COST-AWARE ACTIVE LOCAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Active learning (AL) reduces labeling effort by selecting informative queries, but the labeling costs usually vary substantially in real applications. Existing cost-aware active learning methods usually skip costly points, discount acquisition scores by cost, or query from a cheaper but noisy oracle. We study a complementary setting in which the label of an expensive point can be estimated from labels of cheaper points. We formalize this setting as **cost-aware active local learning** and develop two solutions. After a base learner selects a target, the Lipschitz solution compares the cost of local labels with a direct query, while the disagreement solution queries cheap inputs until the surviving hypotheses agree on its response. For disagreement search, our theory characterizes which labels suffice and bounds total expected cost under realizability and a condition relating agreement on candidates to accuracy at the selected point, including unsuccessful exploration and direct fallback. We conducted experiments on 18 molecular datasets. The results show that adding CALL increases mean area under the learning curve in 289 of 351 comparisons (82%) against the same methods without adding our approach.

1 INTRODUCTION

Active learning (AL) reduces labeling cost by selectively querying informative data points (Settles, 2009; Li et al., 2025). Most classical algorithms, however, treat each label as having the same price. This assumption is often violated in scientific experiments and human annotation. In drug discovery, structurally similar molecules can have different experimental costs because synthetic accessibility and complexity vary across candidates (Ertl & Schuffenhauer, 2009; Coley et al., 2018). Annotation cost can also depend on instance difficulty and the annotator (Herde et al., 2021). Active learning with such unequal labeling costs is commonly called cost-aware active learning.

Existing cost-aware methods usually address heterogeneous costs in one of three ways. Cost-aware selection trades informativeness against price and may avoid expensive points (Herde et al., 2021). Multi-oracle methods choose among annotators with different costs and reliability (Herde et al., 2021). Multi-fidelity methods evaluate an input at a cheaper resolution (Li et al., 2022b). These strategies decide whether to select a point or how to label that point. When a selected point is expensive, the learner can pay its full price, choose a cheaper source of its label, or select another point.

This paper studies a complementary form of cost-aware active learning. When the selected point x is expensive to label, the learner can instead query cheaper related points and use their labels to estimate its response. We formalize this setting as cost-aware active local learning (CALL). The procedure is iterative: a base acquisition rule selects a data point, and CALL either estimates its label using cheaper surrogate points or queries the selected point directly. The model is then retrained using all queried and estimated labels collected so far. This leads to two key questions: which surrogate labels suffice to estimate the label of the selected point, and when is querying them cheaper than labeling the selected point directly?

To answer these questions, we first consider a direct relation between inputs and responses. Our local Lipschitz branch (CALL-LIP) follows active local learning (Backurs et al., 2020): nearby labels constrain the prediction at the selected point. It queries the required local labels when their total remaining cost is no greater than the price of a direct query. Otherwise, it queries the selected point. Our main branch, cost-weighted disagreement CALL (CALL-DIS), is motivated by disagreement-based AL (Hanneke, 2014). It retains hypotheses compatible with queried labels and queries cheap candidates where they disagree. Each response removes incompatible hypotheses, and the search

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.

accepts an estimate once the remaining predictions at the selected point agree or lie within the required tolerance.

Our theory controls the average error of one Lipschitz predictor reused across requests and the accuracy of labels estimated by disagreement search, together with their query costs. For the Lipschitz branch, we improve the per-prediction label bound of Backurs et al. (2020, Theorem 1) from $O(\eta^{-4}\log(1/\eta))$ to $O(\eta^{-3})$ for excess average error η. The cost comparison preserves the shared predictor and limits spending to the cost of directly querying the same selected points (Theorem 1). For disagreement search, we assume that the hypothesis class contains the true response function and that candidate agreement implies accuracy at the selected point. Under these conditions, we bound total expected cost, including unsuccessful exploration and direct fallback, and identify when surrogate queries save cost (Theorem 3). For finite hypothesis classes, Section C further gives a lower bound on expected query cost. These guarantees concern prediction accuracy and query cost for a selected point, or average error for the shared Lipschitz predictor. The molecular experiments separately evaluate learning performance when purchased and estimated labels are used to train the prediction model.

We evaluate CALL-LIP on synthetic data and the practical CALL-DIS wrapper on 18 molecular datasets from BindingDB (Liu et al., 2025) and molecular property benchmarks. We compare the same learning methods with and without the practical implementation of CALL-DIS. These methods include standard active learning strategies and Unc/Cost, which already considers measurement prices. Adding CALL-DIS improves average predictive performance over the evaluated labeling budgets in 166 of 221 binding-affinity comparisons and 123 of 130 classification comparisons, or 82% overall, under the assumed costs. These results indicate that CALL can help existing methods obtain better molecular predictions from a limited labeling budget, with gains depending on the acquisition rule and settings.

2 RELATED WORK

Active learning with unequal labeling costs. Active learning reduces labeling effort by selecting examples expected to improve predictions (Settles, 2009). Informativeness-based methods query examples for which predictions are uncertain or a committee of models disagrees (Li et al., 2025). Representativeness-based methods select examples that cover different parts of the input distribution, reducing redundant queries (Sener & Savarese, 2018; Hacohen et al., 2022). When labels have unequal prices, cost-aware selection balances the expected learning benefit against annotation cost (Herde et al., 2021). Multi-oracle methods also choose among annotators with different costs and reliability (Zhang & Chaudhuri, 2015). Multi-fidelity methods instead vary the accuracy or resolution of an evaluation, using cheaper approximations to reduce the cost of collecting training data (Li et al., 2022b;a; Wu et al., 2023).

Active local learning and query complexity. Active local learning asks how to evaluate a near-optimal predictor at a specified input using fewer labels than are needed to learn the predictor in full (Backurs et al., 2020). For one-dimensional inputs and bounded responses, Backurs et al. (2020, Theorem 1) construct a Lipschitz predictor, whose change in output is bounded by a constant times the input distance. Its prediction at each input can be computed from a small local set of labels, and all such predictions belong to the same function. For a fixed accuracy, the number of labels per prediction is independent of the Lipschitz constant. The guarantee controls average prediction loss relative to the best Lipschitz function. Query-complexity theory further distinguishes labels sufficient to determine an answer from the queries needed to find those labels. A certificate is a set of observed values sufficient to fix the answer. Gavinsky et al. (2017) quantify the relationship between certificate size and the query requirements of randomized algorithms.

Disagreement and decision making. Disagreement-based active learning queries where predictors consistent with observed labels disagree (Hanneke, 2014). Each response removes inconsistent predictors and narrows the remaining disagreement region. Disagreement coefficients relate this region to prediction error (Hanneke, 2011; 2014); splitting measures quantify how queries distinguish candidate predictors (Tosh & Dasgupta, 2017). Recent work studies structured decision trees and version-space cutting for neural networks (Soltani Moakhar et al., 2026; Zhang et al., 2025). EC^2 groups hypotheses by their required answer and selects tests distinguishing groups (Golovin et al.,

2010). Given a known prior, it bounds expected cost relative to the optimal adaptive policy, including unequal test costs. Decision region determination allows several acceptable decisions per hypothesis and stops when all remaining hypotheses support a common decision (Javdani et al., 2014), while cost-sensitive decision trees also address worst-case cost (Nan et al., 2015). Our disagreement analysis instead gives a sufficient query-cost bound under realizability and a condition linking candidate agreement to accuracy at the selected point. Transductive active learning also separates the inputs available for queries from those requiring predictions (Hübotter et al., 2024). It analyzes uncertainty reduction at prediction targets, whereas CALL compares the cost of surrogate queries with directly querying a selected point.

3 METHODOLOGY

3.1 PRELIMINARIES

Let $\mathcal{X}$ be the input space and U_0 a finite pool of unlabeled data points. Each point z has a known labeling cost $c(z) > 0$. Querying z reveals its label y_z at that cost. At round t, the base active learner fits a model on the current training set $\mathcal{L}_{t-1}$ and uses its acquisition rule to select a data point $x_t \in U_{t-1}$. CALL may estimate the unknown label y_{x_t} by querying cheaper related points, which we call surrogates. The eligible surrogates form a set

$$\mathcal{C}_t(x_t) \subseteq \{z \in U_0 \setminus \{x_t\} : c(z) < c(x_t)\}.$$

The learner stores queried pairs (z, y_z) in a cache $\mathcal{A}_{t-1}$; let A_{t-1} be the set of points whose labels are cached. CALL reuses these labels and may query previously unqueried surrogates $Q_t \subseteq \mathcal{C}_t(x_t)$. If their labels allow the chosen method to estimate y_{x_t}, it returns that estimate as $\widehat{y}_t$. Otherwise, it queries x_t directly and sets $\widehat{y}_t = y_{x_t}$. The two methods below specify when an estimate is used.

Every newly queried pair (z, y_z) enters both the training set $\mathcal{L}_t$ and the cache $\mathcal{A}_t$. If the label at x_t is estimated, $(x_t, \widehat{y}_t)$ enters the training set alone. This estimate is replaced if y_{x_t} is later queried. The processed point x_t and newly queried points leave the acquisition pool.

Let $S_t = \sum_{z \in Q_t} c(z)$ be the cost of the new surrogate queries, and let $F_t \in \{0, 1\}$ indicate whether x_t is queried directly. The total cost of this round is

$$C_t = S_t + F_t c(x_t). \tag{1}$$

Cached labels are reused at zero additional cost. Before the next acquisition, the model is retrained on the updated training set, which contains both queried and estimated labels.

Figure 2 in the appendix illustrates CALL-LIP and CALL-DIS within the learning procedure. The methods differ in how they estimate the label of the selected point. CALL-LIP compares the cost of the labels needed by a local Lipschitz predictor with the price of a direct query. Given a hypothesis class $\mathcal{F}$, CALL-DIS maintains the version space, consisting of functions consistent with the queried labels:

$$\mathcal{V}(\mathcal{A}) = \{f \in \mathcal{F} : f(z) = y_z \text{ for all } (z, y_z) \in \mathcal{A}\}.$$

We call the setting realizable when the true response function $f^\star$ belongs to $\mathcal{F}$ and $y_z = f^\star(z)$. It then remains in the version space after every query. For a selected point x, agreement among all surviving functions gives the correct class label. For bounded real-valued predictions, define the smallest and largest possible values at x and half the gap between them:

$$\ell_x(\mathcal{V}) = \inf_{f \in \mathcal{V}} f(x), \qquad u_x(\mathcal{V}) = \sup_{f \in \mathcal{V}} f(x), \qquad \Gamma_x(\mathcal{V}) = \frac{u_x(\mathcal{V}) - \ell_x(\mathcal{V})}{2}. \tag{2}$$

If $f^\star \in \mathcal{V}$, the midpoint estimates $f^\star(x)$ with worst-case error $\Gamma_x(\mathcal{V})$. Thus an error tolerance $\varepsilon \geq 0$ is met whenever $\Gamma_x(\mathcal{V}) \leq \varepsilon$. Throughout, $\operatorname{diam} A = \sup A - \inf A$ for bounded nonempty $A \subset \mathbb{R}$, and $\mathbf{1}\{E\}$ is one when event E holds and zero otherwise. Logarithms are natural unless written as $\log_2$.

3.2 COST-AWARE ACTIVE LOCAL LIPSCHITZ LEARNING

CALL-LIP extends the active local learner of Backurs et al. (2020) to unequal labeling costs. Their method identifies a small set of local labels that suffice to evaluate a prediction. We compare the cost

of obtaining these labels with the price of querying the selected point directly. Thus, local prediction is used only when its required labels are eligible and cost no more than a direct query.

We use their one-dimensional regression setting, with $(X, Y) \sim D$ on $[0,1]^2$. Let $\mathcal{F}_L$ contain the functions $f : [0,1] \to [0,1]$ satisfying $|f(u) - f(v)| \le L|u - v|$, and let $\mathrm{err}_D(f) = \mathbb{E}_D|Y - f(X)|$. Thus L limits how quickly a prediction can change with its input. The labels may be noisy; the objective is to match the average error of the best function in $\mathcal{F}_L$, up to a tolerance $\eta \in (0, 1/2)$.

Before active learning, one unlabeled sample and a fixed partition determine a shared Lipschitz predictor $\widetilde{f}$, fitted locally and interpolated across gaps as in Backurs et al. (2020, Algorithms 1–2). For each x, a set $\mathsf{B}(x)$ of sample labels suffices to evaluate $\widetilde{f}(x)$; these labels are purchased only when needed. All requests evaluate the same fixed function. The sample may differ from U_0, but every uncached label must be eligible in $\mathcal{C}_t(x_t)$. Section A.1 specifies the construction and sample sizes, including the per-interval limit m.

To compute its labeling cost at round t, we reuse cached labels for free. Let D_t be the cost of the missing labels in $\mathsf{B}(x_t)$, or $+\infty$ if any is ineligible:

$$D_t = \begin{cases} \displaystyle\sum_{z \in \mathsf{B}(x_t) \setminus A_{t-1}} c(z), & \mathsf{B}(x_t) \setminus A_{t-1} \subseteq \mathcal{C}_t(x_t), \\ +\infty, & \text{otherwise.} \end{cases} \tag{3}$$

The method obtains all missing labels when $D_t \le c(x_t)$ and otherwise queries x_t directly; Algorithm 3 gives the pseudocode in the appendix. The following theorem gives the shared predictor's accuracy and the labeling cost of this rule.

Theorem 1. *Let D be any distribution on $[0,1]^2$, $L > 0$, and $\eta \in (0, 1/2)$. Use the preprocessing in Section A.1, with sample sizes of Eq.* (10)*. With probability at least $1/2$ over the partition and sample, one function $\widetilde{f} \in \mathcal{F}_L$ satisfies*

$$\mathrm{err}_D(\widetilde{f}) \le \inf_{f \in \mathcal{F}_L} \mathrm{err}_D(f) + \eta, \tag{4}$$

and every label estimated by Algorithm 3 equals $\widetilde{f}(x_t)$. Each such prediction uses at most $b_\eta = 2m = O(\eta^{-3})$ sample labels, independently of L. For every realization and every adaptive acquisition rule,

$$\sum_{t=1}^{T} C_t = \sum_{t=1}^{T} \min\{D_t, c(x_t)\} \le \sum_{t=1}^{T} c(x_t) \qquad \textit{for all } T. \tag{5}$$

Remark. Backurs et al. (2020, Theorem 1) give a per-prediction bound of $O(\eta^{-4}\log(1/\eta))$ for the same excess average error and constant success probability. We sharpen this to $O(\eta^{-3})$ by controlling the local fitting error across multiple approximation scales, using standard entropy bounds. We retain their partition and local fitting construction with smaller sample sizes, and add the cost comparison for unequal label prices. This also sharpens the sufficient price condition: if all missing labels are eligible and cost at most c_s, then $C_t \le \min\{c(x_t), b_\eta c_s\}$. The accuracy claim concerns the shared predictor's average error. The cost bound holds for every run but compares with direct purchase of the same selected sequence; a direct-query learner may select different points in later rounds. Sections A.1 and A.2 give the analysis and explain why adaptive selections require no additional failure probability.

3.3 Cost-aware Active Local Learning by Disagreement

CALL-Lip identifies the labels needed for a prediction before querying them. For a general hypothesis class, useful surrogate queries may instead depend on earlier responses. CALL-Dis chooses them through disagreement among functions compatible with the queried labels.

Disagreement-based AL maintains a version space V of functions consistent with observed labels (Hanneke, 2014). For binary labels generated by $f^\star \in \mathcal{F}$, it queries only in the disagreement region

$$\mathrm{DIS}(V) = \{z : \exists f, g \in V,\ f(z) \ne g(z)\},$$

and removes functions that contradict each queried label. These updates retain $f^\star$, so unanimous predictions are correct. Classical disagreement-based AL uses this rule to reduce error over an input

Algorithm 1 CALL-DIS

Require: Point x, version space V_0 containing $f^\star$ and consistent with cache $\mathcal{A}$, candidate distribution μ_x, costs c, cap $b \geq 0$, and draw limit n
Ensure: Label $\widehat{y}$, surrogate cost S, direct-query indicator F, updated cache $\mathcal{A}$, and newly queried surrogate pairs Q
1: Set $V = V_0$, $S = 0$, $i = 0$, and $Q = \varnothing$.
2: **while** $i < n$ and $x \in \mathrm{DIS}(V)$ **do**
3: Draw $z \sim \mu_x$ independently of previous draws; set $i = i + 1$.
4: **if** $z \notin \mathrm{DIS}(V)$, **continue**.
5: **if** z is uncached and $S + c(z) > b$, **break**.
6: **if** z is uncached, query y_z, add (z, y_z) to $\mathcal{A}$ and Q, and set $S = S + c(z)$.
7: Read y_z from $\mathcal{A}$; set $V \leftarrow \{f \in V : f(z) = y_z\}$.
8: **end while**
9: **if** $x \notin \mathrm{DIS}(V)$, return the common label at x, S, $F = 0$, $\mathcal{A}$, and Q.
10: Query x directly, add (x, y_x) to $\mathcal{A}$, and return y_x, S, $F = 1$, $\mathcal{A}$, and Q.

distribution. CALL-DIS instead draws from a fixed distribution μ_x over eligible, cheaper surrogates and stops once the surviving functions predict the same label at x.

To bound this search, agreement on candidate labels must determine the label at x. If two functions agree on all candidates but differ at x, surrogate queries cannot resolve their predictions. For $r \in [0, 1]$, let

$$B_x(f^\star, r) = \{f \in \mathcal{F} : \mu_x\{f \neq f^\star\} \leq r\}$$

contain the functions that disagree with the truth on at most an r fraction of the candidate distribution. We assume that, for some $r_* \in (0, 1/2]$,

$$f \in B_x(f^\star, r_*) \quad \Longrightarrow \quad f(x) = f^\star(x). \tag{6}$$

Thus reducing every surviving function's candidate error to r_* suffices for agreement at x.

Disagreement among the surviving functions also determines which labels may be queried. If $V \subseteq B_x(f^\star, r)$, queries occur only in $\mathrm{DIS}(B_x(f^\star, r))$. The standard disagreement coefficient bounds this region's probability mass relative to r (Hanneke, 2014). To account for unequal prices, we weight that mass by labeling cost.

Definition 2. *Fix a selected point x, a candidate distribution μ_x, a binary hypothesis class $\mathcal{F}$ containing $f^\star$, and a measurable labeling cost $0 < c(z) \leq c_{\max} < \infty$. Assume the disagreement regions are measurable. For a measurable candidate set A, define its cost-weighted mass by*

$$\mu_x^c(A) = \int_A c(z)\, d\mu_x(z).$$

For $r_0 \in (0, 1]$, the cost-aware disagreement coefficient is

$$\theta_x^c(r_0) = \sup_{r \in [r_0, 1]} \frac{\mu_x^c(\mathrm{DIS}(B_x(f^\star, r)))}{r}. \tag{7}$$

The dependence on $\mathcal{F}$, $f^\star$, and μ_x is implicit in this notation.

This coefficient links the remaining disagreement to spending: before accounting for cached labels, a draw costs at most $\theta_x^c(r_0)r$ in expectation when $V \subseteq B_x(f^\star, r)$ and $r \geq r_0$. The coefficient has units of labeling cost; unit prices recover the unweighted coefficient without its conventional lower bound of one.

To limit unsuccessful exploration, CALL-DIS caps surrogate spending at $b \geq 0$ and candidate draws at n (Algorithm 1). It queries x directly if the next new label would exceed the cap or if agreement is not reached within n draws. The choice $b = c(x)$ bounds total cost by $2c(x)$; Section B.1 compares it with the same uncapped search.

The total cost is $C_x = S + Fc(x)$, where S is surrogate spending and F indicates direct fallback. The caller adds queried pairs and $(x, \widehat{y})$ to the training set, caches only queried labels, and updates the pool as in Section 3.1.

Sufficient conditions for expected savings. The cap limits unsuccessful spending; expected savings also require agreement at x at low cost. We group candidate draws to successively halve an upper bound on the surviving hypotheses' disagreement with $f^\star$ under μ_x. Standard sample bounds under realizability determine the group sizes (Hanneke, 2014), while $\theta_x^c(r_*)$ bounds expected query costs. Sections B.2 and B.7 give the schedule and budget $b_x(\delta)$, including label reuse. Here $\widetilde{O}$ hides only logarithmic factors in $1/r_*$ and $1/\delta$.

