Incentivized Collaboration in Active Learning

Lee Cohen[1] ✉
Stanford University, CA, USA

Han Shao ✉
Toyota Technological Institute of Chicago, IL, USA

──────── **Abstract** ────────

In collaborative active learning, where multiple agents try to learn labels from a common hypothesis, we introduce an innovative framework for incentivized collaboration. Here, rational agents aim to obtain labels for their data sets while keeping label complexity at a minimum. We focus on designing (strict) *individually rational* (IR) collaboration protocols, ensuring that agents cannot reduce their expected label complexity by acting individually. We first show that given any optimal active learning algorithm, the collaboration protocol that runs the algorithm as is over the entire data is already IR. However, computing the optimal algorithm is NP-hard. We therefore provide collaboration protocols that achieve (strict) IR and are comparable with the best known tractable approximation algorithm in terms of label complexity.

1 Introduction

Active learning has emerged as a powerful paradigm in which labels of selected data points are sequentially queried from a large pool of unlabeled data, referred to as the unlabeled pool. The primary objective is to minimize labeling effort to find a classifier that exhibits low error on fresh data points from the same data source, known as generalization error. Typically, if the pool is large enough, a classifier that performs well on the pool can also achieve low generalization error through uniform convergence.

Active learning has also been studied in the distributed setting, where the unlabeled pool is scattered across multiple machines (called agents), (e.g., [45, 2]). While active learning has demonstrated promising results, traditional approaches often operate in isolation, neglecting the potential benefits of collaboration among agents should they agree to collaborate. In this paper, we propose a novel framework for incentivized collaboration active learning, where agents can collaboratively explore their data pools to discover a common target function.

---

[1] This work was done when the author was at Toyota Technological Institute of Chicago.

The motivation for collaboration in active learning stems from real-life scenarios where collaboration and collective intelligence yield improved outcomes, e.g., when agents collect data from the same distribution, and can easily end up labeling the same or very similar points. This redundancy leads to unnecessary and inefficient utilization of resources, as the labeling is often done by experts. Additionally, more data can be translated to improved accuracy, prompting agents to pool their resources and employ a more powerful model.

The incentive-driven nature of our framework aligns with the reality of collaboration in the real world. When agents are incentivized to collaborate only when their expected labeling complexity decreases, it reflects the real-life scenario where individuals are motivated to engage in cooperative endeavors if they perceive a clear benefit, such as reduced effort, faster, and better outcomes. In this work, we focus on a specific notion of incentives, where agents already have access to a baseline algorithm and they are motivated to join the collaboration if their label complexity is smaller than running the baseline algorithm on their own.

Consider, for example, the case of a new drug (e.g., Paxlovid for Covid-19[40], that has different efficacy on patients with different features. While individual hospitals can test the drug on their patients in an active learning fashion by executing their preferred baseline algorithm, collaborating efficiently with other hospitals, each with their own patients, often leads to a better prognosis.

However, if the incentives of the hospitals are not maintained, i.e., the effort of some hospitals is increased, the collaboration may be compromised. By emulating this collaboration within the active learning framework, we unlock the potential of collective intelligence to enhance the learning process. Besides, imagine that several data labeling companies have to recover the labels of unlabeled images assigned to them. Each data labeling company would like to collaborate with other companies to recover the labels of all images while minimizing the query complexity and not increasing their burden.

Our basic model is as follows: there are k agents, each with their own set of unlabeled data points, and a single hypothesis class with a prior on the hypotheses, which all agents are aware of. We assume realizability, meaning that there exists an underlying ground truth labeling function called the target function, labeling all the data points, and that the hypothesis class encompasses such a target function. We refer readers to the discussion for more information about this assumption in the context of active learning.

The agents reach a consensus on an arbitrary baseline algorithm for pool-based active learning (e.g., the best tractable approximation algorithm). To select whether or not to join the collaboration, the agents need to evaluate their utility from joining the collaboration. Since the goal of each individual agent (regardless of the collaboration) is to minimize their expected query complexity, the most natural cost function is the expected query complexity. To ensure each individual benefits from their collaboration, we establish a collaboration protocol that guarantees that each agent cannot reduce their expected label complexity by running the baseline algorithm individually. This concept is referred to as individual rationality (IR). Our objective is to design an IR collaboration protocol that minimizes the overall labeling queries.

There are cases in which collaboration is not necessarily beneficial. For example, each agent has non-zero points on a different axis and the hypothesis class contains every possible halfspace. If the prior distribution is uniform over all labelings, then no agent can reduce their label complexity by joining the collaboration.

Clearly, if each agent has the same set of points, the label complexity of each agent can decrease to $1/k$ of its original label complexity if the collaboration protocol equally splits the labeling burden. Even if agents do not share the same set of points, they can still benefit from collaboration, as we show in the next example.
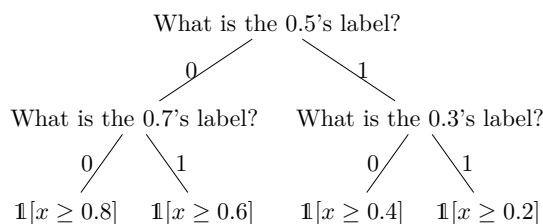
Figure 1 The query tree of binary search for thresholds.

▶ **Example 1.** Consider the scenario with 1-dimensional thresholds $\mathcal{H} = \{\mathbb{1}[x \geq \alpha] | \alpha = 0.2, 0.4, 0.6, 0.8\}$ and a uniform prior distribution over $\mathcal{H}$. Suppose agent 1 has points $\{0.25, 0.5, 0.75\}$ and agent 2 has points $\{0.3, 0.45, 0.55, 0.7\}$. When running binary search collaboratively, each agent only performs one labeling query, as illustrated as a search tree in Fig 1. On the other hand, if they were to run binary search independently, each agent would need to query 2 labels. Thus, collaboration can effectively reduce the label complexity for each agent by 1.

Bayesian Assumption. The reason why we have a Bayesian assumption regarding the hypothesis class is that without it, querying all the labels to discover the target hypothesis can be inevitable, even for a simple class of linear separators in $\mathcal{R}^2$ (see, e.g., Claim 1 in [21]). It is worth noting that as in [21], we *do not require the prior distribution to align with nature.* Instead, the prior distribution serves as a measure for average case analysis. Having a prior belief in our model has the following clear assumption. If the algorithm reaches a point where the remaining consistent hypotheses largely agree on the unlabeled data, it is reasonable to stop and output one of these remaining hypotheses [28]. In a non-Bayesian setting, it does not make sense to operate this way.

Game Theory Interpretation. The agreed-upon baseline algorithm induces a sort of (not private) values for agents- each agent has its (negative) individual labeling complexity as value. The collaboration protocol can be then interpreted as a mechanism: Initially, the collaboration protocol (principal) is introduced to the agents, and each agent can understand it and have confidence in the principal's commitment to implementing it faithfully. Subsequently, the agents either rely on their trust in the algorithm's IR property or have the ability to verify it autonomously. Lastly, the agents behave rationally by joining the collaboration only if it is IR.

We remark that there is an interesting parallelism between our IR collaboration algorithms and truthful mechanisms. It is well known that Vickrey–Clarke–Groves (VCG) mechanism is a truthful mechanism that maximizes social welfare, but since it is hard to compute and to approximate [15], the optimal outcome is replaced by a sub-optimal outcome of an approximation algorithm, and the resulting mechanism is not necessarily truthful. The goal is therefore relaxed to design an efficient approximation algorithm that returns a truthful mechanism.

Contributions and Organization. We formalize the model in Section 2. In Section 3, we demonstrate that any optimal algorithm is individually rational when the baseline is itself. This implies that optimizing for optimality ensures individual rationality for all baseline algorithms.

However, computing (or even approximating) the optimal algorithm is known to be NP-hard. To address this, we then show that the best available tractable approximation algorithm, the greedy algorithm [34, 21], is not individually rational when the baseline is itself. We demonstrate this by presenting an example where joining the collaboration increases the labeling complexity of an agent from $O(1)$ to $\Omega(n)$. In response, we introduce a general approach that can transform any arbitrary baseline algorithm into an IR collaborative algorithm. This conversion ensures that the total label complexity remains competitive with running the baseline algorithm on the entire data set. Furthermore, in Section 4 we present a scheme that converts any IR collaborative algorithm into a strict IR one, guaranteeing the label complexity is strictly lower by joining the collaboration under mild assumptions. When the baseline algorithm is both efficient and approximately optimal, our (strict) IR algorithms efficiently achieve label complexity that is approximately optimal.

1.1   Related Work

The most related work is the recent work of [51], which studies individual rationality in collaborative active learning in a Gaussian Process. While their notion of IR is similar to ours, we focus on query complexity in binary classification. [27] studied incentive compatibility in active learning, where there is a single agent that responds to a learner's query strategically. Our work is situated at the junction of Learning in the presence of strategic behavior and active learning.

Learning in the presence of strategic behavior

encompasses a vast body of research, including [8, 52, 31]. We are particularly driven by prior research in this area, and how to create learning algorithms that incentivize agents to participate while maximizing the overall welfare. For example, *incentivized exploration* in Multi Arm Bandits [35, 37, 38, 39, 20, 19, 5, 4, 6, 32, 33, 42, 7, 48, 47] or MDPs [46], where the principal recommends actions to the agents (in order to explore different alternatives), but the agents ultimately decide whether to follow the given recommendation. This raises the issue of incentives in addition to the exploration-exploitation trade-off. In particular,[3] study this problem in the context of fairness with a group-based regret notion. They show that regret-optimal bandit algorithms can be unfair and design a nearly optimal fair algorithm. Incentivizing agents to share their data has been studied by [50] in federated bandits.

