0.87	0.80	0.82	0.80	0.83
BIDS	0.86	0.81	0.84	0.81	0.84
NoArb (ours)	0.89	0.84	0.85	**0.84**	**0.87**
Round-robin, 93 subtasks	0.79	0.73	0.60	0.60	0.70
BIDS, 93 subtasks	0.58	0.68	0.57	0.57	0.59
NoArb, 93 subtasks	0.52	0.81	0.59	0.52	0.58