Theorem 3. *Suppose μ_x has finite support, $f^\star \in \mathcal{F}$, the binary class $\mathcal{F}$ has Vapnik-Chervonenkis (VC) dimension $v \geq 1$, and Eq.* (6) *holds. For any $\delta \in (0,1)$, the draw schedule in Section B.7 admits a sufficient budget*

$$b_x(\delta) = \widetilde{O}(v\theta_x^c(r_*) + c_{\max})\,. \tag{8}$$

If $b_x(\delta) \leq b \leq c(x)$, Algorithm 1 returns the correct label on every path and avoids fallback with probability at least $1-\delta$. Its total cost satisfies

$$C_x \leq b + c(x), \qquad \mathbb{E}C_x \leq b + \delta c(x). \tag{9}$$

In particular, $b < (1-\delta)c(x)$ ensures lower expected cost than querying x directly.

The term $\delta c(x)$ accounts for occasional fallback, so the saving condition leaves room to pay for it. The analysis retains the standard realizable sampling rate; its cost bound incorporates unequal prices, reuse of queried labels, and unsuccessful exploration. Sections B.2 and B.7 give the sharper expectation bound and proof. The radius r_* and sufficient budget depend on the unknown $f^\star$, so the theorem characterizes sufficient conditions for savings rather than providing a calibrated budget for the empirical wrapper. The cap $b = c(x)$ remains available without this information but alone does not ensure savings.

Scalar responses. To extend this search to real-valued labels with desired error $\varepsilon > 0$, replace exact agreement at x by a prediction interval of width at most 2ε, and return its midpoint. A tolerance $\gamma \geq 0$ determines which functions survive a queried label: retain those satisfying $|f(z) - y_z| \leq \gamma$, including when initializing from cached labels. Query a candidate only if its surviving predictions span more than γ. The cost cap and fallback rule remain unchanged.

For a finite hypothesis class containing $f^\star$, the cost guarantee extends when small disagreement with $f^\star$ on candidates implies error at most ε at x. The budget depends on the cost-aware disagreement coefficient and the logarithm of the class size. Lipschitz regularity provides such a condition when the candidates are close to x. Section B.8 gives this bound and the scalar guarantee with their assumptions; Section B.9 provides the proofs.

3.4 Analyses on the Lower Bound of Query Cost

The preceding results give conditions under which our methods reduce labeling cost. We next ask how many surrogate labels any method needs, and at what cost, to estimate the label at x with specified accuracy and confidence. Section C defines these quantities for finite hypothesis classes containing $f^\star$, with exact queried labels.

Even if one cheap surrogate label determines the label at x, finding it may require several queries (Section C.2.3). Our lower bound considers which queries distinguish hypotheses whose predictions at x cannot share an accurate estimate. It applies to randomized methods that choose queries from earlier responses and meet the requirements for every hypothesis. Theorem 17 and section C.2.2 give the worst-case expected cost bound and proof, with or without direct querying of x.

4 Experiments

We next test whether surrogate querying improves learning per paid cost. Our main evaluation applies CALL-DIS to 18 molecular datasets using a fitted ExtraTrees committee (Geurts et al., 2006), following tree-committee applications in active learning and molecular affinity prediction (Borisov et al., 2011; Reker et al., 2016). Section D.1.3 explains its relation to the theoretical assumptions. Separately, synthetic checks in Section D.2 examine CALL-LIP with smaller samples than the theorem requires. After the main comparison, we examine how search cost and estimation accuracy contribute to the observed gains.

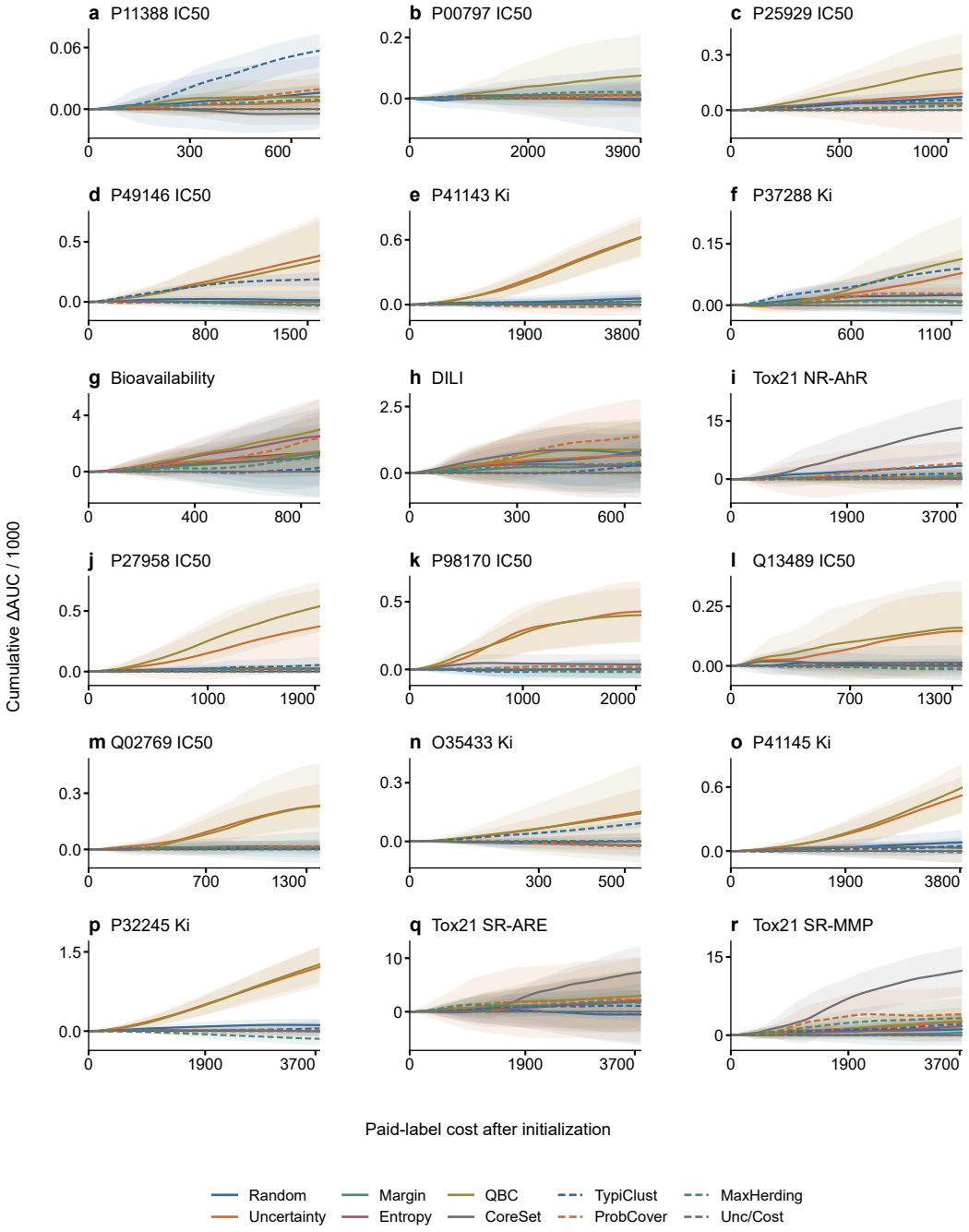


Figure 1: Cumulative improvement with XGBoost (↑) on all 18 datasets. Lines and shading show mean ± one SD over ten paired seeds.

4.1 Experimental settings

Datasets and cost settings. We report all 18 datasets: 13 BindingDB binding-affinity regression pools (Liu et al., 2025) and five molecular property classification tasks (Wu et al., 2018; Huang et al., 2021). Molecular screening is a natural application because obtaining compounds for binding and classification assays can require very different synthesis effort. We use the Synthetic-accessibility (SA) scores (Ertl & Schuffenhauer, 2009) to assign normalized benchmark costs $\{1, 4, 20, 100\}$ to

Table 1: NN3 mean ΔAULC (↑). Bold/underline mark the largest positive/most negative mean per dataset, across both classification blocks.

(a) BindingDB regression: native units

Dataset	Rnd	Unc	QBC	CS	TC	PC	MH	LL	U/C
P11388 IC50	0.008	**0.895**	0.017	−0.007	0.217	0.011	0.048	0.051	0
P00797 IC50	−0.007	**0.055**	0.007	< 0.001	0.004	0.003	−0.001	0.024	0
P25929 IC50	0.055	**0.954**	0.051	0.050	0.055	0.055	0.042	0.118	0
P49146 IC50	−0.015	**0.954**	0.064	−0.016	0.110	−0.019	−0.006	0.127	0
P41143 Ki	0.008	**0.414**	0.019	0.032	0.011	0.018	> −0.001	0.111	0
P37288 Ki	0.023	**0.983**	0.029	0.005	0.056	0.019	0.014	0.082	0
P27958 IC50	0.015	**0.606**	0.009	0.029	0.056	0.005	0.007	0.251	0
P98170 IC50	−0.001	**0.401**	0.003	0.020	0.009	0.010	0.008	0.054	< 0.001
Q13489 IC50	0.032	**0.241**	0.017	0.040	0.038	0.004	−0.008	0.056	0
Q02769 IC50	−0.003	**1.000**	0.023	0.017	0.031	0.007	< 0.001	0.166	< 0.001
O35433 Ki	0.002	**0.945**	0.030	−0.018	0.060	0.013	0.001	0.187	0
P41145 Ki	0.034	**0.548**	0.012	0.031	0.020	> −0.001	0.003	0.150	0
P32245 Ki	0.029	**0.684**	0.016	0.018	0.050	−0.012	0.002	0.110	0

(b) Molecular property classification: AUROC percentage points

Dataset	Rnd	Unc	Mar	Ent	QBC	BALD	BB	CS
Bioavailability	1.67	1.39	1.42	1.44	2.62	0.14	0.14	1.14
DILI	0.59	0.99	0.99	0.99	**1.18**	0.32	0.41	0.62
Tox21 NR-AhR	2.00	0.94	0.94	0.72	0.12	0.73	0.69	**2.71**
Tox21 SR-ARE	−0.71	3.03	3.00	2.95	0.68	0.52	0.44	2.36
Tox21 SR-MMP	0.80	1.54	1.27	1.40	1.04	0.91	0.99	**2.41**

Dataset	TC	PC	MH	BADGE	FB	AM	LL	U/C
Bioavailability	1.09	**3.02**	2.17	1.47	2.69	1.48	1.66	0.37
DILI	0.94	0.83	0.97	0.56	0.75	1.00	0.77	0.02
Tox21 NR-AhR	0.61	−0.05	0.86	1.14	1.58	1.73	2.32	0.55
Tox21 SR-ARE	1.30	1.11	0.71	2.74	2.60	**3.14**	2.40	1.77
Tox21 SR-MMP	0.62	0.73	1.18	0.32	1.27	1.41	0.66	−0.07

Rnd: Random; Unc: Uncertainty; Mar: Margin; Ent: Entropy; BB: BatchBALD; CS: CoreSet; TC: TypiClust; PC: ProbCover; MH: MaxHerding; FB: Fast-BAIT; AM: ALFA-Mix; LL: LearningLoss; U/C: Unc/Cost.

represent varying synthesis effort. The reasons for this cost assignment are given in Section D.1.1. Across these datasets, 84.6-99.0% of BindingDB records and 96.8-98.9% of classification acquisition-pool records cost at most four.

Active learning protocols. Following the studies in molecular active learning (Fralish & Reker, 2024; Graff et al., 2021), we use XGBoost and neural networks as predictive models. The neural model is denoted NN3. Input representations are summarized in Section D.1.1; model settings are in the source archive. Both tracks use fixed 80/20 training-pool/test splits; classification splits are stratified. Initial labels are sampled uniformly: 1% of the pool, with a minimum of two for regression or 20 for classification, capped at 100 for classification. The learner repeatedly selects a point from the unlabeled pool, obtains its label directly or through CALL-DIS, and updates its training data. The prediction model is refitted after each training cycle. The total paid-label budget is 4,000, including initialization, with at most 100 cost and 100 sequential actions per training cycle. Runs stop earlier if the pool is exhausted or the next action cannot be afforded. All active-learning comparisons use ten random seeds. Section D.1.2 describes pairing; the source archive gives the seeds and implementation details.

Compared acquisition rules. We compare each acquisition rule directly and with CALL-DIS. Superscripts give two applicability tags: task (C: classification, R: regression) and model (X: XGBoost, N: neural); combined letters mean both. These tags describe the implemented comparisons. For both tasks and both models, we use Random$^{\text{CR; XN}}$, Uncertainty$^{\text{CR; XN}}$, and query by committee (QBC)$^{\text{CR; XN}}$ (Settles, 2009); CoreSet$^{\text{CR; XN}}$ (Sener & Savarese, 2018), TypiClust$^{\text{CR; XN}}$ (Ha-

cohen et al., 2022), ProbCover$^{CR;\,XN}$ (Yehuda et al., 2022), and MaxHerding$^{CR;\,XN}$ (Bae et al., 2024); and Unc/Cost$^{CR;\,XN}$, which divides uncertainty by price (Herde et al., 2021). Classification additionally uses Margin$^{C;\,XN}$ and Entropy$^{C;\,XN}$ (Settles, 2009). Neural models additionally use LearningLoss$^{CR;\,N}$ (Yoo & Kweon, 2019); for classification, they also use BALD$^{C;\,N}$ (Houlsby et al., 2011), BatchBALD$^{C;\,N}$ (Kirsch et al., 2019), BADGE$^{C;\,N}$ (Ash et al., 2020), Fast-BAIT$^{C;\,N}$ (Ash et al., 2021; Huseljic et al., 2024), and ALFA-Mix$^{C;\,N}$ (Parvaneh et al., 2022). The batch selectors use the sequential adaptations described in Section D.1.2.

We use AUROC for molecular property classification and negative RMSE for BindingDB regression; higher is better. We report the paired change in area under the learning curve (ΔAULC), where a positive value favors CALL-DIS. For accepted estimates, we report classification accuracy and, for regression, mean absolute error and the fraction within 0.5 pAffinity units of the recorded response. Metric definitions and uncertainty calculations are in Section D.1.4; paired significance tests are in Section D.4.

4.2 Results of the Main Performance Comparison

Figure 1 and table 1 compare each acquisition rule with and without CALL-DIS through XGBoost learning-curve gains and NN3 ΔAULC, respectively. Lines average ten seeds; shading shows one standard deviation.

Across 18 datasets, mean ΔAULC is positive in 289 of 351 dataset-model-selector comparisons, indicating that the CALL-DIS wrapper usually improves learning per paid cost. Gains averaged over a learning curve need not hold at every budget. We also compare each method pair at the same labeling cost using a paired t-test; Section D.4 gives the settings and results at costs of 500 and 1,000 beyond initialization. For NN3, Uncertainty gains most on each BindingDB pool. Its direct-query baseline ignores price and often acquires fewer labels at a given cost (Section D.3), which helps explain the improvement. By contrast, classification gains are less concentrated, so the benefit depends on both the acquisition rule and dataset.

4.3 Label Estimation and Query Cost

To examine query cost, we use finite hypothesis classes containing the truth. Surrogate queries save cost when they find informative cheap labels; otherwise, unsuccessful queries add to direct fallback (Section D.6). These comparisons vary query choice and prices without fitting a predictor.

In the molecular experiments, most accepted estimates have only one surviving tree, so agreement provides no independent check of its prediction. Requiring multiple surviving trees reduces regression error retrospectively, but retains far fewer estimates and does not consistently improve classification tasks (Sections D.5.1 and D.5.2). Stricter acceptance therefore needs validation for each task. The two-task ablation shows that purchased labels alone retain gains over direct querying. Adding estimated labels to the same purchases lowers mean AULC on both tasks (Section D.5.3). Thus, these results support a benefit from the labels acquired by the procedure, but do not establish that adding estimated labels improves prediction (Section D.6.1).

5 Conclusion

We introduce cost-aware active local learning (CALL) to study how an active learner can obtain the label of an expensive selected point by querying cheaper related points. We develop two methods. CALL-LIP uses Lipschitz regularity to identify the labels needed for a prediction and compares their total cost with querying the selected point directly. CALL-DIS queries cheaper candidates until the surviving hypotheses agree at the selected point, with direct fallback when needed. Our analysis reduces the number of labels sufficient for local prediction, gives conditions under which surrogate queries lower expected labeling cost, and establishes a lower bound on query cost. Experiments on 18 molecular datasets show improved learning per paid cost in most comparisons under the assumed costs, although the estimated labels can have errors. Future work will investigate better surrogate models and acceptance rules, as well as query strategies that reuse purchased labels across multiple predictions.

Reproducibility Statement

Sections 3 and C define the algorithms, benchmarks, and assumptions; Sections A, B and C.2 give their proofs. Sections 4.1 and D.1 describe the reported studies. Sections D.4, D.5.1, D.5.3 and D.6 report paired tests, accepted-label errors, replays, and search controls. The accompanying anonymous source archive contains the available scripts and implementation settings for the reported studies and identifies the external data inputs they require.

Ethics Statement

This work uses existing benchmark responses and cost proxies. Molecular cost proxies need not reflect every laboratory or procurement setting. Scientific applications require validation of estimated labels against measured responses.

Disclosure of Artificial Intelligence Assistance

In this work, we used generative AI tools for literature search, manuscript editing, code and mathematical review. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *Proceedings of the 8th International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1906.03671.

Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham M. Kakade. Gone fishing: Neural active learning with Fisher embeddings. In *Advances in Neural Information Processing Systems*, volume 34, pp. 8927–8939, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/4afe044911ed2c247005912512ace23b-Abstract.html.

Arturs Backurs, Avrim Blum, and Neha Gupta. Active local learning. In *Proceedings of the 33rd Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 363–390. PMLR, 2020. URL https://proceedings.mlr.press/v125/backurs20a.html.

Wonho Bae, Junhyug Noh, and Danica J. Sutherland. Generalized coverage for more robust low-budget active learning. In *Proceedings of the 18th European Conference on Computer Vision*, volume 15141 of *Lecture Notes in Computer Science*, pp. 318–334. Springer, 2024. doi: 10.1007/978-3-031-73010-8_19.

Alexander Borisov, Eugene Tuv, and George Runger. Active batch learning with stochastic query-by-forest (SQBF). In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, volume 16 of *Proceedings of Machine Learning Research*, pp. 59–69. PMLR, 2011. URL https://proceedings.mlr.press/v16/borisov11a.html.

Connor W. Coley, Luke Rogers, William H. Green, and Klavs F. Jensen. SCScore: Synthetic complexity learned from a reaction corpus. *Journal of Chemical Information and Modeling*, 58(2):252–261, 2018. doi: 10.1021/acs.jcim.7b00622.

Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1):8, 2009. doi: 10.1186/1758-2946-1-8.

Zachary Fralish and Daniel Reker. Finding the most potent compounds using active learning on molecular pairs. *Beilstein Journal of Organic Chemistry*, 20:2152–2162, 2024. doi: 10.3762/bjoc.20.185. URL https://beilstein-journals.org/bjoc/articles/20/185.

Dmitry Gavinsky, Rahul Jain, Hartmut Klauck, Srijita Kundu, Troy Lee, Miklos Santha, Swagato Sanyal, and Jevgenijs Vihrovs. Quadratically tight relations for randomized query complexity. *arXiv preprint arXiv:1708.00822*, 2017. URL https://arxiv.org/abs/1708.00822v1.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006. doi: 10.1007/s10994-006-6226-1.

Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal Bayesian active learning with noisy observations. In *Advances in Neural Information Processing Systems*, volume 23, 2010. URL https://papers.nips.cc/paper_files/paper/2010/hash/1e6e0a04d20f50967c64dac2d639a577-Abstract.html.

David E. Graff, Eugene I. Shakhnovich, and Connor W. Coley. Accelerating high-throughput virtual screening through molecular pool-based active learning. *Chemical Science*, 12:7866–7881, 2021. doi: 10.1039/D0SC06805E. URL https://pubs.rsc.org/en/content/articlelanding/2021/sc/d0sc06805e.