Federated learning

has gained popularity as a method to foster collaboration among large populations of learning agents among else for incentivizing participation and fairness purposes [11, 36, 24, 23, 26, 25, 49]. Our work also addresses fairness, in the sense that if a collaborative algorithm is individually rational, it is fair for all the participating agents. Another related line of research is *kidney exchange* [41, 1, 9, 10, 14, 22], where the goal is to find a maximum match in a directed graph (representing transplant compatibilities between patient–donor pairs). In this problem, incentives arise in the form of individual rationallity when different hospitals have different subsets of patient–donor pairs, and will not join the collaboration if the number of pairs matched by the collaboration is lower than the number of pairs matched they could pair on their own.

Active learning

There are two basic models in active learning– stream-based [28] (where the learner has to determine immediately whether to query the label of the current instance or discard it), and pool-based, which is the basis for our model. Pool-based active learning investigates scenarios in which a learner is confronted with an array of unlabeled data points and the goal is to recover a target function by querying the labels of these points (see [30] for a survey). Active learning has been studied in the context of other societal desiderata such as fairness [44, 16], and safety [17].

To our knowledge, no research has amalgamated these fields to explore strategic constraints in the context of active learning. This is where our work makes a valuable contribution.

2 Preliminaries and Model

Throughout the work, we consider the binary classification problem. Let $\mathcal{X}$ denote the input space, $\mathcal{Y} = \{0, 1\}$ denote the label space, and $\mathcal{H} \subset \mathcal{Y}^{\mathcal{X}}$ denote the hypothesis class.[2] We focus on the *realizable* setting in this work, namely, there exists a target hypothesis $h^* \in \mathcal{H}$ correctly labels every point. In the pool-based active learning setting [30], given a collected unlabeled data set $X = \{x_1, \ldots, x_m\}$, the learning goal is to recover the labels of X. Now just suppose the pool of unlabeled data $x_1, \ldots, x_m$ is available. The possible labelings of these points form a subset of $\{0, 1\}^m$, called the effective hypothesis class, which is

$$\widehat{H} = \{h(X)|h \in \mathcal{H}\},$$

where $h(X) = (h(x_1), \ldots, h(x_m))$ is the labeling of X by h. Note that $|\widehat{H}| \leq 2^m$ and $|\widehat{H}| = \mathcal{O}(m^d)$ if the VC dimension of $\mathcal{H}$ is d.

In this work, we focus on the Bayesian setting [21], where the target hypothesis is chosen in advance from some prior distribution π over $\widehat{H}$. Namely, without any additional information, for any labeling $h \in \widehat{H}$, the probability that h is the correct labeling of X is $\pi(h)$. Since we can eliminate any hypothesis h with $\pi(h) = 0$ before starting to query for labels, we assume w.l.o.g. that $\pi(h) > 0$ for all effective hypotheses in $\widehat{H}$.

We remark that assuming that the unlabeled data X is collected from some distribution D_x, which is essentially a distribution D projected onto its input space, and that this distribution D_x can be accurately classified by a hypothesis in H with VC dimension d, standard generalization guarantees apply when the prior π over H is uniform (see [21] for more details).

Standard active learning model. In the standard pool-based active learning setting, a single agent owns the pool of unlabeled data X. The agent, who knows both $\widehat{H}$ and π, can query the labels of points in X, and her goal is to recover the labeling of X (or to find the target hypothesis) by querying as few points as possible.

A *standard query algorithm* receives as input the prior distribution π and unlabeled data set, X. In each iteration $t = 1, 2, \ldots$, given the history up to time t,

$$\mathcal{F}_t = ((x_1, y_1), \ldots, (x_{t-1}, y_{t-1})) \in (X \times \{0, 1\})^{t-1},$$

---

[2] Results in this work can be directly extended to any active learning problem that can be formalized using a hypothesis class, e.g., multiclass classification.

it selects a point x_t to query and observes its label, y_t. The algorithm stops when all the labels of X are recovered. Alternatively, the algorithm stops when for every two hypotheses $h_1, h_2 \in \widehat{H}$ consistent with $\mathcal{F}_t$ (meaning that $h_1(x_\tau) = h_2(x_\tau) = y_\tau$ for all $\tau = 1, \ldots, t-1$), $h_1(X) = h_2(X)$.

Collaborative active learning model.   In the collaborative setting, we assume there is more than one agent. Formally, there are k agents and each agent i has an individual unlabeled data set X_i such that they together compose the pool, i.e., $\cup_{i \in [k]} X_i = X$, and each can query points from their own set X_i (but cannot query points which are not in their set). The goal of each agent is to recover the true labeling of their own set while performing as few queries as possible. The collaboration protocol, also called principal, who knows $\{X_1, \ldots, X_k\}$, $\widehat{H}$ and π, decides which point should be queried at each iteration, and her goal is to recover all the labels of X using as few queries as possible. We remark that since data points belong to agents, queries of any point $x \in X$ can only be performed by agents whose data set contains x.

The query algorithm in the collaborative setting is similar to that in the standard setting, except that the algorithm needs to coordinate among the agents and decide which agent will query each point as some data points might belong to more than one agent. In this setting, agents can decide to join the collaboration or learn individually at the beginning of the learning. But if they join the collaboration, they commit to follow the instructions of the query algorithm. Therefore, given a prior distribution π over $\widehat{H}$ and a set of agents who would join the collaboration, w.l.o.g. denoted as $\{X_1, \ldots, X_\kappa\}$ for some $\kappa \in [k]$, at time $t = 1, 2, \ldots$, a *collaborative query algorithm* asks agent $i_t \in [\kappa]$ such that $x_t \in X_{i_t}$ to query point x_t, and observes its label, y_t; the algorithm stops when the labels of points in $\cup_{i \in [\kappa]} X_i$ are completely recovered.

It is straightforward to check that standard query algorithms are a special case of collaborative query algorithms when there is a single agent, i.e., $k = 1$. Additionally, a standard algorithm can also be run over multiple agents by considering the union of their data $\cup_{i \in [\kappa]} X_i$ as a single agent. Hence, we omit "standard" or "collaborative" in a query algorithm when it is clear from the context how many agents are involved.

For any collaborative algorithm $\mathcal{A}$, given an input π and any collection of unlabeled data sets $X_1, \ldots, X_\kappa \subseteq X$ of size $\kappa \geq 1$, we denote by $Q(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\}, h)$ the label complexity (number of label queries) of $\mathcal{A}(\pi, \{X_1, \ldots, X_\kappa\})$ when the target hypothesis is h. For randomized algorithms, the label complexity is taken expectation over the randomness of the algorithm. We define the label complexity as follows.

▶ **Definition 2** (Label complexity). *Given any fixed unlabeled pool and effective hypothesis class $(X, \widehat{H})$, for any algorithm $\mathcal{A}$, prior distribution π over $\widehat{H}$ and any collection of unlabeled data sets $X_1, \ldots, X_\kappa \subseteq X$ of size $\kappa \geq 1$, the label complexity of $\mathcal{A}$ with $(\pi, \{X_1, \ldots, X_\kappa\})$ as input, denoted by $Q(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\})$, is the expected number of label queries when h is drawn from the prior π, i.e.,*

$$Q(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\}) = \mathbb{E}_{h \sim \pi} \left[Q(\mathcal{A}, h, \{X_1, \ldots, X_\kappa\}) \right] .$$

For each agent $i \in [\kappa]$ in the collaboration, we let $Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\})$ denote the expected number of queries performed by agent i.

For any $(\pi, \{X_1, \ldots, X_\kappa\})$, let $Q^*(\pi, \{X_1, \ldots, X_\kappa\}) = \min_{\mathcal{A}} Q(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\})$ denote the optimal query complexity. An algorithm $\mathcal{A}$ is said to be *optimal* if $Q(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\}) = Q^*(\pi, \{X_1, \ldots, X_\kappa\})$ for any prior distribution π and $X_1, \ldots, X_\kappa$.

Rational agents. We assume that agents have access to a baseline algorithm and are able to run it on their own local data. Agents can decide to join the collaboration or run the baseline individually at the beginning of the learning. If they join the collaboration, they commit to follow the instructions of the query algorithm. Each agent is incentivized to join the collaboration if she could perform fewer label queries (assuming that all others join the collaboration) by pulling out and running the baseline $\mathcal{A}$ individually. Formally,

▶ **Definition 3** (Individual rationality). *In a collaborative learning problem with prior distribution π and k agents $\{X_1, \ldots, X_k\}$, given a baseline algorithm $\mathcal{A}$, a collaborative algorithm $\mathcal{A}'$ is individually rational (IR) if*

$$Q_i(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) \leq Q(\mathcal{A}, \pi, \{X_i\}), \forall i \in [k] \,. \tag{1}$$

We say $\mathcal{A}'$ is *strictly individually rational* (henceforth, SIR) if

$$Q_i(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) < Q(\mathcal{A}, \pi, \{X_i\}), \forall i \in [k] \,.$$

We remark that in addition to their own sets, each agent knows all the unlabeled data sets, $\{X_1, \ldots, X_k\}$, and the prior distribution π, otherwise, they will not be able to compute $Q_i(\mathcal{A}', \pi, \{X_1, \ldots, X_k\})$. The principal also has access to $\{X_1, \ldots, X_k\}$ and π, and can therefore make sure these constraints are satisfied.