Guy Hacohen, Avihu Dekel, and Daphna Weinshall. Active learning on a budget: Opposite strategies suit high and low budgets. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8175–8195. PMLR, 2022. URL https://proceedings.mlr.press/v162/hacohen22a.html.

Steve Hanneke. Rates of convergence in active learning. *The Annals of Statistics*, 39(1):333–361, 2011. doi: 10.1214/10-AOS843.

Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends in Machine Learning*, 7 (2–3):131–309, 2014. doi: 10.1561/2200000037. URL https://web.ics.purdue.edu/~hanneke/docs/active-survey/active-survey.pdf. Extended version: Theory of Active Learning, version 1.1, September 22, 2014.

Friedrich Hastedt, Klaus Hellgardt, Sophia Yaliraki, Dongda Zhang, and Antonio del Rio Chanona. MolPrice: assessing synthetic accessibility of molecules based on market value. *Journal of Cheminformatics*, 17: 150, 2025. doi: 10.1186/s13321-025-01076-3. URL https://link.springer.com/article/10.1186/s13321-025-01076-3.

Marek Herde, Denis Huseljic, Bernhard Sick, and Adrian Calma. A survey on cost types, interaction schemes, and annotator performance models in selection algorithms for active learning in classification. *IEEE Access*, 9:166970–166989, 2021. doi: 10.1109/access.2021.3135514. URL https://doi.org/10.1109/access.2021.3135514.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011. URL https://arxiv.org/abs/1112.5745.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W. Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics Data Commons: Machine learning datasets and tasks for drug discovery and development. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/4c56ff4ce4aaf9573aa5dff913df997a-Abstract-round1.html.

Jonas Hübotter, Bhavya Sukhija, Lenart Treven, Yarden As, and Andreas Krause. Transductive active learning: Theory and applications. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/e17fe6fe9990fffb637b42c98c005515-Abstract-Conference.html.

Denis Huseljic, Paul Hahn, Marek Herde, Lukas Rauch, and Bernhard Sick. Fast fishing: Approximating BAIT for efficient and scalable deep active image classification. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Research Track*, volume 14947 of *Lecture Notes in Computer Science*, pp. 280–296. Springer, 2024. doi: 10.1007/978-3-031-70368-3_17.

Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, Drew Bagnell, and Siddhartha Srinivasa. Near optimal Bayesian active learning for decision making. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 430–438, 2014. URL https://proceedings.mlr.press/v33/javdani14.html.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and diverse batch acquisition for deep Bayesian active learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.nips.cc/paper_files/paper/2019/hash/95323660ed2124450caaac2c46b5ed90-Abstract.html.

Malte Korn, Philip Judson, Raphael Klein, Christian Lemmen, Marc C. Nicklaus, and Matthias Rarey. SAVI Space—combinatorial encoding of the billion-size synthetically accessible virtual inventory. *Scientific Data*, 12:1064, 2025. doi: 10.1038/s41597-025-05384-z. URL https://www.nature.com/articles/s41597-025-05384-z.

Dongyuan Li, Zhen Wang, Yankai Chen, Renhe Jiang, Weiping Ding, and Manabu Okumura. A survey on deep active learning: Recent advances and new frontiers. *IEEE Transactions on Neural Networks and Learning Systems*, 36(4):5879–5899, 2025. doi: 10.1109/tnnls.2024.3396463. URL https://doi.org/10.1109/tnnls.2024.3396463.

Shibo Li, Jeff M. Phillips, Xin Yu, Robert Kirby, and Shandian Zhe. Batch multi-fidelity active learning with budget constraints. In *Advances in Neural Information Processing Systems*, volume 35, pp. 995–1007, 2022a. doi: 10.52202/068431-0073. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/06ea400b9b7cfce6428ec27a371632eb-Abstract-Conference.html.

Shibo Li, Zheng Wang, Robert Kirby, and Shandian Zhe. Deep multi-fidelity active learning of high-dimensional outputs. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 1694–1711. PMLR, 2022b. URL https://proceedings.mlr.press/v151/li22b.html.

Tiqing Liu, Linda Hwang, Stephen K. Burley, Carmen I. Nitsche, Christopher Southan, W. Patrick Walters, and Michael K. Gilson. BindingDB in 2024: a FAIR knowledgebase of protein-small molecule binding data. *Nucleic Acids Research*, 53(D1):D1633–D1644, 2025. doi: 10.1093/nar/gkae1075. URL https://doi.org/10.1093/nar/gkae1075.

Feng Nan, Joseph Wang, and Venkatesh Saligrama. Max-cost discrete function evaluation problem under a budget. *arXiv preprint arXiv:1501.02702*, 2015. URL https://arxiv.org/abs/1501.02702v1.

Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Gholamreza (Reza) Haffari, Anton van den Hengel, and Javen Qinfeng Shi. Active learning by feature mixing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12237–12246, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Parvaneh_Active_Learning_by_Feature_Mixing_CVPR_2022_paper.html.

D. Reker, P. Schneider, and G. Schneider. Multi-objective active machine learning rapidly improves structure–activity models and reveals new protein–protein interaction inhibitors. *Chemical Science*, 7(6):3919–3927, 2016. doi: 10.1039/C5SC04272K.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t.

Ruben Sanchez-Garcia, Dávid Havasi, Gergely Takács, Matthew C. Robinson, Alpha Lee, Frank von Delft, and Charlotte M. Deane. CoPriNet: graph neural networks provide accurate and rapid compound price prediction for molecule prioritisation. *Digital Discovery*, 2:103–111, 2023. doi: 10.1039/D2DD00071G. URL https://pubs.rsc.org/en/content/articlelanding/2023/dd/d2dd00071g.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *Proceedings of the 6th International Conference on Learning Representations*, 2018. URL https://arxiv.org/abs/1708.00489.

Burr Settles. Active learning literature survey. Technical Report 1648, University of Wisconsin–Madison, 2009. URL https://research.cs.wisc.edu/techreports/2009/TR1648.pdf.

Arshia Soltani Moakhar, Tanapoom Laoaron, Faraz Ghahremani, Kiarash Banihashem, and MohammadTaghi Hajiaghayi. Active learning for decision trees with provable guarantees. *arXiv preprint arXiv:2601.20775*, 2026. URL https://arxiv.org/abs/2601.20775v2.

TargetMol. XCT790, Catalog No. T0593: Product and Pack-Size Prices. Online product catalog, 2026. URL https://www.targetmol.com/compound/XCT790. Accessed September 21, 2026. Listed US-dollar pack prices are illustrative and may vary over time and by market.

Christopher Tosh and Sanjoy Dasgupta. Diameter-based active learning. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3444–3452. PMLR, 2017. URL https://proceedings.mlr.press/v70/tosh17a.html.

University of Tennessee Health Science Center. FY26 Medicinal Chemistry Core Services and Fees: Price List. Institutional service price list, 2026. URL https://www.uthsc.edu/research/institutional-cores/medicinal-chemistry/price-list.php. Accessed September 21, 2026. The page labels its schedule FY26 and states pricing effective July 1, 2024; figures are illustrative listed prices, not project quotations.

Dongxia Wu, Ruijia Niu, Matteo Chinazzi, Yian Ma, and Rose Yu. Disentangled multi-fidelity deep Bayesian active learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 37624–37634. PMLR, 2023. URL https://proceedings.mlr.press/v202/wu23p.html.

Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: A benchmark for molecular machine learning. *Chemical Science*, 9 (2):513–530, 2018. doi: 10.1039/C7SC02664A.

Ofer Yehuda, Avihu Dekel, Guy Hacohen, and Daphna Weinshall. Active learning through a covering lens. In *Advances in Neural Information Processing Systems*, volume 35, pp. 22354–22367, 2022. doi: 10.52202/068431-1624.

Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 93–102, 2019. doi: 10.1109/CVPR.2019.00018.

Chicheng Zhang and Kamalika Chaudhuri. Active learning from weak and strong labelers. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://papers.nips.cc/paper_files/paper/2015/hash/eba0dc302bcd9a273f8bbb72be3a687b-Abstract.html.

Erica Zhang, Fangzhao Zhang, and Mert Pilanci. Active learning of deep neural networks via gradient-free cutting planes. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 76831–76879, 2025. URL https://proceedings.mlr.press/v267/zhang25db.html.

Xiang Zhang, Jia Liu, Bufan Xu, Zihan Zhang, Zifu Huang, Kaixian Chen, Dingyan Wang, and Xutong Li. SynFrag: Synthetic accessibility predictor based on fragment assembly generation in drug discovery. ChemRxiv preprint, 2026. URL https://simmzx.github.io/SynFrag_ChemRxiv.pdf. Preprint; not peer reviewed. The finetuning dataset description uses SA scores in [3.5,6] as an ambiguous decision-boundary subset.

A Proofs for Prediction Accuracy and Local Learning

We prove the accuracy, query, and cost bounds for CALL-Lip. Section B proves the disagreement-search guarantees. Section C develops the lower bounds on query cost summarized in Section 3.4; Section C.2 supplies their proofs. Experimental details appear in Section D.

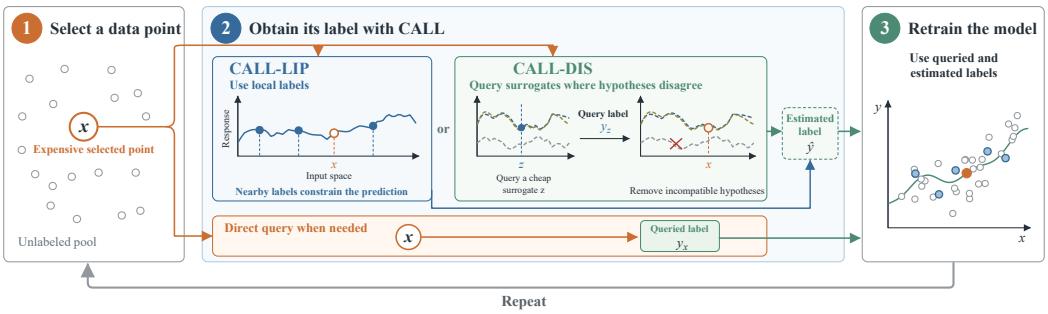


Figure 2: Overview of CALL. A base learner selects a point; CALL-Lip or CALL-Dis obtains its label through surrogate queries or a direct query. Queried and estimated labels are then used to retrain the model.

We use natural logarithms except in $\log_2$, and write $\mathbf{1}\{E\}$ for the indicator of an event E. A transcript lists the queried inputs and the labels obtained. Unless stated otherwise, responses in the realizable results are exact values of a fixed function $f^\star$. The local Lipschitz analysis instead allows noisy labels and bounds average prediction error under a distribution D.

A.1 The shared active-local predictor

We first define the predictor completely, then prove its statistical guarantee and the effect of the cost comparison. The local fitting and interpolation construction follows Backurs et al. (2020, Algorithms 1–2 and Theorem 1). We retain that construction and sharpen its label bound using a standard entropy-integral argument. The explicit parameters below also cover small L, boundary intervals, and repeated input coordinates. This part uses the agnostic model: $(X, Y) \sim D$ on $[0,1]^2$, with input marginal D_X, with no assumption that one Lipschitz function generates all responses. In particular, two observations at the same input can have different labels.

Sample sizes. For $L > 0$ and $\eta \in (0, 1/2)$, define

$$q = \lceil 32/\eta \rceil, \qquad \tau = \eta/24, \qquad \beta = \eta/32, \qquad J_{\max} = \lceil L/(q+1) \rceil + 2.$$

Here q/L bounds a fitting interval's length, τ is the allowed difference between empirical and expected loss within an interval, β bounds a local failure probability, and $J_{\max}$ bounds the number of fitting intervals. Let $A \geq 1$ be the universal constant in Lemma 4, and set $C_0 = 600(A+1)^2$. Choose

$$m = \left\lceil \max\left\{ \frac{C_0(q + \log(2/\beta))}{\tau^2},\, 4\log(1/\beta) \right\} \right\rceil, \qquad M = \left\lceil \frac{64 J_{\max} m}{\eta} \right\rceil. \tag{10}$$

Thus $m = O(\eta^{-3})$ and $b_\eta := 2m$ is independent of L. The unlabeled sample size is $M = O((L + \eta^{-1})\eta^{-3})$. The additional boundary term covers every $L > 0$; when $L \geq \eta^{-1}$, this reduces to $O(L\eta^{-3})$.

The partition determines which labels are used. Independently of the sample, draw J uniformly from $\{1, \ldots, q\}$. On the real line, put a bridge at

$$B_k = [(J + k(q+1))/L,\, (J + k(q+1) + 1)/L), \qquad k \in \mathbb{Z}.$$

Intersect this partition with $[0,1]$. The complementary intervals are the fitting intervals; call their collection $\mathcal{I}_J$. They have length at most q/L, and $|\mathcal{I}_J| \leq J_{\max}$. Use left-closed, right-open intervals

Algorithm 2 Shared preprocessing for CALL-LIP

Require: $L > 0$, accuracy $\eta \in (0, 1/2)$, sampling access to D_X, and an oracle that reveals Y_j when index j is queried
Ensure: A fixed partition, retained indices, and deterministic local fitting rules
1: Set $q = \lceil 32/\eta \rceil$ and choose m, M by Eq. (10).
2: Draw J uniformly from $\{1, \dots, q\}$.
3: Form bridges $[(J+k(q+1))/L, (J+k(q+1)+1)/L) \cap [0,1]$, $k \in \mathbb{Z}$; use the intervals between them as fitting intervals.
4: Draw M independent inputs from D_X, keeping their labels hidden.
5: For each fitting interval I, retain its first m sample indices, or all if fewer occur; denote them by J_I.
6: Fix the empirical-loss minimization and tie-breaking rules in Eq. (11).
7: Store the partition, retained indices, and fitting rules; reuse them for every selected point.

to assign sample points, with 1 assigned by the same real-line partition. Fits are defined on the closures of their assigned intervals. This convention also permits a singleton fitting interval at 1.

Draw (X_j, Y_j), $j = 1, \dots, M$, independently from D and initially hide every Y_j. For each $I \in \mathcal{I}_J$, retain the first m indices with $X_j \in I$, or all such indices if fewer occur, and denote this set by J_I. The retained indices depend only on the X_j and their order, before any Y_j is observed. If $X_i = X_j$ for $i \neq j$, the two labels remain separate observations in the loss. Points in bridges are not used for local fitting. To make sample labels eligible, include their indexed records in U_0 with known prices before learning. The cache identifies observations by record, not input coordinate; distinct noisy observations at the same coordinate are charged separately. Ineligible uncached records trigger direct querying through $D_t = +\infty$.

Fits and interpolation. For a fitting interval I, list its endpoints and distinct retained input coordinates as $u_1 < \cdots < u_s$. Choose a vector $v \in [0,1]^s$ minimizing

$$\sum_{j \in J_I} |Y_j - v_{i(j)}| \quad \text{subject to} \quad |v_i - v_l| \le L|u_i - u_l| \quad \text{for all } i, l, \tag{11}$$

where $i(j)$ is the index with $u_{i(j)} = X_j$. The feasible set is compact and nonempty, and the objective is continuous. Among its minimizers, successively minimize $v_1, v_2, \dots, v_s$; each step preserves a nonempty compact set and the final vector is unique. Linear interpolation between knots defines g_I. A singleton interval uses its single fitted value. With no retained observations, the same rule chooses the constant zero function.

Every bounded L-Lipschitz function restricts to a feasible vector in Eq. (11); conversely, the interpolant of a feasible vector is bounded and L-Lipschitz. Thus g_I is an empirical risk minimizer on the whole interval, not just on the knots. This is the local empirical risk minimization (ERM) step.

Define $\widetilde{f} = g_I$ on fitting intervals. Across an interior bridge $[a, b]$, use

$$\widetilde{f}(u) = g_{I_-}(a) + \frac{u-a}{b-a}\big(g_{I_+}(b) - g_{I_-}(a)\big), \tag{12}$$

where I_-, I_+ are the adjacent fitting intervals. Across a clipped boundary bridge, use the constant endpoint value of its existing neighbor. Such a neighbor exists because the first fitting interval starts at zero and has positive length. Interpolated values agree with adjacent fits at shared endpoints, so assigning a shared endpoint to either interval does not change its prediction.

For a selected point in a fitting interval, $\mathsf{B}(x)$ contains that interval's retained indices. For a selected point in an interior bridge, it contains the two adjacent blocks; for a boundary bridge, the one existing block. Therefore $|\mathsf{B}(x)| \le 2m$. For this sampled construction, costs and cached labels are indexed by j: querying Y_i does not reveal Y_j when $i \neq j$, even if $X_i = X_j$. Set differences involving $\mathsf{B}(x)$ use these indices, and predictions use the corresponding X_j.

Algorithm 2 collects the inherited preprocessing steps. The local fits and interpolation used to evaluate $\widetilde{f}(x)$ are defined above.

We use the following standard uniform-convergence bound to analyze each local fit. For a class $\mathcal{G}$ of bounded functions, $N_\infty(u, \mathcal{G})$ is the smallest number of functions in the class needed so that every member is within u of one of them at every input. The logarithm of this covering number measures how many distinguishable functions remain at accuracy u.

Lemma 4. *Let $\mathcal{G}$ be a nonempty class of measurable functions taking values in $[0, 1]$, separable in the uniform norm, with finite uniform covering numbers and finite integral below. For m independent observations $Z_1, \dots, Z_m$ with common law P, write $Pg = \mathbb{E}g(Z)$ and $P_m g = m^{-1} \sum_{i=1}^m g(Z_i)$. There is a universal constant $A \geq 1$ such that, for $\beta \in (0, 1)$, with probability at least $1 - \beta$,*

$$\sup_{g \in \mathcal{G}} |P_m g - Pg| \leq \frac{A}{\sqrt{m}} \left(1 + \int_0^1 \sqrt{\log N_\infty(u, \mathcal{G})}\, du \right) + \sqrt{\frac{\log(2/\beta)}{2m}}. \tag{13}$$

This is the usual consequence of symmetrization, Dudley's entropy integral, and bounded-difference concentration; see Telgarsky (2021, Sections 13.1–13.2 and Theorem 15.2). Uniform covers also cover the empirical root-mean-square distance used in Dudley's bound. Centering the class at one fixed function and applying the bound to both signs gives the absolute deviation above; the additive constant covers the fixed function. We use this standard result without reproving it.

Proof of the statistical and query bounds in Theorem 1. We follow the partition and local-risk decomposition of Backurs et al. (2020, Theorem 1). The change is the bound on the estimation error within a fitting interval. We first verify that the constructed pieces define one Lipschitz function, then bound their estimation errors and combine them over the partition.

Each g_I is bounded and L-Lipschitz by construction. Every interior bridge has length $1/L$, and its endpoint values differ by at most one. Its interpolant therefore has slope of magnitude at most L. A boundary extension is constant. The pieces meet continuously; summing their Lipschitz inequalities along the segment between any two inputs proves $|\widetilde{f}(x) - \widetilde{f}(x')| \leq L|x - x'|$. The definition also shows that each prediction uses only the retained labels in the adjacent fitting intervals.

Fix a partition and a fitting interval I with mass $p_I = \Pr_D(X \in I) > 0$. Let D_I be the conditional law given $X \in I$, and let $\mathcal{F}_I$ contain all $[0, 1]$-valued L-Lipschitz functions on the closure of I. For $g \in \mathcal{F}_I$, write

$$R_I(g) = \mathbb{E}_{D_I} |Y - g(X)|, \qquad \widehat{R}_I(g) = \frac{1}{m} \sum_{j=1}^m |Y_j - g(X_j)|,$$

where the second expression first refers to m independent observations from D_I. We will verify below that retained observations have this law whenever the interval receives at least m samples.