It is worth noting that our model can also accommodate different individual baseline algorithms. We briefly discuss this in Section 5.

An alternative interpretation of the problem in a game theoretic framework is as follows: each agent has a strategy space of two strategies, joining the collaboration and not. The utility of an agent that performs Q queries is $-Q$. If the algorithm $\mathcal{A}$ is IR, then the case of all agents joining the collaboration is a Nash equilibrium (since switching to not joining will not increase their utility). If $\mathcal{A}$ is SIR, then all agents joining the collaboration is a strict Nash equilibrium.

3 Construction of IR Collaborative Algorithms

When agents are limited to a poor baseline algorithm, e.g., randomly selecting points to query, the principal can simply incentivize agents to collaborate by using a superior algorithm that requires fewer labeling efforts. We therefore start by considering optimal baseline algorithms in Section 3.1. If we are able to find an IR collaborative algorithm for an optimal baseline algorithm, OPT, then it must be IR w.r.t. all baseline algorithms. We demonstrate that, surprisingly, the optimal algorithm OPT is IR given that the baseline algorithm is OPT itself. Since computing an optimal algorithm is known to be NP-hard, we continue by considering the best-known approximation algorithm, the greedy algorithm. In Section 3.2, we show that given the greedy algorithm as baseline, the collaboration protocol that runs the greedy algorithm is not IR. Then in Section 3.3, we provide a general scheme that transforms any baseline algorithm into an IR algorithm while maintaining a comparable label complexity.

3.1 Optimality Implies Universal Individual Rationality

Incorporating individual rationality as an additional constraint to optimality usually requires additional effort in certain settings, e.g., in online learning by [13]. However, in our specific setting, optimality does not contradict the individual rationality property. That is, an optimal algorithm will not increase any agent's label complexity to benefit other agents. In fact, optimizing for optimality implies achieving individual rationality for all baseline algorithms.

▶ **Theorem 4.** *For any optimal collaborative algorithm OPT, we have*

$$Q_i(OPT, \pi, \{X_1, \ldots, X_k\}) \leq Q(OPT, \pi, \{X_i\}) = Q^*(\pi, \{X_i\}), \forall i \in [k].$$

Therefore, OPT is IR w.r.t. any baseline algorithm.

We prove the theorem by contradiction. If OPT is not IR for the baseline being OPT, then there exists an agent i such that $Q(\text{OPT}, \pi, \{X_i\}) < Q_i(\text{OPT}, \pi, \{X_1, \ldots, X_k\})$. In this case, we can construct a new algorithm by first running OPT over $\{X_i\}$ (to recover the labels of X_i) and then running $\text{OPT}(\pi, \{X_1, \ldots, X_k\}\})$ and replacing agent i's queries with the recovered the labels of X_i. This new algorithm incurs a strictly smaller label complexity than OPT, which is a contradiction to the optimality of OPT. The formal proof is deferred to Appendix A. Unfortunately, computing an optimal query algorithm is not just NP-hard, but also hard to approximate within a factor of $\Omega(\log(|\widehat{H}|))$ [29, 18]. One of the most popular heuristics to find an approximated solution is greedy.

3.2   The Greedy Algorithm is Not Individually Rational

For standard Bayesian active learning, [34, 21] presented a simple greedy algorithm called generalized binary search (GBS), which chooses a point leading to the most balanced partition of the set of hypotheses consistent with the history. More specifically, at time step t, given the history $\mathcal{F}_t = ((x_1, i_1, y_1), \ldots, (x_{t-1}, i_{t-1}, y_{t-1}))$, let $\text{VS}(\mathcal{F}_t) = \{h \in \widehat{H} | h(x_\tau) = y_\tau, \forall \tau \in [t-1]\}$ denote the set of hypotheses consistent with the history $\mathcal{F}_t$ (often called the version space associated with $\mathcal{F}_t$). Given $\mathcal{F}_t$ and $(\pi, \{X_1, \ldots, X_\kappa\})$ as input, GBS will query

$$x_t = \underset{x \in \cup_{i \in [\kappa]} X_i}{\arg\max} \ \min(\pi(\{h \in \text{VS}(\mathcal{F}_t) | h(x) = 1\}), \pi(\{h \in \text{VS}(\mathcal{F}_t) | h(x) = 0\}))$$

at time t. When referring to GBS as a collaborative algorithm, we complement it with an arbitrary tie-breaking rule for selecting i_t, as GBS itself does not specify how to choose which agent to query. GBS is guaranteed to achieve competitive label complexity with the optimal label complexity.

▶ **Lemma 5** (Optimality of GBS, Theorem 3 of [21]). *For any prior distribution π over $\widehat{H}$ and k agents $\{X_1, \ldots, X_k\}$, the label complexity of GBS satisfies that*

$$Q(GBS, \pi, \{X_1, \ldots, X_k\}) \leq 4Q^*(\pi, \{X_1, \ldots, X_k\}) \ln\left(\frac{1}{\min_{h \in \widehat{H}} \pi(h)}\right).$$

The greedy algorithm GBS not only achieves approximately optimal label complexity, but it is also computationally efficient, with a running time of $\mathcal{O}(m^2 |\widehat{H}|)$. As GBS is the best-known efficient approximation algorithm, it is natural to think that agents would adopt GBS as a baseline.

As we have shown that the optimal algorithm is IR w.r.t. itself, the next natural question is: *Is GBS (as collaboration protocol) individually rational w.r.t. GBS itself?*

We answer this question negatively, even in the case of two agents. Even worse, we present an example in which an agent's label complexity is $\Omega(n)$ when participating in the collaboration, but only $\mathcal{O}(1)$ when not participating.

▶ **Theorem 6.** *For the algorithm of GBS, there exists an instance of (X_1, X_2, π) with $|X_1| = n$, in which agent 1 incurs a label complexity of $Q_1(GBS, \pi, \{X_1, X_2\}) = \Omega(n)$ when participating the collaboration and can achieve $Q(GBS, \pi, \{X_1\}) = \mathcal{O}(1)$ when not participating.*

Intuitively, at each time step, GBS only searches for an x_t which leads to the most balanced partition of the version space, which does not necessarily lead to the optimal point to query. Given additional label information from the other agent, GBS possibly choose a worse point to query. In addition, the label complexity of GBS is upper bounded by the optimal label complexity multiplied by a logarithmic factor. It is possible that the agent achieves a smaller multiplicative factor by running GBS individually and a larger factor in the collaboration. To prove the theorem, we construct an instance in which there exists a hypothesis with a prior probability of $1/4$, such that if GBS runs on $\{X_1, X_2\}$ and this hypothesis is the target, GBS will query almost all the points in X_1 (in a particular order) before returning this hypothesis. We show this part by induction. Additionally, we compute the query tree by running GBS solely on X_1 and use it to show that in this case, GBS has an expected query complexity of $O(1)$. The full construction of the instance and the proof of Theorem 6 is deferred to Appendix B.

3.3 A Scheme of Converting Algorithms to IR Algorithms

Given that the greedy algorithm has been proven to be not individually rational w.r.t. itself, we raise the following question: *Is it possible to develop a general scheme that can generate an IR algorithm given any baseline algorithm?* In this section, we propose such a scheme that addresses this question. Moreover, given a baseline algorithm $\mathcal{A}$, the resulting IR algorithm can achieve a label complexity comparable to implementing the baseline algorithm over all agents, i.e., $\mathcal{A}(\pi, \{X_1, \ldots, X_k\})$. It is important to note that we aim for the label complexity to be comparable to $Q(\mathcal{A}, \pi, \{X_1, \ldots, X_k\})$ rather than $\sum_{i \in [k]} Q(\mathcal{A}, \pi, X_i)$, as the latter holds true by individual rationality. Given an efficient approximately optimal algorithm as baseline (e.g., GBS), our scheme can provide an algorithm that simultaneously exhibits individual rationality, efficiency, and approximately optimal label complexity.

For any baseline algorithm $\mathcal{A}$, we define a new algorithm B2IR($\mathcal{A}$), which runs $\mathcal{A}$ as a subroutine. Basically, we first calculate the label complexity of agent i both when she is in collaboration with all the other agents, i.e., $Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\})$, and when she is not in collaboration, i.e., $Q(\mathcal{A}, \pi, \{X_i\})$, for all $i \in [k]$. By doing so, we can distinguish which agents can benefit from collaboration when running $\mathcal{A}$ and which cannot. We denote the set of agents who cannot benefit from collaboration with all others when running $\mathcal{A}$ as $S = \{i | Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}) > Q(\mathcal{A}, \pi, \{X_i\})\}$. For those who do not benefit from the collaboration, we just run $\mathcal{A}$ on their own data. For those who benefit from collaborating with the others together, we run $\mathcal{A}$ over all agents $[k]$– Only whenever $\mathcal{A}(\pi, \{X_1, \ldots, X_k\})$ asks to query the label of a point belonging to some $i \in S$, since we already recovered the labels of X_i, we just feed $\mathcal{A}(\pi, \{X_1, \ldots, X_k\})$ with this label without actually asking agent i to query. The detailed algorithm is described in Algorithm 1.