To apply Lemma 4, we bound the covering numbers of $\mathcal{F}_I$. Fix $u \in (0, 1]$. Divide the interval into $k = \lceil 4L|I|/u \rceil$ segments of length at most $u/(4L)$; a singleton requires only one knot. Round each knot value down to a multiple of $u/4$. There are at most $1 + 4/u$ possibilities for the first value. Since the unrounded values at consecutive knots differ by at most $u/4$, there are at most five possibilities for each subsequent rounded value. Consequently there are at most

$$(1 + 4/u) 5^{\lceil 4q/u \rceil}$$

possible sequences. A function differs from its rounded left-knot value by at most $u/2$ throughout each segment; include the final endpoint in the same bound. Functions with the same rounded sequence therefore differ by at most u everywhere. Choosing one member of each nonempty group gives a cover whose centers lie in $\mathcal{F}_I$. Since $|I| \leq q/L$ and $q \geq 1$,

$$\log N_\infty(u, \mathcal{F}_I) \leq \log(1 + 4/u) + (4q/u + 1) \log 5 \leq \frac{32(q + 1)}{u}. \tag{14}$$

The same bound holds for the loss class $\mathcal{G}_I = \{(x, y) \mapsto |y - g(x)| : g \in \mathcal{F}_I\}$, because $\big||y - g(x)| - |y - h(x)|\big| \leq |g(x) - h(x)|$. These classes are separable: bounded Lipschitz functions on a compact interval form a compact class in the uniform norm, and the absolute-loss map is continuous in that norm. Thus the supremum in Lemma 4 is measurable.

Integrating Eq. (14) gives

$$\int_0^1 \sqrt{\log N_\infty(u, \mathcal{G}_I)}\, du \leq 2\sqrt{32(q + 1)}.$$

This integral gives the improved sample-size bound. Integrating over all approximation scales gives a bound proportional to $\sqrt{q}$, whereas a single cover at resolution proportional to η introduces an additional power of $1/\eta$ in the required sample size. By Lemma 4, with probability at least $1-\beta$,

$$\sup_{g\in\mathcal{F}_I} |\widehat{R}_I(g) - R_I(g)| \le \frac{A(1+2\sqrt{32(q+1)})}{\sqrt{m}} + \sqrt{\frac{\log(2/\beta)}{2m}}$$
$$\le \sqrt{\frac{C_0(q+\log(2/\beta))}{m}} \le \tau. \tag{15}$$

For the second inequality, $q \ge 1$ gives $1+2\sqrt{32(q+1)} \le 17\sqrt{q}$; squaring the sum and using $(a+b)^2 \le 2a^2+2b^2$ is bounded by the chosen $C_0 = 600(A+1)^2$. The final inequality follows from Eq. (10).

On this event, the empirical minimizer satisfies, simultaneously for every $f \in \mathcal{F}_L$,

$$R_I(g_I) \le \widehat{R}_I(g_I) + \tau \le \widehat{R}_I(f) + \tau \le R_I(f) + 2\tau \le R_I(f) + \eta/4.$$

Here f is restricted to I, so it is feasible in the local minimization. This establishes the needed conditional risk bound without changing the fitting rule.

Let N_I count all sample points assigned to I. Conditional on the partition, N_I has the binomial law with parameters M, p_I. To determine the law of the retained observations, first condition on which sample indices fall in I. At those indices, the pairs (X_j, Y_j) are independent with law D_I. Whenever at least m indices fall in I, selecting the first m therefore gives the law D_I^m. This law is the same for every such membership pattern, so it also holds when conditioning only on $N_I \ge m$. The conditional failure probability of the uniform deviation bound is consequently at most β whenever $N_I \ge m$.

Call an interval bad if $N_I < m$ or its uniform deviation bound fails. Intervals with $p_I < 2m/M$ have total mass at most

$$2|\mathcal{I}_J|m/M \le 2J_{\max}m/M \le \eta/32.$$

For the remaining intervals, $Mp_I \ge 2m$. A binomial lower-tail bound gives

$$\Pr(N_I < m \mid J) \le e^{-Mp_I/8} \le e^{-m/4} \le \beta.$$

This is the standard multiplicative binomial lower-tail bound with deviation $1/2$ (Boucheron et al., 2013, Section 2.7), applied using $m \le Mp_I/2$. Summing by interval mass, and then including fitting failures, proves

$$\mathbb{E}\left[\sum_{I\in\mathcal{I}_J} p_I \mathbf{1}\{I \text{ is bad}\} \,\middle|\, J\right] \le \eta/32 + \beta + \beta = 3\eta/32. \tag{16}$$

Intervals of zero mass contribute nothing.

For a fixed $x \in [0,1]$, membership in a bridge requires $Lx \in [J+k(q+1), J+k(q+1)+1)$ for some integer k. Among $J = 1, \dots, q$, at most one choice has this property. The half-open convention makes this true also at atoms on partition boundaries. Thus $\Pr_J(x \text{ is in a bridge}) \le 1/q$. Integrating over X gives

$$\mathbb{E}_J \Pr_D(X \text{ is in a bridge} \mid J) \le 1/q \le \eta/32. \tag{17}$$

On good fitting intervals, the ERM inequality bounds excess conditional risk relative to every $f \in \mathcal{F}_L$ by $\eta/4$. On bad intervals and bridges, bounded loss gives the upper bound one. For every realization, therefore,

$$\mathrm{err}_D(\widetilde{f}) - \inf_{f\in\mathcal{F}_L} \mathrm{err}_D(f) \le R := \eta/4 + \sum_{I\in\mathcal{I}_J} p_I \mathbf{1}\{I \text{ is bad}\} + \Pr_D(X \text{ is in a bridge} \mid J).$$

The local loss bound holds for every comparator f on the same event. We can therefore take the infimum over f without changing that event. Eqs. (16) and (17) give $\mathbb{E}R \le 3\eta/8$. Markov's inequality then gives $\Pr(R > \eta) \le 3/8$. On the complementary event the desired risk inequality holds. Its probability is at least $5/8$, and hence at least $1/2$ as stated. Finally, each prediction uses at most two retained blocks, each with at most m labels. Since $q = O(\eta^{-1})$, $\tau = \eta/24$, and $\log(2/\beta) = O(\log(1/\eta))$, Eq. (10) gives $2m = O(\eta^{-3})$, independently of L. This proves the statistical and query bounds for the specified construction. □

A.2 Adaptive selections and query cost

Algorithm 3 summarizes how the shared predictor is used for each selected point, including cached labels and direct querying.

Algorithm 3 CALL-LIP

Require: Selected point x_t, costs c, eligible set $\mathcal{C}_t(x_t)$, cache $\mathcal{A}_{t-1}$ with inputs A_{t-1}, training set $\mathcal{L}_{t-1}$, and pool U_{t-1}
Require: Fixed preprocessing from Section A.1, performed once before active learning
Ensure: Label $\widehat{y}_t$, query cost C_t, and updated cache, training set, and pool
1: Set $R_t = \mathsf{B}(x_t) \setminus A_{t-1}$ and compute D_t by Eq. (3).
2: **if** $D_t \le c(x_t)$ **then**
3: Query and cache the labels at R_t.
4: Evaluate $\widehat{y}_t = \widetilde{f}(x_t)$ from the cache; set $C_t = D_t$.
5: **else**
6: Query and cache y_{x_t}; set $\widehat{y}_t = y_{x_t}$ and $C_t = c(x_t)$.
7: **end if**
8: Apply the training and pool updates in Section 3.1; return $\widehat{y}_t, C_t$ and the updated sets.

Completion of the proof of Theorem 1. Fix the complete sample, including its initially hidden labels, and the partition and deterministic fitting rules. They determine one function $\widetilde{f}$ and all sets $\mathsf{B}(x)$. The preceding argument gives an event of probability at least $1/2$ on which this function satisfies Eq. (4).

At any round, the selected point x_t may depend on earlier queried labels, predictions, and fitted models. If $D_t \le c(x_t)$, every missing label in $\mathsf{B}(x_t)$ is eligible and the algorithm queries all of them. Cached and newly queried labels then supply exactly the sample values defining $\widetilde{f}(x_t)$. The fixed fitting rules therefore return $\widehat{y}_t = \widetilde{f}(x_t)$. Otherwise the algorithm queries the selected point directly and makes no local estimate. Estimated labels never enter the cache. Induction over rounds shows that the cache always contains queried sample values and that every local estimate agrees with the same $\widetilde{f}$.

This reasoning holds for every selected sequence after fixing the preprocessing realization. Hence the same event on which the risk bound holds applies to all rounds, including adaptive selections, without a union bound over T. The guarantee is the population risk of $\widetilde{f}$; adaptive selection does not turn it into a separate error guarantee at each selected point.

For the cost statement, let $R_t = \mathsf{B}(x_t) \setminus A_{t-1}$. If $D_t \le c(x_t)$, the algorithm queries exactly these missing labels and pays $C_t = D_t$. If $D_t > c(x_t)$, including $D_t = +\infty$ when a missing label is ineligible, it queries only x_t and pays $C_t = c(x_t)$. The selected point lies in the acquisition pool, so its label has not already been queried. Thus $C_t = \min\{D_t, c(x_t)\}$ in both cases. This includes equality at the price comparison and an empty required set, which costs zero. Summing over rounds proves Eq. (5) on every realization, independently of the statistical event. Each previously queried label is excluded from R_t and incurs no additional charge.

Finally, if every missing label is eligible and has cost at most c_s, then

$$D_t = \sum_{z \in R_t} c(z) \le |R_t| c_s \le 2mc_s = b_\eta c_s.$$

Together with the equality for C_t at each round, this proves the sufficient price bound stated after the theorem. All costs in the theorem are incurred after the initial cache is supplied; an initialization cost, if charged, is added equally to both sides of the comparison. □

B Proofs for Capped Disagreement Search

For a single search, fix the selected point x, the response function $f^\star$, the class $\mathcal{F}$, the candidate distribution μ_x, prices $0 < c(z) \le c_{\max}$, the initial cache of queried labels, and an initial version space $V_0 \subseteq \mathcal{F}$ containing $f^\star$ and consistent with that cache. The randomness is in the fresh candidate

draws. All disagreement regions and events below are assumed measurable; this is automatic for a finite candidate pool.

For binary labels, $B_x(f^\star, r)$ contains the functions satisfying $\mu_x\{f \neq f^\star\} \leq r$, and $\mathrm{DIS}(V)$ contains the inputs where members of V disagree, as defined in Section 3.3. We use the cost-aware coefficient θ_x^c from Definition 2. Queried labels form a certificate when the surviving predictions guarantee the required accuracy at x. In the detailed analysis, δ bounds the probability that a cycle error bound fails, and α bounds the probability that cost exceeds its limit; Section B.7 combines them into the single failure probability used in the main theorem.

We first prove the detailed cost bound in Theorem 11, then derive the concise statement of Theorem 3. The proof has two ingredients. First, a sample bound ensures that the functions surviving each cycle disagree with the truth on only a small fraction of candidates. Their disagreement region then lies inside a fixed region defined by $f^\star$ and the bound on disagreement for that cycle. Second, we bound the cost of distinct uncached inputs visited in these fixed regions. This cost bounds the algorithm's purchases whenever the first event holds. We prove both ingredients and then account for direct fallback.

B.1 A CAP REQUIRING ONLY THE COST OF QUERYING x

Lemma 5 compares two choices for a fixed search: complete that search if it obtains a cheap certificate for the label at x, or query x directly. The comparison uses whichever choice costs less for the same sequence of draws. The price-threshold argument is the classical rent-or-buy principle, discussed by Dinitz et al. (2024, Section 1). Here the cost is the sum of the costs of new label queries, and an unsuccessful search has infinite certificate cost.

Lemma 5. *Couple capped and uncapped searches with the same initial cache, version space, finite draw horizon, draws, and query rule; only the cap changes the fallback decision. Let W_x be the uncapped surrogate cost until its stopping rule certifies the prediction at x, or $+\infty$ if this does not occur within the horizon. With $b = c(x)$, on every path,*

$$C_x \leq 2\min\{W_x, c(x)\}, \qquad C_x = W_x \quad \text{if } W_x \leq c(x). \tag{18}$$

Adaptation of rent-or-buy to Lemma 5. We follow the classical threshold argument (Dinitz et al., 2024, Section 1), with the costs of label queries as the successive costs and direct querying as the terminal purchase. We first show that the capped search makes the same purchases as the uncapped search until it stops. We then consider whether the uncapped search finds a certificate within the allowed number of draws. Write $d = c(x) > 0$ and fix the cache, truth, draws, and internal random seed. Both searches are then deterministic, with the same horizon and certificate test.

Before the first action, the two procedures have the same information. Suppose their histories agree before an action and neither has stopped. The next candidate and label request are determined by the same rule and random choices. If a new label is requested and the cap permits it, both observe the same exact response, update the same cache and version space, and apply the same certificate test. A cached response also gives identical updates without cost. Induction therefore shows that the capped procedure follows the uncapped one until it certifies an estimate of the label at x or invokes fallback. In particular, its paid queries form a prefix of the uncapped paid queries. This uses the assumption that the cap changes only the stopping decision, rather than the rule for proposing candidates.

Suppose $W_x \leq d$. By definition the uncapped procedure then obtains a certificate within the common horizon. Every paid prefix up to that certificate costs at most W_x, since all cost increments are nonnegative. If the capped procedure rejected a request in this prefix, the prefix including that request would cost more than d, contrary to $W_x \leq d$. It therefore reaches the same first certificate and pays exactly $C_x = W_x$. This includes an initial certificate, for which $W_x = 0$, and a certificate costing exactly d, since the cap rejects only a strict excess. Consequently $C_x \leq 2\min\{W_x, d\}$ in this case.

Now suppose $W_x > d$, including $W_x = +\infty$. The capped cost of surrogate queries S is at most d on every path. If the capped procedure returned a certificate, the same sequence of paid queries would give the uncapped search a certificate at that cost $S \leq d$, contradicting this case. Thus it queries x, either before an excessive purchase or at the horizon. Its total cost is

$$C_x = S + d \leq 2d = 2\min\{W_x, d\}.$$

These two cases prove the claim for every fixed realization, and hence also for randomized searches without any distributional assumption. In the realizable model, exact or tolerance-based elimination retains $f^\star$, so the certificate test has its stated prediction accuracy at x. Direct fallback returns the recorded label at x. □

The definition $W_x = +\infty$ on horizon failure is essential. If all inspected candidates have unanimous predictions while x remains unresolved, the uncapped search can spend zero without finding a certificate. The capped procedure still pays d at the horizon. Comparing with surrogate spending alone would incorrectly give zero cost; using W_x instead selects the direct query in this case.

The factor two is sharp for this cap. Take two candidates with prices $d-h$ and $2h$, where $0 < h < d/3$, and the four binary assignments to their labels. Let the label at x equal the second candidate's label, and fix the query order to be the first candidate followed by the second. The first query never certifies an estimate of the label at x; the second always does, so $W_x = d + h$. The cap pays $d - h$, rejects the second purchase, and queries x. Its ratio to the benchmark is

$$\frac{C_x}{\min\{W_x, d\}} = \frac{2d - h}{d} \longrightarrow 2 \quad \text{as } h \downarrow 0.$$

This comparison uses the fixed query order: querying the second candidate first would cost $2h$. This example explains why the lemma does not compare with the best unrestricted query policy.

B.2 Cycle lengths and the cost of distinct labels

We group the candidate draws into active learning cycles. These are the cycles within one search at x, not the outer rounds that select new points. Each cycle halves the upper bound on the surviving functions' disagreement with $f^\star$ under μ_x. Let $r_k = 2^{-k}$, $K = \lceil \log_2(1/r_*) \rceil$, $a = 64$, and $v \geq 1$ be the VC dimension. Given a bound on the probability that any cycle fails $\delta \in (0, 1)$, set

$$n_k = \left\lceil \frac{a}{r_k} \left(v \log \frac{a}{r_k} + \log \frac{K}{\delta} \right) \right\rceil, \qquad k = 1, \ldots, K. \tag{19}$$

The total candidate-draw budget in Algorithm 1 is $n = \sum_{k=1}^{K} n_k$. Partitioning this sequence into cycles is only for the proof; it does not reset the version space or cache. The proof below uses the standard realizable VC guarantee to justify the required number of draws. Once candidate error is at most r, queries occur only in $A_r = \mathrm{DIS}(B_x(f^\star, r))$. Let R_x be the initially uncached inputs in the finite support of μ_x, and let $p_z = \mu_x(\{z\})$. Define

$$N_z = \sum_{k=1}^{K} n_k \mathbf{1}\{z \in A_{r_{k-1}}\}, \qquad U_x = \sum_{z \in R_x} c(z)\big[1 - (1 - p_z)^{N_z}\big]. \tag{20}$$

The count N_z includes draws from cycles whose fixed disagreement region contains z. Independence gives a probability $1 - (1 - p_z)^{N_z}$ of visiting z at least once in those draws, with value zero when $N_z = 0$. Thus U_x is the expected total price of distinct initially uncached inputs visited in these regions. Whenever all cycle error bounds hold, the search queries only a subset of these inputs. In particular,

$$U_x \leq \min \left\{ \sum_{z \in R_x} c(z),\ \sum_{k=1}^{K} n_k \mu_x^c(A_{r_{k-1}}) \right\}. \tag{21}$$

The first bound charges for the entire uncached pool, and the second charges separately for each draw that could cause a query. The exact sum U_x accounts for reuse and can be smaller than either bound.

To allow for fluctuations around this mean, take $U \geq U_x$ and a failure probability $\alpha \in (0, 1)$, and set

$$B(U, \alpha) = U + \sqrt{2c_{\max} U \log(1/\alpha)} + \frac{2c_{\max}}{3} \log(1/\alpha). \tag{22}$$

Lemma 10 shows that the cost of visiting these distinct inputs is at most $B(U, \alpha)$ with probability at least $1 - \alpha$. A cap $B(U, \alpha) \leq b \leq c(x)$ therefore covers the query cost on this event while limiting exploration to the direct-query price. If the interval is empty, the theorem does not establish savings; directly querying x costs $c(x)$.

These expressions use a radius r_* satisfying Eq. (6) and a bound U. Both r_* and U_x depend on the unknown truth. The uncached pool price in Eq. (21) is a conservative choice of U. A cap $b = c(x)$ needs no estimate of U_x and has the guarantee in Lemma 5.

A threshold instance with explicit savings. Consider four candidates $z_1 < z_2 < x < z_3 < z_4$ and five threshold functions $f_j(z_i) = \mathbf{1}\{i > j\}$ and $f_j(x) = \mathbf{1}\{j \le 2\}$, $j = 0, \dots, 4$. Let $f^\star = f_2$, let μ_x be uniform over the four candidates, and take an empty initial cache. Each surrogate costs one and $c(x) = 20$. The class has VC dimension one, and $r_* = 1/8$ satisfies Eq. (6). The disagreement region is empty for $r < 1/4$, contains z_2, z_3 for $1/4 \le r < 1/2$, and contains all four candidates for $r \ge 1/2$. Thus $\theta_x^c(1/8) = 2$. For the theorem's draw schedule, Eq. (21) gives $U_x \le 4$. Allocating failure probability 0.05 to each of localization and cost concentration gives

$$B(4, 0.05) = 4 + \sqrt{8 \log 20} + \frac{2}{3} \log 20 < 11.$$

With total failure probability 0.1, the cap $b = 11$ therefore gives $\mathbb{E}C_x \le 11 + 0.1(20) = 13 < 20$ by Theorem 11. This illustrates sufficient savings under realizability, without certifying the molecular committee.

B.3 A UNIFORM SAMPLE BOUND FOR BINARY CLASSES

The following standard sample bound controls all hypotheses consistent with one batch. We state it in the finite-pool form needed here; finiteness ensures measurability of the events.

Lemma 6 (Consistent-class tail bound (Blumer et al., 1989, Theorem A2.1 and Proposition A2.1))**.** *Let $\mathcal{H}$ be a binary class of VC dimension $v \ge 1$, let $h^\star \in \mathcal{H}$, and draw n iid inputs from a distribution μ on a finite set. If $2n \ge v$ and $r \in (0, 1)$, then*

$$\Pr\{\exists h \in \mathcal{H} : \; h(Z_i) = h^\star(Z_i) \; (i \le n), \; \mu\{h \ne h^\star\} > r\} \le 2 \left(\frac{2en}{v}\right)^v e^{-nr/4}. \tag{23}$$

The cited bound has $2^{-nr/2}$ in place of $e^{-nr/4}$ and thus implies this form. We next check the explicit cycle length used by our algorithm.

Corollary 7 (Cycle length for realizable VC localization)**.** *Let μ be a distribution on a finite candidate pool, let $\mathcal{H}$ be a binary class of VC dimension $v \ge 1$, let $h^\star \in \mathcal{H}$, and let $r, \eta \in (0, 1)$. There is a universal constant $a \ge e$ such that, for an iid sample of size*

$$n \ge \frac{a}{r} \left(v \log \frac{a}{r} + \log \frac{1}{\eta} \right)$$

from μ, with probability at least $1 - \eta$ every $h \in \mathcal{H}$ agreeing with $h^\star$ on the sample satisfies $\mu\{h \ne h^\star\} \le r$. The choice $a = 64$ suffices.

Proof. Set $a = 64$, $\ell = \log(a/r)$, $d = \log(1/\eta)$, $A = v\ell + d$, and $n_0 = \lceil aA/r \rceil$. Then $n_0 r \ge aA \ge 8$ and $2n_0 \ge v$, so Eq. (23) applies. Also $aA/r \ge 1$, hence $n_0 \le 2aA/r$. Using $\log u \le u$ for $u > 0$ and $\log(4e) \le \ell$, we obtain

$$\begin{aligned} \log(2en_0/v) &\le \log(4ea/r) + \log(A/v) \\ &\le \log(4e) + \ell + (\ell + d/v) \le 3\ell + d/v. \end{aligned}$$

The logarithm of the upper bound in Eq. (23) at n_0 is consequently at most

$$\begin{aligned} \log 2 + v \log(2en_0/v) - n_0 r/4 &\le \log 2 + 3v\ell + d - 16(v\ell + d) \\ &\le -12v\ell - 15d \le -d, \end{aligned}$$

where $\log 2 \le v\ell$ was used in the second line. The failure probability is at most $e^{-d} = \eta$. If $n \ge n_0$, any hypothesis consistent with all n points is consistent with the first n_0 points. The same event therefore suffices for every larger sample size. This proves the stated choice of a. □

B.4 CONCENTRATION OF A BOUNDED COST SUM

We use Bernstein's inequality to bound how far the cost sum can exceed its mean. We state its exponential-moment form as well as the resulting probability bound. The former lets us later handle the dependence between indicators of which candidates are visited.

Lemma 8 (Bernstein's inequality (Boucheron et al., 2013, Section 2.8)). *If $X_1, \dots, X_M$ are independent, centered random variables with $|X_i| \le c_{\max}$, $c_{\max} > 0$, and $\sigma^2 = \sum_i \mathbb{E}X_i^2$, then, for $0 \le \lambda < 3/c_{\max}$,*

$$\log \mathbb{E} \exp\left(\lambda \sum_i X_i\right) \le \frac{\lambda^2 \sigma^2}{2(1 - \lambda c_{\max}/3)}. \tag{24}$$

More generally, any centered random variable S satisfying this exponential-moment bound obeys, for $s > 0$,

$$\Pr\{S > \sqrt{2\sigma^2 s} + 2c_{\max} s/3\} \le e^{-s}.$$

Consequently, if independent $Y_i \in [0, c_{\max}]$ satisfy $\sum_i \mathbb{E}Y_i \le U$, then $\Pr\{\sum_i Y_i > B(U, \alpha)\} \le \alpha$ for $\alpha \in (0, 1)$.

For the last statement, the required substitution is $X_i = Y_i - \mathbb{E}Y_i$ and $\sigma^2 \le \sum_i \mathbb{E}Y_i^2 \le c_{\max} U$, with $s = \log(1/\alpha)$. The displayed threshold is a valid, slightly looser form of the standard Bernstein threshold.

B.5 Counting distinct uncached labels

With caching, each candidate is charged once, regardless of how often it is drawn. We can therefore view candidates as bins and charge the price of each occupied bin. Although draws are independent, the events that different bins are occupied are dependent. The following result bounds their joint exponential moment by the product of the individual moments, allowing us to apply Bernstein's inequality.

Lemma 9 (Occupancy dependence (Dubhashi & Ranjan, 1998, Section 2)). *Assign independent draws to finitely many bins, allowing a different bin distribution for each draw. Let B_z be the count in bin z and $I_z = \mathbf{1}\{B_z > 0\}$. The indicators are negatively associated: for nondecreasing functions g, h of disjoint groups of indicators,*

$$\mathbb{E}[g(I_A)h(I_B)] \le \mathbb{E}[g(I_A)]\mathbb{E}[h(I_B)], \qquad A \cap B = \varnothing.$$

In particular, for nonnegative weights c_z and $\lambda \ge 0$,

$$\mathbb{E} \exp\left(\lambda \sum_z c_z I_z\right) \le \prod_z \mathbb{E} e^{\lambda c_z I_z}.$$

Theorem 13 of the cited preprint gives negative association for the bin counts. Proposition 7 preserves this property when each count is replaced by its nonzero indicator. Lemma 2 then gives the displayed product bound. We use these three results directly.

Lemma 10. *Let $T_1, \dots, T_m$ be independent random variables taking values in a finite set $R \cup \{\perp\}$, with possibly different distributions. The symbol $\perp$ denotes a discarded draw. For $z \in R$, let $p_{jz} = \Pr(T_j = z)$, $I_z = \mathbf{1}\{T_j = z \text{ for some } j\}$, and $0 < c_z \le c_{\max}$. Then, for $Y = \sum_{z \in R} c_z I_z$,*

$$\mathbb{E}Y = \sum_{z \in R} c_z \left[1 - \prod_{j=1}^{m} (1 - p_{jz})\right]. \tag{25}$$

For every $U \ge \mathbb{E}Y$ and $\alpha \in (0, 1)$,

$$\Pr\{Y > B(U, \alpha)\} \le \alpha.$$

An empty product is one. The statement includes $R = \varnothing$ and $m = 0$.

Proof. For each z, independence across draws gives $\Pr(I_z = 0) = \prod_j (1 - p_{jz})$. Linearity of expectation proves Eq. (25). Include $\perp$ as an additional bin and then retain only the indicators indexed by R. The draws meet the conditions of Lemma 9, including when their distributions differ.

Set $X_z = c_z(I_z - \mathbb{E}I_z)$ and $\sigma_0^2 = \sum_z \mathbb{E}X_z^2$. Centering the product-moment inequality gives

$$\mathbb{E}e^{\lambda(Y - \mathbb{E}Y)} \le \prod_z \mathbb{E}e^{\lambda X_z}, \qquad \sigma_0^2 \le \sum_z c_z^2 \mathbb{E}I_z \le c_{\max} U.$$

Each X_z is centered and bounded in magnitude by $c_{\max}$. Apply the one-variable case of Lemma 8 to each factor. For $0 \le \lambda < 3/c_{\max}$ this yields

$$\log \mathbb{E}e^{\lambda(Y-\mathbb{E}Y)} \le \sum_z \log \mathbb{E}e^{\lambda X_z} \le \frac{\lambda^2 \sigma_0^2}{2(1-\lambda c_{\max}/3)}.$$

The tail implication in the same lemma now applies to $Y - \mathbb{E}Y$, with $\sigma_0^2 \le c_{\max} U$ and $\mathbb{E}Y \le U$. Taking $s = \log(1/\alpha)$ proves the claim. If $\sigma_0^2 = 0$, the weighted indicators are constant almost surely and $Y = \mathbb{E}Y \le U$; empty sums are included in this case. □

For disagreement search, assign a draw to bin z only when z is initially uncached and belongs to that active learning cycle's fixed disagreement region; send other draws to $\perp$. The assignments are independent across draws because these regions are fixed before sampling. Thus the lemma applies to the binary and scalar searches without treating their adaptive purchase decisions as independent.

B.6 Proof of the binary cost guarantee

Theorem 11. *In the finite-pool binary setting of Section B.2, assume Eq.* (6) *and choose $U \ge U_x$ and $\delta, \alpha > 0$ with $p = \delta + \alpha < 1$. With a cap $B(U, \alpha) \le b \le c(x)$ and $n = \sum_{k=1}^{K} n_k$ candidate draws from Eq.* (19)*, Algorithm 1 returns the correct label at x on every path and completes without fallback with probability at least $1 - p$. Its total cost satisfies*

$$C_x \le b + c(x), \qquad \mathbb{E}C_x \le \min\{b + pc(x),\ U + \delta b + pc(x)\}. \tag{26}$$

Moreover,

$$U_x \le 2a\theta_x^c(r_*) \sum_{k=1}^{K} \left(v \log \frac{a}{r_k} + \log \frac{K}{\delta} \right) + Kc_{\max}. \tag{27}$$

In particular, $b < (1-p)c(x)$ guarantees lower expected cost than a direct query.

Proof of Theorem 11. Write $d_x = c(x)$ and $p = \delta + \alpha$. All parameters are fixed before drawing candidates, including $U \ge U_x$ and $B(U, \alpha) \le b \le d_x$. Recall that

$$r_k = 2^{-k}, \qquad K = \lceil \log_2(1/r_*) \rceil.$$

Since $0 < r_* \le 1/2$, $K \ge 1$, $r_K \le r_*$, and $r_{k-1} > r_*$ for $1 \le k \le K$. We may use the constant $a = 64$ from Corollary 7 in the cycle lengths.

Draw in advance an independent array $Z_{ki} \sim \mu_x$, where $k = 1, \dots, K$ and $i = 1, \dots, n_k$. Run an auxiliary search that uses these draws, the same initial cache and version space, and the same query and elimination rules, but never applies the spending cap. Continue this auxiliary search through all K cycles even if agreement at x occurs earlier. This defines every cycle's version space on every realization.

Let $V_{k,i}$ be its version space after draw i of cycle k, let $V_{k,0} = V_{k-1}$, and let $V_k = V_{k,n_k}$, with the given initial space denoted V_0. At every update the true function survives: its response equals the queried label. Induction therefore gives

$$f^\star \in V_{k,i} \subseteq V_{k,i-1} \subseteq \mathcal{F}. \tag{28}$$

The actual capped search follows the same draws and updates until it stops, either by certification or by direct fallback. In particular, its paid surrogate queries form a prefix of the auxiliary search's paid queries.

Fix a draw Z_{ki}. If it is in current disagreement, its exact response is retrieved from the cache or purchased, and the update retains only functions agreeing with $f^\star$ there. If it is outside current disagreement, every current function has the same prediction there. Since $f^\star$ is one of those functions by Eq. (28), that prediction already equals $f^\star(Z_{ki})$. No query is required to ensure consistency at this draw. Subsequent updates only remove functions. Hence, at the end of cycle k,

$$f \in V_k \quad \Longrightarrow \quad f(Z_{ki}) = f^\star(Z_{ki}) \quad \text{for every } i = 1, \dots, n_k. \tag{29}$$

Repeated candidates satisfy the same implication, while caching avoids a repeated purchase.

Let G_k be the event that every function in the full fixed class $\mathcal{F}$ consistent with the entire kth batch has μ_x-error at most r_k. The batch size in Eq. (19) and Corollary 7, with $\eta = \delta/K$, imply

$$\Pr(G_k^c) \le \delta/K.$$

The corollary applies to the fixed class and therefore covers the data-dependent survivors in Eq. (29). On G_k,

$$V_k \subseteq B_x(f^\star, r_k).$$

Set $G = \bigcap_{k=1}^{K} G_k$. The union bound gives $\Pr(G^c) \le \delta$. On G, $V_K \subseteq B_x(f^\star, r_K) \subseteq B_x(f^\star, r_*)$, and the transfer assumption Eq. (6) implies that every member of V_K predicts $f^\star(x)$ at x. The auxiliary search has therefore reached agreement at x by the end of cycle K.

For each cycle define $A_k = \mathrm{DIS}(B_x(f^\star, r_{k-1}))$. For $z \in R_x$, put

$$I_z = \mathbf{1}\{Z_{ki} = z \text{ for some } (k,i) \text{ with } z \in A_k\}, \qquad Y = \sum_{z \in R_x} c(z) I_z.$$

The notation A_k denotes the region $A_{r_{k-1}}$ from Section B.2. These regions and the initial uncached set depend on the class, true function, candidate distribution, and cache fixed before search, not on the sampled array. For a candidate z, exactly N_z draws occur in cycles whose region contains it. Each of those draws independently visits z with probability p_z. Hence

$$\mathbb{E}Y = \sum_{z \in R_x} c(z)\big[1 - (1-p_z)^{N_z}\big] = U_x \le U. \tag{30}$$

When $N_z = 0$, the event is empty and its probability is zero, including when $p_z = 1$.

On G, the version space at the beginning of cycle k lies in $B_x(f^\star, r_{k-1})$. For $k = 1$ this holds without an event, since $r_0 = 1$ and every binary hypothesis has error at most one. For $k > 1$ it follows from G_{k-1}. By monotonicity within the cycle,

$$\mathrm{DIS}(V_{k,i-1}) \subseteq \mathrm{DIS}(B_x(f^\star, r_{k-1})) = A_k.$$

Every paid query on G is initially uncached and is drawn in a cycle k with $z \in A_k$, so its indicator I_z is one. Each such label incurs a cost at most once. If W is the auxiliary search's total paid surrogate cost, this proves

$$G \quad \Longrightarrow \quad W \le Y. \tag{31}$$

We now bound Y using the original independent draws, before conditioning on G. Regard each candidate as a bin. Assign draw Z_{ki} to that bin if it lies in $R_x \cap A_k$; otherwise assign it to the discard bin $\perp$. The sets used for this assignment are fixed, so the assignments remain independent across draws. This is the setting of Lemma 10, even though the resulting indicators of occupied bins are dependent.

Apply Lemma 10 to Y, using Eq. (30). Let $H = \{Y \le B(U, \alpha)\}$. Then $\Pr(H^c) \le \alpha$. On $G \cap H$,

$$W \le Y \le B(U, \alpha) \le b.$$

Every prefix of the auxiliary search's paid queries costs at most W, because prices are positive. If the capped search were to reject its next purchase on this event, that purchase together with its preceding paid prefix would be a prefix of the auxiliary search costing more than b, contradicting $W \le b$. Thus the cap cannot force fallback on $G \cap H$. The event G bounding disagreement also ensures that agreement occurs by cycle K, so the final fallback is unnecessary there.

Let $F \in \{0, 1\}$ be the actual fallback indicator. We have shown

$$\{F = 1\} \subseteq G^c \cup H^c, \qquad \Pr(F = 1) \le \delta + \alpha = p.$$

This uses a union bound only; independence of G and H is not needed.

The actual search, like its auxiliary counterpart, retains $f^\star$ at every update. If it stops by agreement, the common label at x is therefore $f^\star(x)$. If it falls back, its queried label is $f^\star(x)$. This proves correctness on every realization, including those outside $G \cap H$.

Let S be its total paid surrogate cost. The cap is checked before a new purchase, so $0 \le S \le b$ on every path. The complete cost is $C_x = S + F d_x$. It follows that

$$C_x \le b + d_x, \qquad \mathbb{E}C_x \le b + p d_x.$$

For the second expectation bound, the capped search makes only an initial sequence of the uncapped search's purchases, giving $S \le W$ on every path. On G, Eq. (31) gives $S \le Y$. Splitting the expectation over G and its complement therefore yields

$$\begin{aligned}\mathbb{E}S &= \mathbb{E}[S\mathbf{1}\{G\}] + \mathbb{E}[S\mathbf{1}\{G^c\}] \\ &\le \mathbb{E}[Y\mathbf{1}\{G\}] + b\Pr(G^c) \\ &\le \mathbb{E}Y + \delta b \le U + \delta b.\end{aligned}$$

Here $Y \ge 0$ justifies replacing its restricted expectation by its full expectation. No independence between S, F, Y, or G is used. Adding $d_x\mathbb{E}F \le pd_x$ gives

$$\mathbb{E}C_x \le U + \delta b + pd_x.$$

Taking the smaller of the two valid bounds proves Eq. (26).

For every integer $N \ge 0$ and $p \in [0,1]$, the probability of at least one success in N independent trials is at most Np and at most one. Therefore

$$U_x \le \sum_{z\in R_x} c(z)\min\{1, N_z p_z\} \le \min\left\{\sum_{z\in R_x} c(z),\ \sum_{k=1}^{K} n_k \mu_x^c(A_k)\right\}.$$

For the second term we enlarged R_x to the whole candidate support and interchanged finite sums. This proves Eq. (21). It remains to bound the sum that counts each draw separately.

Let $L_k = v\log(a/r_k) + \log(K/\delta)$. Since $r_{k-1} \in [r_*, 1]$, the definition of $\theta_x^c(r_*)$ gives

$$\mu_x^c(A_k) \le \theta_x^c(r_*) r_{k-1}.$$

Also $\mu_x^c(A_k) \le c_{\max}$ because μ_x is a probability measure. The ceiling in the cycle length satisfies $n_k \le aL_k/r_k + 1$. Multiplying these bounds in two separate terms gives

$$\begin{aligned}n_k\mu_x^c(A_k) &\le \frac{aL_k}{r_k}\mu_x^c(A_k) + \mu_x^c(A_k) \\ &\le aL_k\theta_x^c(r_*)\frac{r_{k-1}}{r_k} + c_{\max} \\ &= 2aL_k\theta_x^c(r_*) + c_{\max}.\end{aligned}$$

Summation over $k = 1, \dots, K$ proves Eq. (27). Finally, if $b < (1-p)d_x$, the first expectation bound gives $\mathbb{E}C_x \le b + pd_x < d_x$, the stated sufficient saving condition. □

B.7 Sufficient budget in the main binary theorem

Proof of Theorem 3. Fix the desired failure probability $\delta \in (0,1)$. Use $\delta/2$ as the bound on the probability that any cycle fails in Eq. (19), and evaluate U_x in Eq. (20) with those cycle lengths. Define

$$b_x(\delta) = B(U_x, \delta/2), \qquad n = \sum_{k=1}^{K} n_k. \tag{32}$$

This specifies both the budget and the draw horizon used in the main statement. Applying Theorem 11 with $U = U_x$ and with each of its two failure probabilities bounded by $\delta/2$ gives correctness on every path and fallback probability at most δ. For every cap $b_x(\delta) \le b \le c(x)$, its first expectation bound gives Eq. (9). If $b < (1-\delta)c(x)$, that expectation is strictly below $c(x)$.

It remains to justify the stated order of $b_x(\delta)$. Since $K = \lceil \log_2(1/r_*) \rceil$, we have $r_K > r_*/2$ and, for $k \le K$, $\log(a/r_k) \le \log(2a/r_*)$. Substituting these inequalities and the allowance $\delta/2$ into Eq. (27) yields

$$U_x \le 2a\theta_x^c(r_*)K\left(v\log\frac{2a}{r_*} + \log\frac{2K}{\delta}\right) + Kc_{\max}.$$

For $U \ge 0$ and $\ell \ge 0$, the inequality $\sqrt{2c_{\max}U\ell} \le U + c_{\max}\ell$ gives

$$B(U, \delta/2) \le 2U + \frac{5}{3}c_{\max}\log\frac{2}{\delta}.$$

Consequently,

$$b_x(\delta) \le 4a\theta_x^c(r_*)K\left(v\log\frac{2a}{r_*}+\log\frac{2K}{\delta}\right)+2Kc_{\max}+\frac{5}{3}c_{\max}\log\frac{2}{\delta}. \tag{33}$$

Here $a = 64$ is universal, $v \ge 1$, and $K = O(\log(1/r_*))$. All factors beyond $v\theta_x^c(r_*) + c_{\max}$ are bounded by logarithmic factors in $1/r_*$ and $1/\delta$. This proves Eq. (8). The exact budget in Eq. (32) retains the smaller cost sum obtained from label reuse, whereas Eq. (33) bounds it through the disagreement coefficient alone. □

B.8 Scalar guarantees and Lipschitz transfer

For a finite class $\mathcal{F}$ of N bounded scalar functions, define

$$e_{x,\gamma}(f,g) = \mu_x\{|f-g| > \gamma\}, \qquad B_{x,\gamma}(f^\star, r) = \{f \in \mathcal{F} : e_{x,\gamma}(f, f^\star) \le r\},$$
$$\mathrm{DIS}_\gamma(V) = \{z : \mathrm{diam}\{f(z) : f \in V\} > \gamma\}.$$

These replace exact disagreement by differences greater than γ. Define $\theta_{x,\gamma}^c$ as in Definition 2, using $B_{x,\gamma}$ and DIS_γ. The next result extends the binary guarantee when small disagreement with $f^\star$ on candidates implies accuracy within ε at x.