▶ **Theorem 7.** *For any baseline algorithm $\mathcal{A}$, the algorithm B2IR($\mathcal{A}$) satisfies the following properties:*

- ▬ ***IR property:*** *B2IR($\mathcal{A}$) is individually rational w.r.t. the baseline algorithm $\mathcal{A}$.*
- ▬ ***Efficiency:*** *B2IR($\mathcal{A}$) runs in $\mathcal{O}(kT_{\mathcal{A},Q} + mT_{\mathcal{A},0})$ time, where $T_{\mathcal{A},0}$ is the time of computing (i_t, x_t) at each time t for $\mathcal{A}$ and $T_{\mathcal{A},Q}$ is the maximum time of computing $Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\})$ for an agent i, unlabeled data $\{X_1, \ldots, X_\kappa\}$, and algorithm $\mathcal{A}$.*
- ▬ ***Label complexity:*** $Q(B2IR(\mathcal{A}), \pi, \{X_1, \ldots, X_k\}) \leq Q(\mathcal{A}, \pi, \{X_1, \ldots, X_k\})$.

🟨 **Algorithm 1** B2IR.

1: **input:** A query algorithm $\mathcal{A}$, set $\{X_1, \ldots, X_k\}$ and prior π over $\widehat{H}$
2: For each $i \in [k]$, calculate $Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\})$ and $Q(\mathcal{A}, \pi, \{X_i\})$.
3: Let $S \leftarrow \{i | Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}) > Q(\mathcal{A}, \pi, \{X_i\})\}$ and $X_S \leftarrow \cup_{i \in S} X_i$ // the agents who do not benefit from collaboration
4: **for** each $i \in S$ **do** $Y_i \leftarrow$ Run $\mathcal{A}$ over $\{X_i\}$ // recover the labels for agent i
5: **for** $t = 1, \ldots$ **do**
6: $\quad$ $(i_t, x_t) \leftarrow$ the querying agent and the query point from $\mathcal{A}(\pi, \{X_1, \ldots, X_k\})$
7: $\quad$ **if** $i_t \in S$ **then** Feed the label of x_t from Y_{i_t} // we already recovered the labels of X_S
8: $\quad$ **else** Ask agent i_t to query the label of x_t
9: **end for**

The proof follows the algorithm description immediately. Note that when the baseline is GBS, we have $T_{\mathrm{GBS},0} = \mathcal{O}(m)$. We can compute $Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_\kappa\})$ by simulating over all effective hypotheses $h \in \widehat{H}$. For each h, we will query at most m rounds. Therefore, we have $T_{\mathrm{GBS},Q} = \mathcal{O}(m^2 |\widehat{H}|)$ and we can run B2IR(GBS) in $\mathcal{O}(km^2 |\widehat{H}|)$ time. Using GBS as the baseline, we derive the following corollary.

▶ **Corollary 8.** *Given GBS as the baseline, B2IR(GBS) is IR; runs in $\mathcal{O}(km^2 |\widehat{H}|)$ time; and satisfies that $Q(B2IR(GBS), \pi, \{X_1, \ldots, X_k\}) \leq 4Q^*(\pi, \{X_1, \ldots, X_k\}) \ln(\frac{1}{\min_{h \in \widehat{H}} \pi(h)})$.*

4 Converting Algorithms to SIR Algorithms

In Section 3, we provided a generic scheme for constructing an IR algorithm given any baseline algorithm. In this section, we focus on constructing SIR algorithms given IR algorithms. Since strict individual rationality requires that agents strictly benefit from collaboration, this is impossible without further assumptions. For example, consider a set of agents who only have one single independent point in their own sets and a prior distribution that is uniform over all labelings. In this case, each agent, regardless of whether she collaborates or not, has a label complexity of 1 and cannot *strictly* benefit from collaboration as the other agents cannot obtain information about her data.

Now, let us consider a notion weaker than SIR, called i-partially SIR, in which only agent i strictly benefits from the collaboration, and any other agent $j \neq i$ does not get worse by joining the collaboration. More formally,

▶ **Definition 9** (Partially SIR algorithms). *For any baseline algorithm $\mathcal{A}$, for all $i \in [k]$, an algorithm $\mathcal{O}_i$ is i-partially SIR, if $\mathcal{O}_i$ satisfies that*

$$Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}) < Q(\mathcal{A}, \pi, \{X_i\}),$$

and

$$Q_j(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}) \leq Q(\mathcal{A}, \pi, \{X_j\}), \forall j \in [k] \setminus \{i\}.$$

If we are given an i-partially SIR algorithm $\mathcal{O}_i$ for each i, then we can construct a SIR algorithm by running a mixture of an IR algorithm $\mathcal{A}'$ (e.g., B2IR($\mathcal{A}$) in Algorithm 1) and $\{\mathcal{O}_i | i \in [k]\}$ with the label complexity a little (arbitrarily small) higher than that of $\mathcal{A}'$.

▶ **Lemma 10.** *For any baseline algorithm $\mathcal{A}$, given an IR algorithm $\mathcal{A}'$ and partially SIR algorithms $\{\mathcal{O}_i | i \in [k]\}$, for any $\varepsilon > 0$, let $\mathcal{A}''_\varepsilon$ be the algorithm of running $\mathcal{A}'$ with probability $(1 - \frac{\varepsilon}{n})$ and running $\mathcal{O}_i$ with probability $\frac{\varepsilon}{kn}$. Then $\mathcal{A}''_\varepsilon$ satisfies the following properties.*

- *SIR property: $\mathcal{A}''_\varepsilon$ is SIR with respect to the baseline algorithm $\mathcal{A}$.*
- *Label complexity: $Q(\mathcal{A}''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) \leq Q(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) + \varepsilon$.*

The proof is straightforward from the definition, and we include it in Appendix C for completeness. Since a SIR algorithm is also i-partially SIR for all $i \in [k]$, constructing a SIR algorithm is equivalent to constructing a set of partially SIR algorithms $\{\mathcal{O}_i | i \in [k]\}$. Therefore, the problem of constructing a SIR algorithm is reduced to constructing partially SIR algorithms $\{\mathcal{O}_i | i \in [k]\}$.

For the remainder of this section, we will present the SIR results for an optimal baseline algorithm in Section 4.1, where we propose a sufficient and necessary assumption for the existence of SIR algorithms and then provide a SIR algorithm. This algorithm is SIR w.r.t. any baseline algorithm but again, computationally inefficient. In Section 4.2, we provide a general scheme that transforms any baseline algorithm into a SIR algorithm.

4.1 A Universal SIR Algorithm for Any Baseline Algorithm

Constructing a universal SIR algorithm w.r.t. any baseline is equivalent to constructing a SIR algorithm for an optimal baseline. For the existence of SIR algorithms given any optimal baseline algorithm, we propose the following assumption, which is sufficient and necessary. We include the proof for the necessity of this assumption in Appendix D. The sufficiency of this assumption will be verified immediately after we construct a SIR algorithm.

▶ **Assumption 1.** We assume that for any $i \in [k]$, the optimal label complexity of agent i given the information regarding the labels of all other agents is strictly smaller than that without this additional information, i.e., $Q^*(\pi, \{X_i\}) - \mathbb{E}_{h \sim \pi} [Q^*(\pi_{h,-i}, \{X_i\})] > 0$, where $\pi_{h,-i}$ is the posterior distribution of π after observing $\{(x, h(x)) | x \in \cup_{j \neq i} X_j\}$.

According to Lemma 10, we can construct a SIR algorithm by constructing a set of partially SIR algorithms $\{\mathcal{O}_i | i \in [k]\}$.

Let $\mathcal{O}_i$ be the algorithm of running an optimal algorithm OPT over $(\pi, \{X_j | j \neq i\})$ first, then given the query-label history of $\{(x, h(x)) | x \in \cup_{j \neq i} X_j\}$ for some $h \in \widehat{H}$, run OPT over $(\pi_{h,-i}, \{X_i\})$. Then it immediately follows that $\mathcal{O}_i$ is i-partially SIR from Assumption 1. Let OPT''_ε denote the algorithm of of running OPT with probability $(1 - \frac{\varepsilon}{n})$ and running $\mathcal{O}_i$ with probability $\frac{\varepsilon}{kn}$ for all $i \in [k]$. By Lemma 10, we have

▶ **Corollary 11.** *Under Assumption 1, for any $\varepsilon > 0$, OPT''_ε is SIR w.r.t. OPT and satisfies*

$$Q(OPT''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) \leq Q^*(\pi, \{X_1, \ldots, X_k\}) + \varepsilon.$$

In addition, OPT''_ε is SIR w.r.t. any baseline algorithm $\mathcal{A}$ as

$$Q_i(OPT''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) < Q^*(\pi, \{X_i\}) \leq Q(\mathcal{A}, \pi, \{X_i\}).$$

4.2 A Scheme of Converting Algorithms to SIR Algorithms

As mentioned before, computing an optimal algorithm is NP-hard. Assumption 1 assumes that collaboration can strictly benefit agents when the collaboration protocol can compute the optimal algorithm given $\pi_{h,-i}$. Hence, the assumption does not take the computational issue into consideration and thus might not be enough for the existence of an efficient SIR algorithm w.r.t. an efficient approximation algorithm like GBS.

Instead, we propose prior-independent assumption that is sufficient for the existence of efficient SIR algorithms when we are given an efficient baseline and an efficient IR algorithm w.r.t. the baseline. Basically, we assume that, there exists an effective hypothesis $h \in \widehat{H}$,

given the information that all other agents are labeled by h, the number of labelings of X_i consistent with the label information is strictly smaller than the total number of labelings of X_i by $\widehat{H}$. Formally, for any $i \in [k]$, let $X_{-i} = \cup_{j \neq i} X_j$ denote the union of all agents' data except agent i. Let $H(X_i) = \{h'(X_i) | h' \in \widehat{H}\}$ denote the effective hypothesis class of X_i, i.e., all labelings of X_i. For any $h \in \widehat{H}$, let $H(X|h) = \{h'(X_i) | h'(X_{-i}) = h(X_{-i}), h' \in \widehat{H}\}$ denote the subset which are consistent with all other agents being labeled by h.

▶ **Assumption 2.** For all $i \in [k]$, there exists an $h \in \widehat{H}$ s.t. the number of labelings of X_i consistent with $(X_{-i}, h(X_{-i}))$ is strictly smaller than the number of labelings by $\widehat{H}$, i.e., $|H(X_i|h)| < |H(X_i)|$.

Intuitively, Assumption 2 means that for every agent i, there exists an hypothesis h such that when $h^* = h$, the cardinality of the set of hypotheses consistent with $(X_{-i}, h(X_{-i}))$ is strictly smaller than $|\widehat{H}|$. We will show that this assumption is sufficient for the *existence* of algorithms satisfying SIR property. Without it, it is unclear if there exist SIR algorithms. The assumption can be easily verified by iterating each $h \in \widehat{H}$ (this is polynomial in $|\widehat{H}|$ and m).

Notice that each deterministic query algorithm $\mathcal{A}$ can be represented as a binary tree, $\mathcal{T}_\mathcal{A}$ whose internal nodes at level t are queries ("what is the x_t's label?"), and whose leaves are labelings as illustrated in Figure 1. Under Assumption 2, we can prune the query tree of $\mathcal{A}(\pi, \{X_i\})$ by removing all subtrees whose leaves are all in $H(X_i) \setminus H(X_i|h)$. We do not need to construct this pruned tree when we implement the algorithm. At time t, we just need to generate an x_t from $\mathcal{A}(\pi, \{X_i\})$, then check if this node should be pruned by checking if all the hypotheses $H(X_i|h)$ agree on the label of x_t. If this is true, it means that we have already recovered the label of x_t and thus we just need to feed the label to the algorithm without actually querying x_t again. Then we can construct a i-partially SIR algorithm $\mathcal{O}_i$ by running B2IR($\mathcal{A}$) over $(\pi, \{X_j | j \neq i\})$ to recover the labeling of X_{-i} first, then running pruned version of $\mathcal{A}(\pi, \{X_i\})$. Note that the implementation also works when $\mathcal{A}$ is randomized.

Consider Example 1, where the hypothesis class $\mathcal{H} = \{x \geq \alpha | \alpha = 0.2, 0.4, 0.5, 0.6, 0.8\}$ with a uniform prior, agent 1 with points $X_1 = \{0.25, 0.5, 0.75\}$ and agent 2 with points $X_2 = \{0.3, 0.45, 0.55, 0.7\}$. When agent 1 runs a binary search, the query tree has 0.5 as a root, then if 0.25 if $h^*(0.5) = 1$, and 0.75 otherwise. Now, algorithm $\mathcal{O}_1$ runs a binary search on X_2 and obtains all the labels of the points in X_2. The hypothesis $h = 1(x \geq 0.5)$ holds $|H(X_1|h)| = 1 < |\widehat{H}(X_1)| = 4$. When h is the labeling function, 0.3 and 0.45 are labeled as negative, and 0.55 and 0.7 are labeled as positive. Then, $\mathcal{O}_1$ will only need agent 1 to query 0.5 as the labels of 0.25 and 0.75 can be inferred and they are pruned in the query tree.

▶ **Lemma 12.** *Under Assumption 2, the algorithm $\mathcal{O}_i$ constructed above is i-partially SIR and runs in time $\mathcal{O}(\mathcal{T}_{B2IR(\mathcal{A})} + m(|\widehat{H}| + T_{\mathcal{A},0}))$ time, where $\mathcal{T}_{B2IR(\mathcal{A})}$ is the running time of B2IR($\mathcal{A}$) and $T_{\mathcal{A},0}$ is the time of computing (i_t, x_t) at each time t for $\mathcal{A}$.*

The proof of Lemma 12 is deferred to Appendix E.

We can then construct an algorithm $\mathcal{A}''_\varepsilon$ by running B2IR($\mathcal{A}$) with probability with probability $(1 - \frac{\varepsilon}{n})$ and running $\mathcal{O}_i$ constructed in the above way with probability $\frac{\varepsilon}{kn}$ for all $i \in [k]$. By combining Lemmas 10 and 12, we derive the following theorem. Then, combining it with Corollary 8, we derive a SIR algorithm for GBS as baseline GBS.

▶ **Theorem 13.** *Under Assumption 2, for any baseline algorithm $\mathcal{A}$, for any $\varepsilon > 0$, $\mathcal{A}''_\varepsilon$ is SIR and satisfies*

$$Q(\mathcal{A}''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) \leq Q(B2IR(\mathcal{A}), \pi, \{X_1, \ldots, X_k\}) + \varepsilon.$$

In addition, Algorithm $\mathcal{A}''_\varepsilon$ runs in $\mathcal{O}(\mathcal{T}_{B2IR(\mathcal{A})} + m(|\widehat{H}| + T_{\mathcal{A},0}))$ time.

▶ **Corollary 14.** *Given GBS as the baseline, Algorithm GBS''_ε is SIR; runs in $\mathcal{O}(km^2|\widehat{H}|)$ time; and satisfies that*

$$Q(GBS''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) \leq 4Q^*(\pi, \{X_1, \ldots, X_k\}) \ln\left(\frac{1}{\min_{h \in \mathcal{H}} \pi(h)}\right) + \varepsilon.$$

5 Discussion

In this paper, we have initiated the study of collaboration in active learning in the presence of incentivized agents. We first show that an optimal collaborative algorithm is IR w.r.t. any baseline algorithm while approximate algorithms are not. Then we provide meta-algorithms capable of producing IR/SIR algorithms given any baseline algorithm as input.

Our model and algorithms can also allow different agents to have different baseline algorithms – Whenever the principal plans to run an algorithm on a union of datasets, she can simply check which baseline algorithm has the lowest expected query complexity on this union, and run it. When she needs to run an algorithm on a dataset of an individual, she can simply run their baseline. This way the (S)IR is preserved.

There are a few problems we leave open. First, relaxing the assumption that each agent i has full knowledge of X_{-i} (e.g., due to privacy concerns). Second, relaxing the assumption that agents provide reliable labels. Third, deriving results in non-realizable settings. Realizability is a standard assumption in learning theory at large and particularly within active learning as highlighted in the classical machine learning theory textbook by [43] and the active learning theory survey by [30]. Moreover, realizability has also been adopted in the collaborative learning setting (e.g., [12]). The reason is that without realizability, additional complications might arise in collaboration such as why collaboration would yield benefits. In general, we believe that additional assumptions would be required to relax this assumption. Forth, towards a more game theory orientation, it would be interesting to design collaborative algorithms in a setting where agents can form coalitions. Finally, finding a necessary and sufficient assumption(s) for the existence of efficient SIR algorithms will be an interesting direction (and we have found a sufficient one in this work).

──── **References** ────

1   Itai Ashlagi and Alvin E. Roth. Individual rationality and participation in large scale, multi-hospital kidney exchange. In *Proceedings 12th ACM Conference on Electronic Commerce (EC-2011)*, 2011.

2   Nicolas Aussel, Sophie Chabridon, and Yohan Petetin. Combining federated and active learning for communication-efficient distributed failure prediction in aeronautics. *arXiv preprint arXiv:2001.07504*, 2020.

3   Jackie Baek and Vivek F. Farias. Fair exploration via axiomatic bargaining. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 22034–22045, 2021.

4   Gal Bahar, Omer Ben-Porat, Kevin Leyton-Brown, and Moshe Tennenholtz. Fiduciary bandits. In *International Conference on Machine Learning*, 2019.

5   Gal Bahar, Rann Smorodinsky, and Moshe Tennenholtz. Economic recommendation systems: One page abstract. In *Proceedings of the 2016 ACM Conference on Economics and Computation, EC*, page 757, 2016.

6   Gal Bahar, Rann Smorodinsky, and Moshe Tennenholtz. Social learning and the innkeeper's challenge. In *Proceedings of the 2019 ACM Conference on Economics and Computation*,

EC '19, page 153–170, New York, NY, USA, 2019. Association for Computing Machinery. `doi:10.1145/3328526.3329569`.

7   Kiarash Banihashem, MohammadTaghi Hajiaghayi, Suho Shin, and Aleksandrs Slivkins. Bandit social learning: Exploration under myopic behavior, 2023. `arXiv:2302.07425`.

8   Omer Ben-Porat and Rotem Torkan. Learning with exposure constraints in recommendation systems. In *Proceedings of the ACM Web Conference 2023*, WWW '23, 2023.