Proposition 12. *Suppose μ_x has finite support, $f^\star \in \mathcal{F}$, and, for $\varepsilon > 0$ and $r_* \in (0, 1/2]$,*

$$\sup_{f \in B_{x,\gamma}(f^\star, r_*)} |f(x) - f^\star(x)| \le \varepsilon. \tag{34}$$

For $\delta \in (0,1)$, the scalar draw schedule in Section B.9.1 admits a sufficient budget

$$b_{x,\gamma}(\delta) = \widetilde{O}\big(\theta_{x,\gamma}^c(r_*)\log(2N) + c_{\max}\big).$$

If $b_{x,\gamma}(\delta) \le b \le c(x)$, the scalar search has prediction error at most ε on every path, avoids fallback with probability at least $1-\delta$, and obeys Eq. (9)*.*

Lipschitz regularity gives a concrete way to satisfy the transfer condition: agreement on nearby candidates limits how far predictions can differ at x.

Proposition 13. *If every $f \in \mathcal{F}$ is L-Lipschitz on a metric space $(\mathcal{X}, d)$ and $q_x(s) = \mu_x\{z : d(x,z) \le s\}$ for $s \ge 0$, then*

$$e_{x,\gamma}(f, f^\star) \le r < q_x(s) \quad \Longrightarrow \quad |f(x) - f^\star(x)| \le \gamma + 2Ls. \tag{35}$$

Thus $r_* < q_x(s)$ and $\gamma + 2Ls \le \varepsilon$ imply Eq. (34)*.*

Here $r < q_x(s)$ ensures agreement within γ at some candidate no farther than s from x; the two Lipschitz bounds then add at most $2Ls$. Section B.9 gives the proofs.

B.9 Scalar localization and prediction accuracy at x

Fix a finite class $\mathcal{F}$ of $N \ge 1$ bounded real-valued functions containing $f^\star$. Let $\gamma \ge 0$ be the response tolerance used in elimination, and $\varepsilon > 0$ the desired prediction accuracy at x. The quantity $e_{x,\gamma}(f,g) = \mu_x\{|f-g| > \gamma\}$ is the probability that the two functions differ by more than the allowed response tolerance. We control it by bounding the chance that a sample misses the inputs where this happens.

Lemma 14 (Finite-class localization (Shalev-Shwartz & Ben-David, 2014, Section 2.3.1))**.** *For $r \in (0,1)$ and $n \ge 1$ iid inputs $Z_1, \dots, Z_n \sim \mu_x$, the probability that some $f \in \mathcal{F}$ satisfies both*

$$e_{x,\gamma}(f, f^\star) > r \quad \text{and} \quad |f(Z_i) - f^\star(Z_i)| \le \gamma \text{ for all } i$$

is at most Ne^{-nr}.

This is the standard finite-class consistency bound applied to the binary functions $z \mapsto \mathbf{1}\{|f(z) - f^\star(z)| > \gamma\}$, with the identically zero function as truth. The transformed class has at most N members and contains that truth because $f^\star \in \mathcal{F}$. Thus the scalar tolerance changes the error sets, while the cited sampling bound applies directly.

Proposition 15. *In the finite-pool scalar setting of Proposition 12, assume Eq.* (34)*. For* $r_k = 2^{-k}$, $K = \lceil \log_2(1/r_*) \rceil$*, and* $\delta \in (0,1)$*, use*

$$n_k = \left\lceil r_k^{-1} \log \frac{NK}{\delta} \right\rceil, \qquad n = \sum_{k=1}^{K} n_k. \tag{36}$$

Let U_x *be the sum in Eq.* (20) *with these cycle lengths and* $A_r = \mathrm{DIS}_\gamma(B_{x,\gamma}(f^\star, r))$*. If* $U \geq U_x$*,* $\alpha > 0$*,* $p = \delta + \alpha < 1$*, and* $B(U,\alpha) \leq b \leq c(x)$*, the scalar search has error at most* ε *on every path, avoids fallback with probability at least* $1 - p$*, and obeys Eq.* (26)*. Moreover,*

$$U_x \leq 2K\theta^c_{x,\gamma}(r_*) \log(NK/\delta) + Kc_{\max}. \tag{37}$$

Proof of Proposition 15. Use $0 < r_* \leq 1/2$, $r_k = 2^{-k}$, and $K = \lceil \log_2(1/r_*) \rceil$ as in the binary theorem. Fix an initial $V_0 \subseteq \mathcal{F}$ containing $f^\star$, an exact cache, and all costs before search. Define

$$L_N = \log(NK/\delta), \qquad n_k = \lceil L_N / r_k \rceil, \qquad A^\gamma_k = \mathrm{DIS}_\gamma(B_{x,\gamma}(f^\star, r_{k-1})).$$

For each initially uncached $z \in R_x$, let $N^\gamma_z = \sum_k n_k \mathbf{1}\{z \in A^\gamma_k\}$. Define

$$U^\gamma_x = \sum_{z \in R_x} c(z)\big[1 - (1-p_z)^{N^\gamma_z}\big], \qquad p_z = \mu_x(\{z\}),$$

with a zero bracket when $N^\gamma_z = 0$. This is the scalar quantity denoted U_x in the proposition. The cap uses a fixed $U \geq U^\gamma_x$. Since $N, K \geq 1$ and $\delta < 1$, $L_N > 0$ and all cycle lengths are positive integers.

Use the same capped/uncapped coupling as in the binary proof, now querying when the current prediction diameter at a candidate exceeds γ. Upon receiving the exact response $y_z = f^\star(z)$, retain

$$\{f \in V : |f(z) - y_z| \leq \gamma\}.$$

The true function always survives since its difference from y_z is zero. Version spaces remain nonempty and only shrink.

At a queried draw, the update ensures tolerance γ for every survivor. At an unqueried draw, the current diameter is at most γ. Because $f^\star$ belongs to that class, every current function is then within γ of $f^\star$ at the draw. Later elimination preserves this property. Thus every function surviving a cycle satisfies the empirical inequalities of Lemma 14 on every draw of that cycle, including repeated and cached draws.

Let G^γ_k be the event that every function in the full class satisfying that cycle's empirical inequalities has probability $e_{x,\gamma}(f, f^\star)$ at most r_k. The finite-class lemma gives

$$\Pr((G^\gamma_k)^c) \leq Ne^{-n_k r_k} \leq Ne^{-L_N} = \delta/K.$$

On $G^\gamma = \bigcap_{k=1}^K G^\gamma_k$, whose complement has probability at most δ, the uncapped version spaces satisfy

$$V_k \subseteq B_{x,\gamma}(f^\star, r_k).$$

Since $r_K \leq r_*$, assumption Eq. (34) implies

$$f(x) \in [f^\star(x) - \varepsilon, f^\star(x) + \varepsilon] \quad \text{for every } f \in V_K.$$

The diameter of predictions at x is therefore at most 2ε by the end of cycle K.

For the independent cycle draws define

$$I^\gamma_z = \mathbf{1}\{Z_{ki} = z \text{ for some } (k,i) \text{ with } z \in A^\gamma_k\}, \qquad Y^\gamma = \sum_{z \in R_x} c(z) I^\gamma_z.$$

The regions and initial cache are fixed before the draws. The occupancy mean formula gives $\mathbb{E}Y^\gamma = U^\gamma_x \leq U$. At the first cycle, $V_0 \subseteq B_{x,\gamma}(f^\star, 1)$, because a probability is at most one. On G^γ, the corresponding inclusion with radius r_{k-1} holds at entry to every later cycle. Prediction diameters can only decrease when the class shrinks, so

$$\mathrm{DIS}_\gamma(V_{k,i-1}) \subseteq A^\gamma_k \quad \text{on } G^\gamma.$$

On G^γ, every paid query is initially uncached and occurs in a cycle k with $z \in A_k^\gamma$. Since each label incurs a cost once, the full uncapped paid cost satisfies $W^\gamma \le Y^\gamma$ on G^γ. Sending draws outside $R_x \cap A_k^\gamma$ to $\perp$ gives the independent draws in $R_x \cup \{\perp\}$ required by Lemma 10.

Lemma 10 gives an event $H^\gamma = \{Y^\gamma \le B(U, \alpha)\}$ with failure probability at most α. On $G^\gamma \cap H^\gamma$, all uncapped purchases together cost at most b. Every paid prefix also costs at most b, so the capped procedure follows that search until a valid stop based on prediction agreement. The event G^γ ensures such a stop by cycle K. Thus the fallback indicator obeys $\Pr(F = 1) \le \delta + \alpha = p$.

Always $S \le b$, and on G^γ the cost of the capped search satisfies $S \le W^\gamma \le Y^\gamma$. Therefore

$$\begin{aligned}\mathbb{E}S &\le \mathbb{E}[Y^\gamma \mathbf{1}\{G^\gamma\}] + b\Pr((G^\gamma)^c) \\ &\le U + \delta b.\end{aligned}$$

Combining this with the unconditional bound $\mathbb{E}S \le b$ and the total cost $C_x = S + Fc(x)$ gives

$$C_x \le b + c(x), \qquad \mathbb{E}C_x \le \min\{b + pc(x), U + \delta b + pc(x)\}.$$

At any stop based on prediction agreement, let $\ell = \min_{f \in V} f(x)$ and $u = \max_{f \in V} f(x)$. These extrema exist because V is finite and nonempty. Since $f^\star$ remains in V, we have $f^\star(x) \in [\ell, u]$, and the stopping rule gives $u - \ell \le 2\varepsilon$. Consequently

$$\left| \frac{\ell + u}{2} - f^\star(x) \right| \le \frac{u - \ell}{2} \le \varepsilon.$$

Direct fallback has zero prediction error at x. Thus accuracy holds also when the disagreement or cost bound fails.

Counting each draw separately gives $U_x^\gamma \le \sum_k n_k \mu_x^c(A_k^\gamma)$, by the same union bound as in the binary proof. For every cycle, $r_{k-1} \in [r_*, 1]$, so

$$\mu_x^c(A_k^\gamma) \le \theta_{x,\gamma}^c(r_*) r_{k-1}, \qquad \mu_x^c(A_k^\gamma) \le c_{\max}.$$

Using $n_k \le L_N / r_k + 1$ and $r_{k-1} = 2r_k$ yields

$$\begin{aligned}n_k \mu_x^c(A_k^\gamma) &\le \frac{L_N}{r_k} \theta_{x,\gamma}^c(r_*) r_{k-1} + c_{\max} \\ &= 2\theta_{x,\gamma}^c(r_*) L_N + c_{\max}.\end{aligned}$$

Summing over K cycles proves Eq. (37). The proof also covers $\gamma = 0$ and $N = 1$; in the latter case the initial diameter of predictions at x is already zero and the actual search costs zero. □

B.9.1 Sufficient budget in the main scalar proposition

Proof of Proposition 12. Use $\delta/2$ in the scalar cycle lengths Eq. (36), and let U_x^γ denote the resulting cost sum in Eq. (20) with scalar disagreement regions. Set

$$b_{x,\gamma}(\delta) = B(U_x^\gamma, \delta/2), \qquad n = \sum_{k=1}^{K} n_k. \tag{38}$$

Apply Proposition 15 with $U = U_x^\gamma$ and both failure probabilities bounded by $\delta/2$. For $b_{x,\gamma}(\delta) \le b \le c(x)$, this gives prediction error at most ε on every path, fallback probability at most δ, and Eq. (9).

With this choice of cycle lengths, Eq. (37) gives

$$U_x^\gamma \le 2K\theta_{x,\gamma}^c(r_*) \log \frac{2NK}{\delta} + Kc_{\max}.$$

Using $B(U, \delta/2) \le 2U + (5/3)c_{\max} \log(2/\delta)$ as in the binary proof yields

$$b_{x,\gamma}(\delta) \le 4K\theta_{x,\gamma}^c(r_*) \log \frac{2NK}{\delta} + 2Kc_{\max} + \frac{5}{3} c_{\max} \log \frac{2}{\delta}. \tag{39}$$

Since $K = O(\log(1/r_*))$ and $N \ge 1$, this is $\widetilde{O}(\theta_{x,\gamma}^c(r_*) \log(2N) + c_{\max})$, hiding only logarithmic factors in $1/r_*$ and $1/\delta$. As before, the exact budget can be smaller because it counts each queried label once. □

B.9.2 LIPSCHITZ REGULARITY IMPLIES TRANSFER

Proof of Proposition 13. Let $B_d(x,s) = \{z : d(x,z) \le s\}$, so $q_x(s) = \mu_x(B_d(x,s))$. Suppose, for contradiction, that $e_{x,\gamma}(f, f^\star) \le r < q_x(s)$ but $|f(x) - f^\star(x)| > \gamma + 2Ls$. For every $z \in B_d(x,s)$, insert $f(z)$ and $f^\star(z)$ between the two possible labels at x to obtain

$$|f(x) - f^\star(x)| \le |f(x) - f(z)| + |f(z) - f^\star(z)| + |f^\star(z) - f^\star(x)|.$$

Both outside terms are bounded by $Ld(x,z)$ because both functions are L-Lipschitz. Rearranging therefore gives

$$\begin{aligned} |f(z) - f^\star(z)| &\ge |f(x) - f^\star(x)| - 2Ld(x,z) \\ &> \gamma + 2Ls - 2Ld(x,z) \ge \gamma. \end{aligned}$$

Thus every input in the metric ball satisfies $|f(z) - f^\star(z)| > \gamma$, and

$$e_{x,\gamma}(f, f^\star) \ge \mu_x(B_d(x,s)) = q_x(s) > r,$$

a contradiction. This proves Eq. (35).

For the final implication, take any $f \in B_{x,\gamma}(f^\star, r_*)$. Its error probability is at most r_*. If $r_* < q_x(s)$, the proved implication gives $|f(x) - f^\star(x)| \le \gamma + 2Ls$. The additional condition $\gamma + 2Ls \le \varepsilon$ then gives the bound for every function in that ball; taking the supremum proves Eq. (34). The strict condition $r_* < q_x(s)$ supplies the contradiction. If $q_x(s) = 0$, there is no nonnegative $r < q_x(s)$ and the stated implication has no instance to check. □

C ANALYSES ON THE LOWER BOUND OF QUERY COST

The method guarantees give sufficient conditions for savings. This appendix develops the lower bounds summarized in Section 3.4: how many surrogate labels any method needs, how much they must cost, and how query choice affects that cost. We distinguish obtaining enough information from finding the useful labels. A certificate is a set of queried labels that leaves all surviving functions with a common accurate prediction at x. We first derive a cost lower bound, then evaluate it on an example in which a one-label certificate is costly to find.

C.1 THE COST OF ESTIMATING A LABEL

To bound the query cost of any method, we account for known labels and the prices of unqueried candidates. Fix a point x, a finite nonempty version space $V = \mathcal{V}(\mathcal{A})$ of functions consistent with the cached labels $\mathcal{A}$, and a finite set $\mathcal{C}$ of eligible uncached surrogates. A terminating policy π adaptively queries exact labels generated by an unknown $f \in V$ and returns $\widehat{y}_\pi$. Let N_π count its new surrogate labels and S_π their total price. The policy knows V, $\mathcal{C}$, and the prices, but not the true response function. Probability $\Pr_f$ and expectation $\mathbb{E}_f$ refer to its internal randomness, with the pool and response function fixed.

Definition 16. *For classification, let $L_x(a,f) = \mathbf{1}\{a \ne f(x)\}$ and set $\varepsilon = 0$, where $\mathbf{1}$ is the event indicator. For scalar responses, let $L_x(a,f) = |a - f(x)|$ and choose $\varepsilon \ge 0$. A surrogate-only policy is (ε,δ)-correct if*

$$\sup_{f \in V} \Pr_f\{L_x(\widehat{y}_\pi, f) > \varepsilon\} \le \delta.$$

For $0 \le \delta < 1/2$, define the optimal hard budgets, which limit cost or label count on each run, by

$$K_x(\varepsilon,\delta) = \inf\{b \ge 0 : \exists\ (\varepsilon,\delta)\text{-correct } \pi,\ S_\pi \le b \text{ almost surely for every } f \in V\}, \tag{40}$$

$$\Lambda_x(\varepsilon,\delta) = \inf\{n \in \mathbb{N}_0 : \exists\ (\varepsilon,\delta)\text{-correct } \pi,\ N_\pi \le n \text{ almost surely for every } f \in V\}. \tag{41}$$

Here $\mathbb{N}_0$ denotes the nonnegative integers, and an infimum over no feasible policy is $+\infty$. Dependence on $(V, \mathcal{C}, c)$ is implicit; unit prices make the two quantities equal.

The same policy must work for every possible truth. The quantity Λ_x asks how many surrogate labels suffice, while K_x asks how much they must cost. The selected point itself is excluded from $\mathcal{C}$, since one direct query would otherwise settle label complexity. These quantities measure recovery of $f(x)$ under realizability. They complement the population-risk guarantee of active local learning (Backurs et al., 2020), retained by the shared predictor in Theorem 1.

Information needed to distinguish predictions at x. Two hypotheses need to be distinguished only when they require incompatible answers at x. For finite V, write

$$E_x(V) = \begin{cases} \{\{f,g\} \subset V : f(x) \neq g(x)\}, & \text{classification,} \\ \{\{f,g\} \subset V : |f(x) - g(x)| > 2\varepsilon\}, & \text{scalar responses.} \end{cases} \tag{42}$$

The factor two in the scalar case comes from the accuracy intervals: no prediction can be within ε of both values when they are farther apart than 2ε. For a fixed f, we must distinguish it from each function that requires an incompatible prediction at x. Write this set as $G_x(f) = \{g \in V : \{f,g\} \in E_x(V)\}$, and let $D(f,g) = \{z \in \mathcal{C} : f(z) \neq g(z)\}$ be the candidates that distinguish such a function g from f. Define

$$\Psi_x(f) = \min_{w_z \geq 0} \left\{ \sum_{z \in \mathcal{C}} c(z) w_z : \sum_{z \in D(f,g)} w_z \geq 1 \quad \text{for every } g \in G_x(f) \right\}. \tag{43}$$

A sufficient query set must contain a candidate that distinguishes f from each incompatible hypothesis. This optimization relaxes that requirement: instead of selecting or omitting a candidate, it assigns a nonnegative weight to each one. For every such function, the weights on distinguishing candidates must sum to at least one. We call the minimum weighted price the fractional certificate cost. The value is zero when there are no incompatible hypotheses and infinite when the constraints are infeasible. The relaxation adapts the fractional certificate program of Gavinsky et al. (2017, Definition 6) by weighting queries by their prices and comparing predictions at x. It lower-bounds the spending of any adaptive query policy. When direct purchase is allowed, write C_π for the total cost of querying surrogates and, if needed, x.

A fractional certificate is defined for each possible true response function. The learner must choose queries without knowing which function is true. To account for this search, we assign weights to incompatible pairs and limit how much pair weight a query can separate for every possible set of surviving hypotheses.

Let $\mathcal{T}$ be the allowed queries: either $\mathcal{C}$ or $\mathcal{C} \cup \{x\}$, with $c(x) = d_x$. Choose hypothesis weights $\nu_f \geq 0$ summing to one and pair weights $w_{fg} \geq 0$ for $\{f,g\} \in E_x(V)$. For $W \subseteq V$, put $\nu(W) = \sum_{f \in W} \nu_f$. Define the pair-separation bound

$$\mathsf{LB}_x(\mathcal{T}) = \sup_{\nu, w} \left\{ \sum_{\{f,g\} \in E_x(V)} w_{fg} : \sum_{\substack{\{f,g\} \in E_x(W) \\ f(z) \neq g(z)}} w_{fg} \leq c(z)\nu(W) \quad (W \subseteq V,\ z \in \mathcal{T}) \right\}. \tag{44}$$

The weights ν_f let us average a policy's cost over possible truths. At a query whose surviving class is W, only truths in W incur its price. The constraint therefore limits the pair weight that this query can separate to its price multiplied by $\nu(W)$. Requiring the constraint for every W makes the bound valid regardless of how the policy adapts. The learner need not know these weights; they are chosen only to prove the lower bound. The optimization is a finite linear program with a constraint for every subset and allowed query.