9   Avrim Blum, Ioannis Caragiannis, Nika Haghtalab, Ariel D. Procaccia, Eviatar B. Procaccia, and Rohit Vaish. Opting into optimal matchings. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, page 2351–2363, USA, 2017. Society for Industrial and Applied Mathematics.

10   Avrim Blum and Paul Gölz. Incentive-compatible kidney exchange in a slightly semi-random model. In *EC '21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*. ACM, 2021.

11   Avrim Blum, Nika Haghtalab, Richard Lanas Phillips, and Han Shao. One for one, or all for all: Equilibria and optimality of collaboration in federated learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, 2021.

12   Avrim Blum, Nika Haghtalab, Ariel D Procaccia, and Mingda Qiao. Collaborative pac learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL: `https://proceedings.neurips.cc/paper_files/paper/2017/file/186a157b2992e7daed3677ce8e9fe40f-Paper.pdf`.

13   Avrim Blum and Thodoris Lykouris. Advancing Subgroup Fairness via Sleeping Experts. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 55:1–55:24, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. `doi:10.4230/LIPIcs.ITCS.2020.55`.

14   Avrim Blum and Yishay Mansour. Kidney exchange and endless paths: On the optimal use of an altruistic donor. *CoRR*, abs/2010.01645, 2020. `arXiv:2010.01645`.

15   David Buchfuhrer, Shaddin Dughmi, Hu Fu, Robert Kleinberg, Elchanan Mossel, Christos H. Papadimitriou, Michael Schapira, Yaron Singer, and Christopher Umans. Inapproximability for vcg-based combinatorial auctions. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*. SIAM, 2010.

16   Romain Camilleri, Andrew Wagenmaker, Jamie Morgenstern, Lalit Jain, and Kevin Jamieson. Fair active learning in low-data regimes. *CoRR*, abs/2312.08559, 2023. `doi:10.48550/ARXIV.2312.08559`.

17   Romain Camilleri, Andrew Wagenmaker, Jamie H. Morgenstern, Lalit Jain, and Kevin G. Jamieson. Active learning with safety constraints. In *NeurIPS*, 2022.

18   Venkatesan T Chakaravarthy, Vinayaka Pandit, Sambuddha Roy, Pranjal Awasthi, and Mukesh Mohania. Decision trees for entity identification: Approximation algorithms and hardness results. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 53–62, 2007.

19   Yeon-Koo Che and Johannes Hörner. Optimal design for social learning, 2013.

20   Lee Cohen and Yishay Mansour. Optimal algorithm for bayesian incentive-compatible exploration. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, EC '19, page 135–151, New York, NY, USA, 2019. Association for Computing Machinery. `doi:10.1145/3328526.3329581`.

21   Sanjoy Dasgupta. Analysis of a greedy active learning strategy. *Advances in neural information processing systems*, 17, 2004.

22   John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. *Manag. Sci.*, 2019.

23 Kate Donahue, Alexandra Chouldechova, and Krishnaram Kenthapadi. Human-algorithm collaboration: Achieving complementarity and avoiding unfairness. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1639–1656, New York, NY, USA, 2022. Association for Computing Machinery. `doi:10.1145/3531146.3533221`.

24 Kate Donahue and Jon Kleinberg. Model-sharing games: Analyzing federated learning under voluntary participation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

25 Kate Donahue and Jon Kleinberg. Fairness in model-sharing games. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 3775–3783, New York, NY, USA, 2023. Association for Computing Machinery. `doi:10.1145/3543507.3583483`.

26 Kate Donahue and Jon M. Kleinberg. Optimality and stability in federated learning: A game-theoretic approach. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, 2021.

27 Federico Echenique and Siddharth Prasad. Incentive compatible active learning. *arXiv preprint arXiv:1911.05171*, 2019.

28 Yoav Freund, H. Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Mach. Learn.*, 1997.

29 Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal bayesian active learning with noisy observations. *Advances in Neural Information Processing Systems*, 23, 2010.

30 Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends® in Machine Learning*, 7, 2014.

31 Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, pages 111–122, 2016.

32 Nicole Immorlica, Jieming Mao, Aleksandrs Slivkins, and Zhiwei Steven Wu. Bayesian exploration with heterogeneous agents. In *The World Wide Web Conference*, WWW '19, page 751–761, New York, NY, USA, 2019. Association for Computing Machinery. `doi:10.1145/3308558.3313649`.

33 Nicole Immorlica, Jieming Mao, Aleksandrs Slivkins, and Zhiwei Steven Wu. Incentivizing exploration with selective data disclosure. In *Proceedings of the 21st ACM Conference on Economics and Computation*, EC '20, page 647–648, New York, NY, USA, 2020. Association for Computing Machinery. `doi:10.1145/3391403.3399487`.

34 S Rao Kosaraju, Teresa M Przytycka, and Ryan Borgstrom. On an optimal split tree problem. In *Algorithms and Data Structures: 6th International Workshop, WADS'99 Vancouver, Canada, August 11–14, 1999 Proceedings*, pages 157–168. Springer, 2002.

35 Ilan Kremer, Yishay Mansour, and Motty Perry. Implementing the "wisdom of the crowd". *J. of Political Economy*, 122:988–1012, October 2014. Preliminary version appeared in *ACM Conf. on Economics and Computation*, 2014.

36 Lingjuan Lyu, Xinyi Xu, Qian Wang, and Han Yu. *Collaborative Fairness in Federated Learning*, pages 189–204. Springer International Publishing, Cham, 2020. `doi:10.1007/978-3-030-63076-8_14`.

37 Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15, pages 565–582, 2015.

38 Yishay Mansour, Aleksandrs Slivkins, Vasilis Syrgkanis, and Zhiwei Steven Wu. Bayesian exploration: Incentivizing exploration in bayesian games. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 661–661, New York, NY, USA, 2016. ACM. `doi:10.1145/2940716.2940755`.

39 Yishay Mansour, Aleksandrs Slivkins, and Zhiwei Steven Wu. Competing Bandits: Learning Under Competition. In *Innovations in Theoretical Computer Science Conference (ITCS 2018)*, 2018.

40   Ronza Najjar-Debbiny, Naomi Gronich, Gabriel Weber, Johad Khoury, Maisam Amar, Nili Stein, Lee Hilary Goldstein, and Walid Saliba. Effectiveness of Paxlovid in Reducing Severe Coronavirus Disease 2019 and Mortality in High-Risk Patients. *Clinical Infectious Diseases*, 2022.

41   Alvin E. Roth, Tayfun Sönmez, and M. Utku Ünver. Kidney Exchange*. *The Quarterly Journal of Economics*, 2004.

42   Mark Sellke and Aleksandrs Slivkins. The price of incentivizing exploration: A characterization via thompson sampling and sample complexity. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, EC '21, page 795–796, New York, NY, USA, 2021. Association for Computing Machinery. `doi:10.1145/3465456.3467549`.

43   Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

44   Jie Shen, Nan Cui, and Jing Wang. Metric-fair active learning. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, Proceedings of Machine Learning Research, 2022.

45   Pengcheng Shen, Chunguang Li, and Zhaoyang Zhang. Distributed active learning. *IEEE Access*, 4:2572–2579, 2016.

46   Max Simchowitz and Aleksandrs Slivkins. Exploration and incentives in reinforcement learning, 2023. `arXiv:2103.00360`.

47   A. Slivkins. *Introduction to Multi-Armed Bandits*. Foundations and Trends in Machine Learning Series. Now Publishers, 2019. URL: `https://books.google.com/books?id=6ViCzQEACAAJ`.

48   Aleksandrs Slivkins. Incentivizing exploration via information asymmetry. *ACM Crossroads*, 24(1):38–41, 2017. `doi:10.1145/3123744`.

49   Xinran Wang, Qi Le, Ahmad Faraz Khan, Jie Ding, and Ali Anwar. A framework for incentivized collaborative learning. *arXiv preprint arXiv:2305.17052*, 2023.

50   Zhepei Wei, Chuanhao Li, Haifeng Xu, and Hongning Wang. Incentivized communication for federated bandits. *arXiv preprint arXiv:2309.11702*, 2023.

51   Xinyi Xu, Zhaoxuan Wu, Arun Verma, Chuan Sheng Foo, and Bryan Kian Hsiang Low. FAIR: fair collaborative active learning with individual rationality for scientific discovery. In *International Conference on Artificial Intelligence and Statistics, 25-27 April 2023, Palau de Congressos, Valencia, Spain*, 2023.

52   Hanrui Zhang, Yu Cheng, and Vincent Conitzer. Efficient algorithms for planning with participation constraints. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, page 1121–1140, New York, NY, USA, 2022. Association for Computing Machinery. `doi:10.1145/3490486.3538280`.

A   Proof of Theorem 4

Proof. For any randomized optimal algorithm OPT, every realization of the internal randomness of OPT must have the same query complexity and be optimal. Otherwise, there exists a realization with query complexity smaller than OPT, which conflicts with that OPT is optimal. Therefore, it suffices to prove the theorem for deterministic optimal collaborative algorithms.

We will prove the theorem for deterministic algorithms by contradiction. Suppose that there exists a deterministic optimal collaborative algorithm OPT that is not individually rational w.r.t. running itself as baseline. Hence, there exists an agent $i \in [k]$ such that $Q_i(\text{OPT}, \pi, \{X_1, \ldots, X_k\}) > Q(\text{OPT}, \pi, \{X_i\})$. In this case, we can construct another algorithm $\mathcal{A}'$ with smaller label complexity, which will contradict the optimality of OPT.

The basic idea of $\mathcal{A}'$ is to run OPT over $(\pi, \{X_i\})$ first and to recover the labels of X_i. Then, $\mathcal{A}'$ simulates $\text{OPT}(\pi, \{X_1, \ldots, X_k\})$ and asks $\text{OPT}(\pi, \{X_1, \ldots, X_k\})$ what point to query. But whenever $\text{OPT}(\pi, \{X_1, \ldots, X_k\})$ asks to query the label of some point in X_i,

since we already know the labeling of X_i, we can just feed $\mathrm{OPT}(\pi, \{X_1, \ldots, X_k\})$ with these labels without actually asking agent i to query them.

Thus, the label complexity of $\mathcal{A}'$ is

$$
\begin{aligned}
Q(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) &= Q(\mathrm{OPT}, \pi, \{X_i\}) + \sum_{j:j\neq i} Q_j(\mathrm{OPT}, \pi, \{X_1, \ldots, X_k\}) \\
&< Q_i(\mathrm{OPT}, \pi, \{X_1, \ldots, X_k\}) + \sum_{j:j\neq i} Q_j(\mathrm{OPT}, \pi, \{X_1, \ldots, X_k\}) \\
&= Q(\mathrm{OPT}, \pi, \{X_1, \ldots, X_k\}) = Q^*(\pi, \{X_1, \ldots, X_k\}),
\end{aligned}
$$

where the first inequality holds due to that OPT is not IR and the last equality holds since OPT is optimal. Since $Q^*(\pi, \{X_1, \ldots, X_k\}) \leq Q(\mathcal{A}', \pi, \{X_1, \ldots, X_k\})$ by definition, there is a contradiction. ◀

B   Proof of Theorem 6

Proof. The construction is inspired by [21]. Consider $k = 2$ and let the unlabeled data set of agent 1 be

$$
X_1 = \{(0,1,0),(0,2,0),(0,0,1),(0,0,2),\ldots,(0,0,n)\}
$$

for some $n \in \mathbb{N}_+$.

Let the unlabeled data set of agent 2 be

$$
X_2 = \{(1,0,0)\}.
$$

Let the unlabeled pool $X = X_1 \cup X_2$. Let $h_{i,j,l}$ denote the hypothesis which labels $(i,0,0),(0,j,0),(0,0,l)$ as 1 and the rest as 0.

Let the hypothesis class be $\mathcal{H} = \{h_{i,j,l} | i \in \{0,1\}, j \in [2], l \in [n]\}$.

Let the prior distribution $\pi_0 = \pi$ be defined as follows:

$$
\begin{cases}
\pi(h_{0,0,0}) = \frac{1}{4} \\
\pi(h_{0,j,l}) = \frac{1}{4\cdot 3^l} & \text{for } j = 1,2, l = 1,\ldots,n-1 \\
\pi(h_{0,j,n}) = \frac{1}{8\cdot 3^{n-1}} & \text{for } j = 1,2 \\
\pi(h_{1,1,l}) = \frac{1}{3^l} & \text{for } l = 1,\ldots,n-1 \\
\pi(h_{1,1,n}) = \frac{1}{2\cdot 3^{n-1}}.
\end{cases}
$$

Now we show that the label complexity of agent 1 in the collaboration is $Q_1(\mathrm{GBS}, \pi, \{X_1, X_2\}) = \Omega(n)$. While the label complexity of running GBS itself is $Q(\mathrm{GBS}, \pi, \{X_1\}) = \mathcal{O}(1)$.

Label complexity of agent 1 in the collaboration

Let VS denote the version space. And for any point x, let $\mathrm{VS}_x^+ = \{h \in \mathrm{VS} | h(x) = 1\}$ denote the subset of the version space which labels x by 1. Similarly, let $\mathrm{VS}_x^- = \{h \in \mathrm{VS} | h(x) = 0\}$.

Now let us consider the length of the path in the query tree when the target hypothesis is $h_{0,0,0}$.

A-priori (before starting to query), for point $(1,0,0)$, we have

$$
\pi(\mathrm{VS}_{(1,0,0)}^+) = \sum_{l=1}^n \pi(h_{1,1,l}) = \frac{1}{3} + \frac{1}{3^2} + \ldots + \frac{1}{3^{n-1}} + \frac{1}{2\cdot 3^{n-1}} = \frac{1}{2}.
$$

For point $(0, 1, 0)$, we have

$$\pi(\mathrm{VS}^+_{(0,1,0)}) = \sum_{l=1}^{n}(\pi(h_{0,1,l}) + \pi(h_{1,1,l})) = \sum_{l=1}^{n} \pi(h_{0,1,l}) + \pi(\mathrm{VS}^+_{(1,0,0)}) > \frac{1}{2}.$$

For point $(0, 2, 0)$, we have

$$\pi(\mathrm{VS}^+_{(0,2,0)}) = \sum_{l=1}^{n} \pi(h_{0,2,l}) = \frac{1}{4 \cdot 3} + \frac{1}{4 \cdot 3^2} + \ldots + \frac{1}{4 \cdot 3^{n-1}} + \frac{1}{8 \cdot 3^{n-1}} = \frac{1}{8}.$$

For other points $(0, 0, l)$ for $l \in [n-1]$, we have $\pi(\mathrm{VS}^+_{(0,0,l)}) = \frac{1}{4 \cdot 3^l} + \frac{1}{3^l} = \frac{5}{4 \cdot 3^l} < \frac{1}{2}$ and for $(0, 0, n)$, we have $\pi(\mathrm{VS}^+_{(0,0,n)}) < \pi(\mathrm{VS}^+_{(0,0,n-1)}) < \frac{1}{2}$.

Therefore, the algorithm $\mathrm{GBS}(\pi, \{X_1, X_2\})$ will query $(1, 0, 0)$ at time 1. Suppose the label of $(1, 0, 0)$ is 0 since we consider the path corresponding to $h_{0,0,0}$ as the target hypothesis.

Now we show that $\mathrm{GBS}(\pi, \{X_1, X_2\})$ will query points $(0, 0, 1), (0, 0, 2), \ldots, (0, 0, n)$ sequentially by induction.

At time 1, the version space is $\mathrm{VS} = \{h_{0,0,0}\} \cup \{h_{i,j,l} \in \widehat{H} | i = 0\}$. We list $\pi(h_{0,j,l})$ for $j \in \{1, 2\}$ and $l \in [n]$ in Table 1 for illustration.

■ **Table 1** Table of $\pi(h_{0,j,l})$ for $j \in \{1, 2\}$ and $l \in [n]$.

	$(0, 1, 0)$	$(0, 2, 0)$
$(0, 0, 1)$	$\frac{1}{4 \cdot 3}$	$\frac{1}{4 \cdot 3}$
$(0, 0, 2)$	$\frac{1}{4 \cdot 3^2}$	$\frac{1}{4 \cdot 3^2}$
$\ldots$	$\ldots$	$\ldots$
$(0, 0, n)$	$\frac{1}{8 \cdot 3^{n-1}}$	$\frac{1}{8 \cdot 3^{n-1}}$

Then we can compute that

$$\pi(S^+_{(0,1,0)}) = \pi(S^+_{(0,2,0)}) = \frac{1}{4 \cdot 3} + \frac{1}{4 \cdot 3^2} + \ldots + \frac{1}{4 \cdot 3^{n-1}} + \frac{1}{8 \cdot 3^{n-1}} = \frac{1}{8},$$

$$\pi(S^+_{(0,0,1)}) = \frac{1}{6} > \pi(S^+_{(0,0,l)}),$$

$$\pi(S^+_{(0,0,l)}) \le \pi(S^+_{(0,0,2)}) = \frac{1}{18},$$

for all $l \ge 2$.

Thus, the algorithm $\mathrm{GBS}(\pi, \{X_1, X_2\})$ will choose $(0, 0, 1)$ at time 2.

Suppose that at time $t = 2, 3, \ldots, l$, $\mathrm{GBS}(\pi, \{X_1, X_2\})$ has picked $(0, 0, 1), \ldots, (0, 0, l-1)$ and all are labeled 0.

Now we show that $\mathrm{GBS}(\pi, \{X_1, X_2\})$ will pick $(0, 0, l)$ at time $t = l + 1$. The version space at the beginning of time $l + 1$ is $\mathrm{VS} = \{h_{0,0,0}\} \cup \{h_{i,j,p} \in \widehat{H} | p \ge l\}$. We can compute that

$$\pi(S^+_{(0,0,l)}) = \frac{1}{2 \cdot 3^l} > \pi(S^+_{(0,0,p)})$$

for all $p > l$, and that

$$\pi(S^+_{(0,1,0)}) = \pi(S^+_{(0,2,0)}) = \frac{1}{4 \cdot 3^l} + \frac{1}{4 \cdot 3^{l+1}} + \ldots + \frac{1}{4 \cdot 3^{n-1}} + \frac{1}{8 \cdot 3^{n-1}} = \frac{1}{8 \cdot 3^{l-1}}.$$

Hence, $\mathrm{GBS}(\pi, \{X_1, X_2\})$ will pick $(0, 0, l)$.

Therefore, we proved that when the target hypothesis is $h_{0,0,0}$, $\text{GBS}(\pi, \{X_1, X_2\})$ will query
$(1, 0, 0), (0, 0, 1), (0, 0, 2), \dots, (0, 0, n)$ sequentially.