Theorem 17. *Let V and $\mathcal{C}$ be finite, with exact responses, positive prices, and $0 \leq \delta < 1/2$. Every uniformly (ε, δ)-correct adaptive randomized policy using queries in $\mathcal{T}$ satisfies*

$$\sup_{f \in V} \mathbb{E}_f C_\pi \geq (1 - 2\delta)\mathsf{LB}_x(\mathcal{T}). \tag{45}$$

Here $C_\pi = S_\pi$ if direct purchase is excluded. Moreover,

$$\mathsf{LB}_x(\mathcal{C}) \geq \max_{f \in V} \Psi_x(f), \qquad d_x \geq \mathsf{LB}_x(\mathcal{C} \cup \{x\}) \geq \max_{f \in V} \min\{\Psi_x(f), d_x\}. \tag{46}$$

Under the same allowed queries and prices, the pair-separation bound is at least as large as the largest fractional certificate bound over all possible truths. In the family below, it is $(n+1)/2$, while every fractional certificate costs one. The proof follows the approach of tracking how queries distinguish weighted pairs (Aaronson, 2004, Section 4). Here only pairs requiring incompatible predictions at x receive weight. Query prices enter the constraints, the initial cache determines the surviving class, and the allowed query set specifies whether direct purchase is available. Brânzei et al. (2023,

Theorem 3) also bound the weight a query can separate for each set of remaining hypotheses. Here we allow unequal query prices and consider pairs that require different estimates at the selected point. This lower bound does not show that CALL-DIS achieves the minimum possible query cost. Section C.2.2 gives the proof.

The fractional certificate bound concerns a sufficient set of labels for a fixed true function; the pair-separation bound also accounts for the queries needed to find such labels. The following family shows how large this difference can be. There are $2n$ unit-cost candidates $z_1, \dots, z_{2n}$ and $2n$ possible truths:

$$f_i(z_j) = \mathbf{1}\{i = j\}, \qquad f_i(x) = \mathbf{1}\{i > n\}, \qquad i, j = 1, \dots, 2n.$$

Exactly one surrogate has response one; its index determines the label at x. Under f_i, querying z_i determines $f_i(x)$ with one label. Initially the learner does not know that index.

The pair-separation bounds are $\mathsf{LB}_x(\mathcal{C}) = (n+1)/2$ and $\mathsf{LB}_x(\mathcal{C} \cup \{x\}) = \min\{d_x, (n+1)/2\}$. Section C.2.3 evaluates these bounds.

This family shows that some search cost is unavoidable, even when a short certificate exists.

C.2 PROOFS FOR QUERY COMPLEXITY AND LABEL INFERENCE

This appendix proves the results in Section C: the general cost lower bound and its search example. Fix the selected point x, its direct-query price $d_x = c(x)$, a finite version space V consistent with the initial cache, and the remaining candidates $\mathcal{C}$. All queried labels are exact and can be reused without another query. A policy is a rule for choosing queries from the labels observed so far; all policies considered here terminate. Recall that $E_x(W)$ contains pairs in a surviving class W that predict different class labels at x, or scalar values more than 2ε apart (Eq. (42)). A certified answer must therefore rule out at least one member of each such pair.

C.2.1 COUPLING USED BY THE GENERAL LOWER BOUND

Fix $f \in V$ and $g \in G_x(f)$. Couple their runs using the same random seed. Until the first query in $D(f, g)$, all responses and decisions coincide. Therefore the event that such a query is ever made is the same event in the two coupled runs: the first distinguishing input is chosen from identical preceding histories.

On the event of no distinguishing query, both runs stop with the same output. The incompatible labels at x force at least one of these identical outputs to be inaccurate. Consequently,

$$\Pr_f\{\text{no query in } D(f,g)\} \le \Pr_f\{L_x(\widehat{y}_\pi, f) > \varepsilon\} + \Pr_g\{L_x(\widehat{y}_\pi, g) > \varepsilon\} \le 2\delta. \tag{47}$$

This step uses correctness under both truths, not an average-case assumption on a prior.

C.2.2 A PAIR-SEPARATION LOWER BOUND FOR ADAPTIVE SEARCH

The certificate bound fixes a true response function before choosing a sufficient set. We now prove that Eq. (44) also accounts for finding useful labels while that function is unknown. The proof tracks the first query that separates each incompatible pair, following the weighted progress argument of Aaronson (2004, Section 4). We adapt the accounting to unequal query costs and pairs requiring different predictions at x.

Proof of Theorem 17. Fix any feasible (ν, w) in Eq. (44). All weights are chosen before the policy's random seed. Zero hypothesis weights are allowed.

Responses are exact and the allowed input set is finite. A repeated query can be simulated from the preceding response without a purchase, so it can be removed. After a direct query the policy can immediately return the label at x. These changes preserve correctness and do not increase cost. Since the class is finite and the policy terminates almost surely under each truth, outside one null set of seeds it terminates under all truths. Fix such a seed. The remaining policy is a finite deterministic tree. At a query node v, let W_v be its nonempty surviving class and z_v its query. Exactly the truths in W_v reach this node.

A pair in $E_x(W_v)$ is separated at this node precisely when its members give different responses to z_v. Before that node their histories coincide. After it their paths differ, so this pair cannot be separated at any other node. By the constraint for (W_v, z_v),

$$\sum_{\substack{\{f,g\}\in E_x(W_v)\\ f(z_v)\neq g(z_v)}} w_{fg} \le c(z_v)\nu(W_v).$$

Sum over nodes. The left side counts every pair whose two runs ever separate exactly once. On the right side, expand $\nu(W_v)$ and interchange finite sums:

$$\sum_{\{f,g\}\in E_x(V)} w_{fg}\mathbf{1}\{\text{the two runs separate}\} \le \sum_v c(z_v)\nu(W_v) = \sum_{f\in V}\nu_f C_\pi(f). \tag{48}$$

Here $C_\pi(f)$ is the cost of the fixed-seed run under f. The identity charges a node only to truths that actually reach it.

For each incompatible pair, couple its two runs with the same seed. If they never separate, their outputs are equal and at least one is inaccurate. Uniform correctness therefore gives

$$\Pr\{\text{the two runs separate}\} \ge 1 - 2\delta,$$

by the argument in Eq. (47). Taking expectation in Eq. (48), using nonnegative weights, yields

$$(1-2\delta)\sum_{\{f,g\}\in E_x(V)} w_{fg} \le \sum_f \nu_f \mathbb{E}_f C_\pi \le \sup_f \mathbb{E}_f C_\pi.$$

The same inequality holds for the original policy because deleting repeated purchases only lowers its cost. Take the supremum over feasible weights to prove Eq. (45). If the supremum is infinite, the inequality means that no uniformly correct finite-cost policy exists.

Fix $f_0 \in V$. The dual of its fractional certificate program is

$$\sup_{a_g\ge 0}\left\{\sum_{g\in G_x(f_0)} a_g : \sum_{\substack{g\in G_x(f_0)\\ f_0(z)\neq g(z)}} a_g \le c(z) \quad (z\in\mathcal{C})\right\}. \tag{49}$$

If the certificate program is feasible, finite linear-programming duality equates its optimum $\Psi_x(f_0)$ with the displayed dual optimum (Boyd & Vandenberghe, 2004). Otherwise, some alternative hypothesis has no distinguishing input. Its dual weight then appears in no constraint and can grow without bound, matching the infinite certificate cost. If $G_x(f_0)$ is empty, both optima are zero.

Set $\nu_{f_0} = 1$, all other hypothesis weights to zero, $w_{f_0 g} = a_g$, and all other pair weights to zero. If $f_0 \notin W$, both the separated pair weight and $\nu(W)$ are zero. If $f_0 \in W$, the separated pair weight is bounded by the full sum in the corresponding constraint of Eq. (49), while $\nu(W) = 1$. These weights are therefore feasible in Eq. (44). This proves $\mathsf{LB}_x(\mathcal{C}) \ge \Psi_x(f_0)$.

If direct querying is allowed, the dual gains the constraint $\sum_g a_g \le d_x$. Its optimum is $\min\{\Psi_x(f_0), d_x\}$. To see attainment when the surrogate optimum is finite, keep an optimal dual solution if it satisfies the new constraint; otherwise scale all its weights until their sum is d_x. Scaling down preserves every surrogate constraint. When the surrogate optimum is infinite, give weight d_x to an alternative hypothesis that no surrogate distinguishes from f_0. The same construction proves the second lower inequality in Eq. (46). Maximizing over f_0 gives both claims. Finally the constraint $W = V, z = x$ in Eq. (44) reads $\sum_{\{f,g\}\in E_x(V)} w_{fg} \le d_x$, since every incompatible pair differs at x and $\nu(V) = 1$. This proves its upper inequality. □

Computing the bound. For $N = |V|$, the program has $N + |E_x(V)|$ variables and at most $2^N|\mathcal{T}|$ separation constraints, in addition to normalization and nonnegativity. It can be solved directly for small classes. A feasible choice of weights already proves a lower bound, so an analytic construction can avoid solving the program. Removing constraints has the opposite effect: it can increase the optimum and does not in general give a valid lower bound. The example with one surrogate labeled 1 below admits an exact analytic solution.

C.2.3 EVALUATING THE GENERAL BOUND ON THE SEARCH EXAMPLE

Let $I_0 = \{1, \dots, n\}$ and $I_1 = \{n+1, \dots, 2n\}$ be the two groups of possible marked indices. Under f_i, querying z_i reveals one and identifies the truth. Every other query reveals zero and rules out its own index. Thus an all-zero history is unresolved precisely when an index from each group remains.

The incompatible pairs form a complete bipartite graph between I_0 and I_1. Permutations within either group, and exchanging the two groups, preserve the program in Eq. (44). Average any feasible weights over these permutations. The constraints are linear, so their average remains feasible with the same objective. Every hypothesis then has weight $1/(2n)$, and every incompatible pair has a common weight $w \geq 0$.

Suppose a subset W contains the queried index and b indices from the opposite group. That query separates exactly b pairs. If $a \geq 1$ indices of its own group belong to W, its constraint is

$$bw \leq \frac{a+b}{2n}.$$

For $b \geq 1$ the smallest right-to-left ratio occurs at $a = 1, b = n$, yielding $w \leq (n+1)/(2n^2)$. This choice meets every surrogate constraint; subsets without the queried index have zero separated weight. Since there are n^2 pairs, the surrogate-only optimum is $(n+1)/2$.

For the direct query, a subset with group sizes a, b gives $abw \leq d_x(a+b)/(2n)$. For $a, b > 0$, the ratio $ab/(a+b)$ increases with either argument and is at most $n/2$, with equality at $a = b = n$. Thus all direct constraints are equivalent to $n^2 w \leq d_x$. The combined optimum is $\min\{d_x, (n+1)/2\}$, as claimed. This calculation also covers $n = 1$.

D ADDITIONAL EXPERIMENTAL RESULTS

We provide the detailed settings, then analyze prediction accuracy, labeling cost, and the contribution of estimated labels. Tables report the full comparisons; the source archive provides implementation settings and scripts.

D.1 EXPERIMENTAL SETTINGS

D.1.1 DATASETS, PRICES, AND PREDICTION MODELS

The reported tasks are 13 BindingDB pAffinity regressions and five binary molecular property tasks, listed in Table 4. The BindingDB pools were selected using molecular structures and assigned costs to include expensive compounds with several cheap, structurally similar candidates; affinity labels were not used for ranking. These results describe the selected pools and cost settings. BindingDB uses deterministic 80/20 splits (Liu et al., 2025); classification uses stratified 80/20 splits with split seed 42. Bioavailability measures absorption; the other four tasks measure toxicity. We do not rebalance the pools by cost.

Molecular acquisition costs. We use the effort of obtaining a compound for an assay as a proxy for label cost. The Ertl–Schuffenhauer synthetic-accessibility score $\mathrm{SA}(x)$ is computed from structure before the response is known; lower scores indicate easier synthesis (Ertl & Schuffenhauer, 2009). The reported experiments map it to fixed, normalized costs:

$$c(x) = \begin{cases} 1, & \mathrm{SA}(x) \leq 3.5, \\ 4, & 3.5 < \mathrm{SA}(x) \leq 6, \\ 20, & 6 < \mathrm{SA}(x) \leq 8, \\ 100, & \mathrm{SA}(x) > 8. \end{cases} \tag{50}$$

The boundary at six follows the SAVI Space interpretation of higher scores as difficult synthesis (Korn et al., 2025). SynFrag uses SA scores in $[3.5, 6]$ for molecules requiring further retrosynthetic assessment (Zhang et al., 2026). We split the upper range at eight to distinguish high from extreme difficulty. The wrapper's eligible cost range $c(x) \leq 4$ corresponds to $\mathrm{SA}(x) \leq 6$.

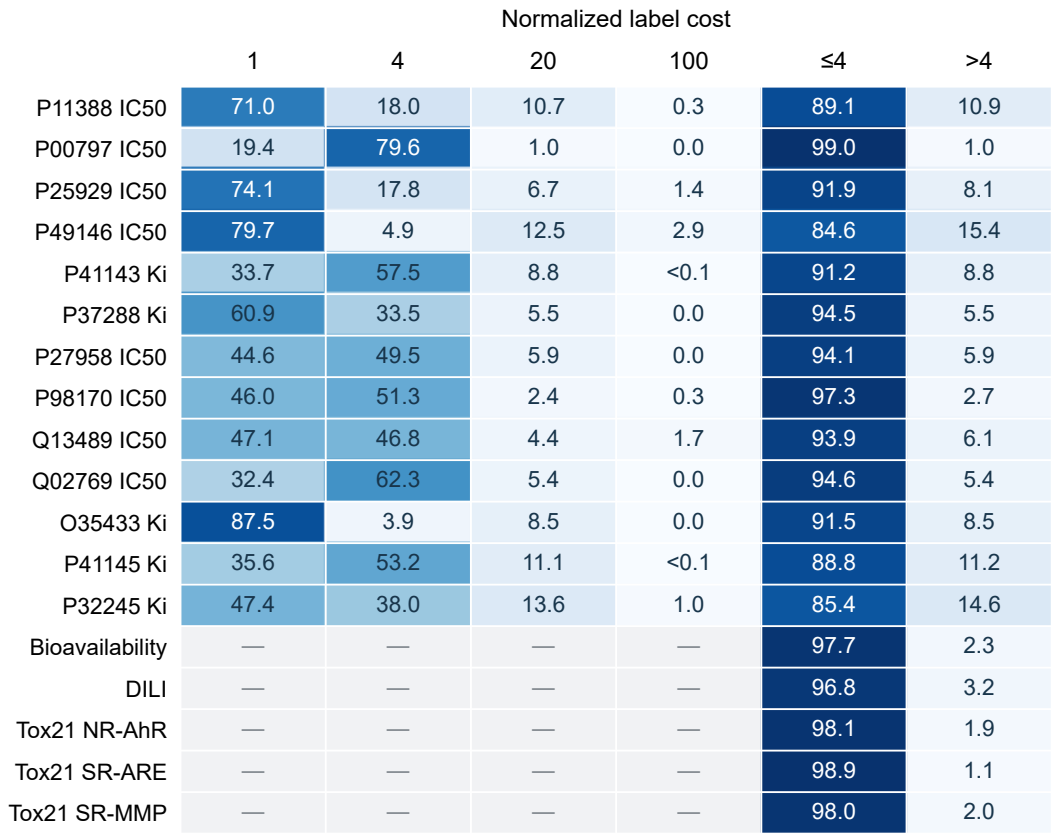


Figure 3: Cost composition of the 18 datasets. Cells give percentages; darker blue denotes a larger fraction. Classification summaries distinguish only costs at most four versus above four.

Empirical cost schedule. The normalized costs $\{1, 4, 20, 100\}$ increase by factors of four, five, and five. They span two orders of magnitude, allowing substantial savings from cheaper surrogate queries. For context, MolPrice reports prices for approximately 5.5 million catalog compounds whose natural logs span 0–12 when prices are measured in USD/mmol (Hastedt et al., 2025).

Public procurement examples illustrate the tier of 20. The University of Tennessee Health Science Center lists a five-step synthesis of 20–50 mg at approximately $2,985, excluding starting materials and special reagents, with 35% and 50% surcharges for external academic and commercial users (University of Tennessee Health Science Center, 2026). TargetMol lists XCT790 at $189 for 25 mg and $345 for 50 mg (TargetMol, 2026). These unmatched examples give price ratios of roughly 9–16 before surcharges and 12–24 with them. They motivate 20 as an illustrative custom-synthesis cost. The tiers are assumed benchmark costs, not measured procurement prices, which vary widely and are only partly explained by SA (Hastedt et al., 2025; Sanchez-Garcia et al., 2023).

Prediction models. XGBoost and the three-layer neural network (NN3) use radius-2, 1,024-bit Morgan fingerprints (Rogers & Hahn, 2010). The auxiliary committee uses radius-4, 1,024-bin Morgan count features with chirality. The source archive gives model parameters and data-processing code for the reported tasks.

D.1.2 Acquisition rules and paired runs

Each selector listed in Section 4.1 is compared with and without the practical CALL-Dis wrapper. Paired runs share the split, initial labels, prices, and prediction-model settings; later training sets and selected targets may diverge. Ten seeds change initialization and algorithm randomness on the same split. The base prediction model remains fixed within each training cycle; the selector is

called on the updated pool at each action. The auxiliary committee is refitted for each target attempt. BatchBALD, BADGE, and Fast-BAIT use sequential one-point adaptations. Each exact response is charged once, including unsuccessful exploration and fallback. The source archive specifies the selector implementations, seed list, and stopping rules.

D.1.3 PRACTICAL DISAGREEMENT WRAPPER

Committee choice. ExtraTrees randomizes candidate features and split thresholds to form efficient classification and regression ensembles (Geurts et al., 2006). Each tree predicts a complete response, allowing queried labels to eliminate inconsistent trees. Related work uses random-forest committees for active learning (Borisov et al., 2011) and prediction variance for molecular pAffinity selection (Reker et al., 2016); here we use ExtraTrees.

Implementation. The implementation reserves the target's purchase cost within each cycle's budget of 100 and requires at least one new cheap query before accepting an estimate. Thus, accepted estimates can replace targets costing 20, whereas targets costing 100 are purchased directly when processed. The tier of 100 represents expensive direct queries in this evaluation. For a target costing more than four, a 64-tree ExtraTrees committee (Geurts et al., 2006) is fitted to paid labels and fixed during that attempt. The wrapper queries uniformly from affordable, unqueried disagreement points costing at most four, then removes trees inconsistent with the response. It accepts an estimate after at least one fresh cheap query if the maximum pairwise disagreement probability on cheap points is at most 0.05 and the survivors agree at the target. For regression, consistency uses $\gamma = 0.5$, target prediction diameter is at most 2ε with $\varepsilon = 0.5$, and the estimate is the midpoint. Otherwise it queries the target directly. Exploration reserves enough of the cycle budget for that fallback. Estimated labels enter predictor training only; committee fitting, hypothesis filtering, and the cache use purchased labels exclusively.

Relation to XGBoost. The auxiliary committee estimates labels; XGBoost or NN3 selects targets and predicts test responses. Individual boosted XGBoost trees are additive components of one predictor (Chen & Guestrin, 2016), so they cannot replace complete ExtraTrees predictions in the agreement test. Separately trained XGBoost models could form a committee, but their estimation accuracy, acceptance rate, and computation require a matched comparison. All reported runs use ExtraTrees.

Interpretation of agreement. The fitted committee may exclude the true response function, and its candidate selection differs from the independent draws in Theorem 3. That theorem therefore does not guarantee the molecular wrapper's accuracy or expected savings. The empirical ε is an acceptance tolerance, not a calibrated error bound. Sections D.5.1 and D.5.2 assess accepted-label errors directly.