Thus, we have that $Q_1(\text{GBS}, \pi, \{X_1, X_2\}, h_{0,0,0}) = n + 1$, and $Q_1(\text{GBS}, \pi, \{X_1, X_2\}) \geq \frac{n+1}{4}$ as $\pi(h_{0,0,0}) = \frac{1}{4}$.

Label complexity of agent 1 when she runs the (GBS) baseline individually

Now we show that $Q(\text{GBS}, \pi, \{X_1\}) = \mathcal{O}(1)$. Since X_1 does not contain $(1, 0, 0)$, both $h_{0,j,l}$ and $h_{1,j,l}$ label X_1 identically. Every effective hypothesis over X_1 can be written as $h_{*,j,l}$ with $\pi(h_{*,j,l}) = \pi(h_{0,j,l}) + \pi(h_{1,j,l})$, which is listed in Table 2.

🟨 **Table 2** Table of $\pi(h_{*,j,l})$ for $j \in \{1, 2\}$ and $l \in [n]$.

	$(0, 1, 0)$	$(0, 2, 0)$
$(0, 0, 1)$	$\frac{1}{4 \cdot 3} + \frac{1}{3}$	$\frac{1}{4 \cdot 3}$
$(0, 0, 2)$	$\frac{1}{4 \cdot 3^2} + \frac{1}{3^2}$	$\frac{1}{4 \cdot 3^2}$
$\cdots$	$\cdots$	$\cdots$
$(0, 0, n)$	$\frac{1}{8 \cdot 3^{n-1}} + \frac{1}{2 \cdot 3^{n-1}}$	$\frac{1}{8 \cdot 3^{n-1}}$

Notice that if we know that the label of $(0, 0, l)$ is positive for some l, then the version space has at most 2 effective hypotheses, $h_{*,1,l}$ and $h_{*,2,l}$. In this case, the algorithm needs at most 2 more queries.

At time $t = 1$, we have

$$\pi(S^+_{(0,0,1)}) = \frac{1}{4 \cdot 3} + \frac{1}{3} + \frac{1}{4 \cdot 3} = \frac{1}{2},$$

$$\pi(S^+_{(0,0,l)}) < \pi(S^+_{(0,0,1)}), \forall l \geq 2,$$

$$\pi(S^+_{(0,2,0)}) = \frac{1}{4 \cdot 3} + \frac{1}{4 \cdot 3^2} + \dots + \frac{1}{4 \cdot 3^{n-1}} + \frac{1}{8 \cdot 3^{n-1}} = \frac{1}{8},$$

$$\pi(S^+_{(0,1,0)}) = \pi(S^+_{(0,2,0)}) \cdot 5 = \frac{5}{8}.$$

Therefore, $\text{GBS}(\pi, \{X_1\})$ will query $(0, 0, 1)$ at $t = 1$.

We complete the proof by exhaustion. If $(0, 0, 1)$ is labeled as 1, then the algorithm needs at most two more queries as aforementioned.

If $(0, 0, 1)$ is labeled as 0, then $h_{*,1,1}$ and $h_{*,2,1}$ will be removed from the version space and $\text{GBS}(\pi, \{X_1\})$ will query $(0, 1, 0)$ at $t = 2$ then.

If the label is 1, the version space is reduced to $\{h_{*,1,l} | l = 2, \dots, n\}$ and $\text{GBS}(\pi, \{X_1\})$ will query $(0, 0, 2), (0, 0, 3), \dots$ sequentially until receiving a positive label.

If the label of $(0, 1, 0)$ is 0, $\text{GBS}(\pi, \{X_1\})$ will query $(0, 2, 0)$ at time $t = 3$. If the label of $(0, 2, 0)$ is 1, then it is similar to the case of $(0, 1, 0)$ being labeled 1 and the algorithm will query $(0, 0, 2), (0, 0, 3), \dots$ sequentially.

If the label of $(0, 2, 0)$ is 0, we know the target hypothesis is $h_{0,0,0}$ and we are done.

Hence we have $Q(\text{GBS}, \pi, \{X_1\}) \leq \sum_{l=1}^{n} (\pi(h_{*,1,l}) + \pi(h_{*,2,l})) \cdot (3 + l) + \pi(h_{0,0,0}) \cdot 3 = \sum_{l=1}^{n-1} \frac{1}{2 \cdot 3^{l-1}} \cdot (3 + l) + \frac{1}{4} \cdot 3 = \mathcal{O}(1)$. ◀

C   Proof of Lemma 10

Proof. SIR property: Since $\mathcal{A}'$ and $\{\mathcal{O}_i | i \in [k]\}$ are IR and agent i can strictly benefit from $\mathcal{O}_i$, we have $Q_i(\mathcal{A}''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) < Q(\mathcal{A}, \pi, \{X_i\})$ for all $i \in [k]$.

Label complexity: The label complexity of $\mathcal{A}''_\varepsilon$ is

$$
\begin{aligned}
Q(\mathcal{A}''_\varepsilon, \pi, \{X_1, \ldots, X_k\}) =& (1 - \frac{\varepsilon}{n}) Q(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) + \frac{\varepsilon}{kn} \sum_{i=1}^{k} Q(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}) \\
\leq& (1 - \frac{\varepsilon}{n}) Q(\mathcal{A}', \pi, \{X_1, \ldots, X_k\}) + \varepsilon \,.
\end{aligned}
$$

Then we are done.   ◀

D   Proof of necessity of Assumption 1

Proof of necessity. Suppose that there exists an SIR algorithm $\mathcal{A}''$ when the baseline algorithm is optimal. We therefore have $Q_i(\mathcal{A}'', \pi, \{X_1, \ldots, X_k\}) < Q^*(\pi, \{X_i\})$ by definition. We claim that $\mathcal{A}''$ must satisfy $Q_i(\mathcal{A}'', \pi, \{X_1, \ldots, X_k\}) \geq \mathbb{E}_{h \sim \pi} [Q^*(\pi_{h, -i}, \{X_i\})]$. This is because we can construct another algorithm $\mathcal{B}$ by running $\mathcal{A}''$ over all other agents except agent i, i.e., running $\mathcal{A}''$ over (π, X_{-i}) with $X_{-i} = \{X_j | j \neq i\}$ first to recover the labels of all other agents X_{-i}. Then, $\mathcal{B}$ simulates $\mathcal{A}''$ over $(\pi, \{X_1, \ldots, X_k\})$ without actually querying any point in X_{-i} (similarly to Algorithm 1). In this case, the label complexities of agent i are identical for algorithms $\mathcal{B}$ and $\mathcal{A}''$, i.e., $Q_i(\mathcal{B}, \pi, \{X_1, \ldots, X_k\}) = Q_i(\mathcal{A}'', \pi, \{X_1, \ldots, X_k\})$. Since $Q^*(\pi_{h, -i}, \{X_i\})$ is the optimal label complexity of agent i given the label information of X_{-i}, we have $Q_i(\mathcal{B}, \pi, \{X_1, \ldots, X_k\}) \geq Q^*(\pi_{h, -i}, \{X_i\})$. Therefore, we have $Q^*(\pi, \{X_i\}) > Q_i(\mathcal{A}'', \pi, \{X_1, \ldots, X_k\}) \geq Q^*(\pi_{h, -i}, \{X_i\})$.   ◀

E   Proof of Lemma 12

Proof. First, note that $\mathcal{O}_i$ is IR as B2IR$(\mathcal{A})$ is IR, and pruning the query tree does not increase label complexity. For any $i \in [k]$, suppose that there exists an hypothesis $h \in \widehat{H}$ s.t. $|H(X_i | h)| < |H(X_i)|$. Then in the query tree of $\mathcal{A}(\pi, \{X_i\})$, either all leaves are inconsistent with $h(X_{-i})$ or there exists one internal node v who has exactly one subtree with all leaves inconsistent with $h(X_{-i})$. This node v as well as the corresponding subtree are pruned in $\mathcal{O}_i$ and thus the leaves in the other subtree rooted at v have their depth reduced by at least 1. Now, there exists an hypothesis $h'' \in \widehat{H}$ such that $h''(X_i) \in H(X_i | h)$ and $h''(X_i)$ is in the other subtree. Since $h''(X_{-i}) = h(X_{-i})$, when the underlying hypothesis is h'', the pruned tree given $h''(X_{-i})$ is the same as that given $h(X_{-i})$. Hence, we have $Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}, h'') \leq Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}, h'') - 1$.

Then we have

$$
\begin{aligned}
& Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}) = \mathbb{E}_{h \sim \pi} [Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}, h)] \\
=& \pi(h'') Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}, h'') + (1 - \pi(h'')) \mathbb{E}_{h \sim \pi | h \neq h''} [Q_i(\mathcal{O}_i, \pi, \{X_1, \ldots, X_k\}, h)] \\
\leq& \pi(h'') (Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}, h'') - 1) + (1 - \pi(h'')) \mathbb{E}_{h \sim \pi | h \neq h''} [Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}, h)] \\
<& Q_i(\mathcal{A}, \pi, \{X_1, \ldots, X_k\}) \,,
\end{aligned}
$$

where the last inequality holds due to $\pi(h'') > 0$ since w.l.o.g., we assumed $\pi(h) > 0$ for all $h \in \widehat{H}$ in Section 2.   ◀