D.1.4 SCORES AND UNCERTAINTY

Classification uses AUROC and BindingDB uses negative RMSE; higher is better. For each paired run, scores are linearly interpolated only on the common observed paid-cost interval of the direct and CALL runs $[a, b]$. The cost-normalized area and paired difference are

$$\mathrm{AULC}_j = \frac{1}{b-a}\int_a^b s_j(c)\,dc, \qquad \Delta\mathrm{AULC} = \mathrm{AULC}_{\mathrm{CALL}} - \mathrm{AULC}_{\mathrm{direct}}. \tag{51}$$

Positive values favor CALL-DIS. For the figures, each dataset uses the smallest additional-cost endpoint b_* shared by all displayed selectors, direct and CALL runs, and seeds. The cumulative paired effect is

$$\Delta\mathrm{AUC}_i(c) = \int_0^c [s_{\mathrm{CALL},i}(u) - s_{\mathrm{direct},i}(u)]\,du, \qquad 0 \le c \le b_*. \tag{52}$$

Lines average ten paired seeds; bands show one sample standard deviation. There is no smoothing or extrapolation. A positive cumulative difference need not indicate better performance at every budget; the integrated figure values also depend on the cost range. Classification effects are in AUROC percentage points and regression effects remain in pAffinity units. The paired t-test settings for

Table 2: Prediction error and query cost of CALL-LIP with the experimental sample sizes. Entries are mean ± SD; lower is better. Bold marks the largest error and costs below direct labeling.

(a) Average error ↓

Noise SD	n	Excess loss	Maximum
0.00	2,230	0.0177 ± 0.0052	0.0236
0.00	8,920	0.0165 ± 0.0052	0.0230
0.05	2,230	0.0160 ± 0.0054	**0.0243**
0.05	8,920	0.0134 ± 0.0055	0.0217

(b) Total query cost ↓

		CALL-LIP	
Price	Direct	$n = 2,230$	$n = 8,920$
20	2,000	$2,000 \pm 0$	$2,000 \pm 0$
2,000	200,000	$\mathbf{708.0} \pm 6.3$	$\mathbf{3,438.4} \pm 967.2$
10,000	1,000,000	$\mathbf{708.0} \pm 6.3$	$\mathbf{2,838.4} \pm 5.1$

performance at fixed labeling costs are given in Section D.4. Accepted-label accuracy compares estimates with recorded responses only after the run; repeated targets across runs count as separate events.

D.2 CONTROLLED REGRESSION CHECKS OF CALL-LIP

We compare CALL-LIP with direct purchase of the same 100 requested targets. Inputs are uniform on $[0, 1]$; the cheap pool has 2,230 or 8,920 points, and target prices are 20, 2,000, or 10,000. Responses follow $(1 + \sin(4\pi x))/2$ with zero or 0.05 Gaussian noise, clipped to $[0, 1]$. The predictor uses $L = 2\pi$ and $\eta = 0.25$. The two sample sizes retain at most 355 and 1,420 records per fitting interval; each record costs one. These samples are smaller than required by Eq. (10). We compute excess loss by subtracting the loss of the known conditional median on the same evaluation data. Both methods receive the same selected points, so the comparison measures the cost of obtaining their labels without changing the acquisition rule.

On 100,000 independent evaluation pairs per seed, all 40 excess mean absolute losses are below 0.25, with a maximum of 0.0243. Reordering requests leaves estimates unchanged. At target price 20, the method queries directly; at 2,000 and 10,000, it saves 98.3–99.9% of labeling cost. These experiments check prediction error and query cost with the chosen sample sizes. They do not test how the required number of labels changes with η or guarantee accuracy at each selected point.

D.3 FURTHER ANALYSIS OF MOLECULAR PERFORMANCE

NN3 Uncertainty gains most on all 13 BindingDB pools, partly reflecting a weak direct-query baseline. On P49146, its AULC improves from -2.129 to -1.175 with CALL-DIS, but remains below direct Random (-0.839). At the first checkpoint with at least half the pool labeled, direct Uncertainty spends 1,724 additional cost units, compared with 1,038 for CALL-DIS and 640 for direct Random. For CALL-DIS, the number of labeled training points includes both points with estimated labels and surrogate points whose labels were purchased. Thus, the wrapper obtains more training labels per cost, but this comparison does not isolate which change causes the gain. Classification depends more on the selector: on SR-ARE, NN3 ALFA-Mix gains 3.14 AUROC points while Random loses 0.71.

For Unc/Cost, 27 of 36 mean differences are numerically zero and eight are positive; large gains over acquisition rules that ignore price need not extend to rules that already account for cost.

D.4 SIGNIFICANCE TESTS AT THE SAME LABELING COST

For each dataset, prediction model, and acquisition rule, we compare the rule with and without CALL-DIS at labeling costs of 500 and 1,000 beyond initialization. Each pair shares the initial labels and their cost. We use AUROC for classification and negative RMSE for regression, so a positive difference favors CALL-DIS. If a budget falls between recorded checkpoints, we linearly interpolate the score without extrapolation. We include a dataset at a budget only when every method and seed reaches that cost. All 18 datasets qualify at 500; 14 qualify at 1,000.

At each budget, we pair the scores from the same seed and apply a two-sided paired t-test to the ten differences. The null hypothesis is that their mean is zero. With $p < 0.05$, we count a win if the mean difference is positive and a loss if it is negative; otherwise, we count a tie. A tie means that the test does not detect a difference. The seeds vary initialization and algorithm randomness on the same

Table 3: Paired t-test wins/ties/losses across acquisition rules at the same labeling cost ($p < 0.05$). Costs exclude initialization. A dash indicates that some runs stop before that cost.

	Cost 500		Cost 1,000	
Dataset	XGBoost	NN3	XGBoost	NN3
P11388 IC50	2/6/0	3/6/0	—	—
P00797 IC50	0/8/0	1/8/0	0/8/0	1/8/0
P25929 IC50	3/5/0	5/4/0	4/4/0	5/4/0
P49146 IC50	3/5/0	2/7/0	3/4/1	2/7/0
P41143 Ki	2/6/0	2/7/0	2/6/0	2/7/0
P37288 Ki	3/5/0	2/7/0	3/5/0	3/6/0
Bioavailability	1/9/0	0/16/0	—	—
DILI	0/10/0	0/16/0	—	—
Tox21 NR-AhR	1/9/0	0/16/0	1/9/0	4/12/0
P27958 IC50	4/4/0	4/5/0	3/5/0	2/7/0
P98170 IC50	3/4/1	1/8/0	3/5/0	2/7/0
Q13489 IC50	1/7/0	1/8/0	2/6/0	1/8/0
Q02769 IC50	3/5/0	2/7/0	3/5/0	1/8/0
O35433 Ki	2/6/0	1/8/0	—	—
P41145 Ki	2/6/0	2/7/0	2/6/0	1/8/0
P32245 Ki	2/6/0	1/8/0	2/5/1	2/7/0
Tox21 SR-ARE	0/10/0	1/15/0	1/9/0	9/7/0
Tox21 SR-MMP	0/10/0	0/16/0	2/8/0	3/13/0
Total	32/121/1	28/169/0	31/85/2	38/109/0

data split. Each test concerns one method pair at one budget; the counts summarize these individual comparisons.

Table 3 reports 60 wins, 290 ties, and one loss at cost 500, and 69 wins, 194 ties, and two losses at cost 1,000. Most comparisons do not show a significant difference, although detected gains outnumber declines at both budgets. The two budgets cover different numbers of datasets, so their counts should be read separately.

D.5 Accepted estimates and training effects

We audit accepted estimates on all 18 datasets and replay training on P00797 IC50 and O35433 Ki with purchases held fixed. Recorded target responses are used only after each run to assess accuracy. Table 4 pools accepted events across selectors and seeds, counting repeated targets in different runs separately.

D.5.1 Agreement and accepted-label accuracy

Of 58,636 accepted estimates, 55,249 (94.2%) have one surviving tree, for which agreement is automatic. Only 16,469 of 40,238 regression estimates fall within 0.5 pAffinity units of the recorded response; 12,360 of 18,398 classification estimates match the recorded label. Agreement therefore provides limited evidence of accuracy in this fitted committee.

At a tolerance of 1.0 pAffinity unit, regression error ranges from 6.3% on P11388 IC50 to 69.5% on P00797 IC50, where 444 of 639 estimates exceed the tolerance. Classification error ranges from 23.3% on NR-AhR to 47.2% on Bioavailability. The learning gains in the main comparison therefore coexist with substantial, dataset-dependent estimation errors.

D.5.2 Error tolerance and number of surviving trees

Regression error means $|\widehat{y} - y| > t$ for recorded pAffinity y. Since pAffinity is $-\log_{10}$ affinity in molar units, $t = 0.5$ permits a factor of $10^{0.5} \approx 3.16$ in affinity. This equals the implementation's target tolerance $\varepsilon = 0.5$, not a measured assay-noise bound. We vary t for the same 40,238 estimates, keeping predictions, purchases, ε, and γ fixed.

Table 4: Pool sizes and accepted-label errors, pooled across runs. Regression errors exceed 1.0 pAffinity unit; classification errors are label mismatches.

Dataset	Pool	Test	Accepted	Errors	Error (%)
BindingDB					
P11388 IC50	270	68	1,928	121	6.3
P00797 IC50	2,358	590	639	444	69.5
P25929 IC50	396	99	2,034	738	36.3
P49146 IC50	326	82	2,547	961	37.7
P41143 Ki	2,757	690	5,679	2,682	47.2
P37288 Ki	508	127	2,110	938	44.5
P27958 IC50	700	175	3,341	560	16.8
P98170 IC50	735	184	1,443	391	27.1
Q13489 IC50	275	69	360	93	25.8
Q02769 IC50	447	112	1,380	538	39.0
O35433 Ki	365	92	2,239	868	38.8
P41145 Ki	2,772	694	7,572	3,354	44.3
P32245 Ki	1,506	377	8,966	3,197	35.7
Classification					
Bioavailability	512	128	2,053	969	47.2
DILI	380	95	2,042	887	43.4
Tox21 NR-AhR	5,239	1,310	4,803	1,117	23.3
Tox21 SR-ARE	4,665	1,167	4,141	1,333	32.2
Tox21 SR-MMP	4,648	1,162	5,359	1,732	32.3

In Table 5a, error falls from 75.6% at $t = 0.25$ to 59.1% at $t = 0.5$ and 37.0% at $t = 1.0$. Mean absolute error is 1.009 pAffinity units; the median is 0.677. Loosening the audit criterion changes reported error, not prediction quality, and substantial error remains even at a tenfold affinity tolerance.

More surviving trees are associated with lower regression error. Spearman correlations with the error indicator and absolute error are -0.155 and -0.164. In all ten datasets containing both estimates based on one surviving tree and those based on multiple surviving trees, the latter have lower error. The mean within-run difference is -19.6 percentage points across 1,044 runs containing both groups, weighted equally. However, correlating ranks after ranking and centering within each dataset-model-selector-seed run gives only -0.030 for the error indicator. These associations are descriptive because accepted events share targets and training data.

We then retain only estimates with at least m surviving trees and measure their error and the fraction of estimates retained (Table 5b). For regression, $m = 2$ lowers pooled error from 59.1% to 33.0% and mean absolute error from 1.009 to 0.516, retaining only 7.60% of estimates. At $m = 4$, error is 25.7% and retention 4.44%. No estimates remain on P00797, P41143, or P41145 for $m \geq 2$. Classification shows no consistent benefit: the correlation is $+0.015$, and $m = 2$ raises error from 32.8% to 38.3% while retaining 1.79%. Larger cutoffs leave too few classification events for a general conclusion. These retrospective filters motivate task-specific validation; their effect on online cost and learning remains untested.

D.5.3 Training with fixed purchases

On P00797 IC50 and O35433 Ki, we retrain the prediction model using the purchased labels from each recorded stage of active learning, either including or excluding the estimated labels. Both versions use the same purchased labels and are evaluated at the same labeling costs. This comparison tests whether adding estimated labels improves prediction. Separate online ablations train only on purchased labels, query cheap candidates without requiring disagreement, or omit purchased cached labels from initial hypothesis filtering. These changes can alter later acquisitions; none uses estimated labels to filter hypotheses.

The ablation study was run separately from the main comparison using the same seed values. The main-table NN3 Uncertainty gains are 0.0547 on P00797 and 0.9454 on O35433. The gains below come from the ablation runs, whose predictions differ from those in the main comparison. For NN3 Uncertainty on P00797, the full wrapper gains 0.0623 ± 0.0660 AULC over direct querying,

Table 5: Post-run label sanity checks. (a) Error at audit tolerance t for the same 40,238 regression estimates. (b) Estimates retained by a minimum survivor count m. Regression errors exceed 0.5 pAffinity units; classification errors are label mismatches. Retention is relative to originally accepted events in each task type.

(a) Regression tolerance

t (pAffinity)	Affinity factor 10^t	Errors / 40,238	Error (%)
0.1	1.26	35,355	87.9
0.25	1.78	30,417	75.6
0.5	3.16	23,769	59.1
0.75	5.62	18,859	46.9
1	10.00	14,885	37.0
1.5	31.62	9,645	24.0
2	100.00	6,077	15.1

(b) Minimum surviving trees

	Regression			Classification		
m	Retained	Retain (%)	Error (%)	Retained	Retain (%)	Error (%)
1	40,238	100.00	59.1	18,398	100.00	32.8
2	3,058	7.60	33.0	329	1.79	38.3
3	2,258	5.61	28.3	163	0.89	27.6
4	1,787	4.44	25.7	93	0.51	22.6
5	1,516	3.77	23.2	56	0.30	21.4
8	1,021	2.54	21.1	19	0.10	5.3
16	511	1.27	20.2	2	0.01	0.0
32	216	0.54	19.4	0	0.00	—

while adding estimates in replay changes AULC by -0.0069 ± 0.0284. On O35433, the respective changes are $+0.9318 \pm 1.0574$ and -0.0534 ± 0.0941 (mean $\pm$ one standard deviation across seeds). Online training using purchased labels alone also retains positive mean gains over direct querying: 0.0681 ± 0.0664 and 0.9974 ± 0.9720, respectively. These findings support a contribution from cheaper label acquisition, but do not show an advantage from adding estimated labels on these two tasks.

The control that queries cheap candidates uniformly without requiring disagreement retains the same acceptance criteria and budget rules; accepted estimates still enter predictor training. With NN3 Uncertainty, its mean ΔAULC over direct querying is 0.0644 ± 0.0558 on P00797 and 0.9049 ± 0.9306 on O35433, compared with 0.0623 ± 0.0660 and 0.9318 ± 1.0574 for the full procedure (mean $\pm$ one standard deviation over ten seeds). Requiring disagreement therefore gives no consistent mean improvement on these two tasks.

D.6 Finite-class search controls

We use known finite hypothesis classes containing the true function. Queries eliminate inconsistent functions until the target label is determined, or its scalar value is known within tolerance. No molecular predictor is fitted. Table 6 varies query choice, prices, and adaptation. Panels (a,b) report means and Monte Carlo standard errors over 1,000 trials; panel (c) averages worst-case cost over ten instances.

Finding informative candidates. In panel (a), only the first candidate determines the target label, and every query costs one. Querying that candidate costs one regardless of pool size, whereas uniform disagreement search rises from 1.506 with two candidates to 6.390 with twelve. Extra uninformative candidates increase the cost of finding the useful label without changing the cost of that label.

Prices and fallback. Panel (b) assigns prices $[1, 1, 2, 2, 3, 4, 4]$ to seven candidates; direct querying costs 250 and exploration is capped at 250. Making informative candidates expensive raises mean cost from 2.385 to 4.497 for threshold functions, but query costs vary little across the price assignments in the scalar-response experiments (3.273–3.352). All six means remain below direct purchase.

Table 6: Cost to obtain a selected target's label (↓). Panels (a,b): mean ± Monte Carlo SE over 1,000 trials; panel (c): mean worst-case cost over ten instances.

(a) Finding one useful query

Candidates	Uniform search	Best policy
2	1.506 ± 0.016	1
4	2.497 ± 0.035	1
8	4.502 ± 0.072	1
12	6.390 ± 0.110	1

(c) Choosing queries after earlier responses

Functions	Tolerance	Fixed set	Adaptive	Uniform
Multiclass	–	2.40	2.30	7.50
Scalar	.02	1.00	1.00	3.70
Scalar	.05	1.00	1.00	3.70
Scalar	.10	1.00	1.00	3.70
Threshold	–	2.40	2.40	8.30

(b) Useful-query prices and direct fallback

Functions	Price placement	Mean total cost
Candidate labels determine the target		
Threshold	Informative points cheap	2.385 ± 0.049
Threshold	Informative points costly	4.497 ± 0.056
Threshold	Random assignment	4.330 ± 0.064
Scalar	Informative points cheap	3.327 ± 0.069
Scalar	Informative points costly	3.352 ± 0.057
Scalar	Random assignment	3.273 ± 0.075
Candidate labels cannot determine the target		
Uninformative labels	Cheap-first assignment	251.773 ± 0.024
Uninformative labels	Costly-first assignment	253.024 ± 0.029
Uninformative labels	Random assignment	252.824 ± 0.036

Uniform: uniform disagreement search. The best policy in (a) always queries the first candidate. Fixed set in (c) chooses all queries before seeing responses; adaptive chooses later queries using earlier responses. Tolerance is the allowed scalar prediction error. In (b), direct labeling costs 250 and exploration is capped at 250. Uninformative candidate labels cannot determine the target.

When candidate labels cannot determine the target, fallback raises total cost to 251.773–253.024. Savings therefore require informative candidates as well as favorable prices; the cap only limits failed exploration.

Choosing queries from earlier responses. Panel (c) compares a fixed query set with adaptive query choice. Adaptation reduces average worst-case cost from 2.40 to 2.30 on multiclass tasks, versus 7.50 for uniform search. On threshold and scalar tasks, choosing queries in advance and choosing them from earlier responses achieve the same cost, both below uniform search. In these controls, selecting informative candidates accounts for most of the reduction; adaptation helps when responses change which query is useful next.

D.6.1 Improving label estimates

The finite-class controls assume the true function is retained, so agreement ensures accuracy. A fitted committee lacks this guarantee, particularly when filtering leaves one tree. The audit and replay results therefore motivate two improvements: reduce model error through alternative surrogate models or molecular representations, and validate acceptance rules on separate paid labels. A committee with less correlated errors may also make agreement more informative.

The analysis of the number of surviving trees suggests accepting an estimate only when enough models remain and their predictions agree within the required tolerance. The minimum number of models would be chosen separately for each task; otherwise, the learner would query the selected point directly. Correlated models can still agree incorrectly, and stricter acceptance can sacrifice savings. Future online comparisons should report estimation error and acceptance rate alongside prediction performance at matched labeling budgets, charging for validation labels and additional fallbacks. The present controls do not evaluate these alternatives.

Appendix References

Scott Aaronson. Lower bounds for local search by quantum arguments. *arXiv preprint arXiv:quant-ph/0307149*, 2004. URL https://arxiv.org/abs/quant-ph/0307149v2. Revised February 2004.

Arturs Backurs, Avrim Blum, and Neha Gupta. Active local learning. In *Proceedings of the 33rd Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 363–390. PMLR, 2020. URL https://proceedings.mlr.press/v125/backurs20a.html.

Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM*, 36(4):929–965, 1989. doi: 10.1145/76359.76371.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013. doi: 10.1093/acprof:oso/9780199535255.001.0001.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. doi: 10.1017/CBO9780511804441. URL https://web.stanford.edu/~boyd/cvxbook/.

Simina Brânzei, Davin Choo, and Nicholas Recker. The sharp power law of local search on expanders, 2023. URL https://arxiv.org/abs/2305.08269.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

Michael Dinitz, Sungjin Im, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Controlling tail risk in online ski-rental. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 4247–4263, 2024. doi: 10.1137/1.9781611977912.147. URL https://arxiv.org/abs/2308.05067.

Devdatt Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures & Algorithms*, 13(2):99–124, 1998. doi: 10.1002/(SICI)1098-2418(199809)13:2<99::AID-RSA1>3.0.CO;2-M. URL https://www.brics.dk/RS/96/25/BRICS-RS-96-25.pdf. Preprint version: BRICS RS-96-25, 1996.

Dmitry Gavinsky, Rahul Jain, Hartmut Klauck, Srijita Kundu, Troy Lee, Miklos Santha, Swagato Sanyal, and Jevgenijs Vihrovs. Quadratically tight relations for randomized query complexity. *arXiv preprint arXiv:1708.00822*, 2017. URL https://arxiv.org/abs/1708.00822v1.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019. URL https://www.cambridge.org/core/books/understanding-machine-learning/3059695661405D25673058E43C8BE2A6.

Matus Telgarsky. Deep learning theory lecture notes, 2021. URL https://mjt.cs.illinois.edu/dlt/. Version 2021-10-27, v0.0-e7150f2d.