Generalized Binary Search For Split-Neighborly Problems

Stephen Mussmann
Stanford University

Percy Liang
Stanford University

Abstract

In sequential hypothesis testing, Generalized Binary Search (GBS) greedily chooses the test with the highest information gain at each step. It is known that GBS obtains the gold standard query cost of $O(\log n)$ for problems satisfying the k-neighborly condition, which requires any two tests to be connected by a sequence of tests where neighboring tests disagree on at most k hypotheses. In this paper, we introduce a weaker condition, split-neighborly, which requires that for the set of hypotheses two neighbors disagree on, any subset is splittable by some test. For four problems that are not k-neighborly for any constant k, we prove that they are split-neighborly, which allows us to obtain the optimal $O(\log n)$ worst-case query cost.

1 Introduction

Sequential hypothesis testing (Young & Young, 1998) aims to find the true hypothesis from a set of hypotheses by performing tests. Some examples are gathering observations to deduce the location of a hidden object or labeling data points to infer an underlying classifier. One commonly used algorithm is Generalized Binary Search (GBS), also known as the splitting algorithm, which greedily chooses the test that most evenly splits the hypothesis version space (Garey & Graham, 1974; Nowak, 2008), or equivalently greedily chooses the test with the maximal information gain (for binary tests). Greedy information gain is surprisingly effective in practice and has become the gold standard with a variety of applications, approximations, and extensions (Settles, 2012; Chu & Ghahramani, 2005; Bellala et al., 2010; Karbasi et al., 2012; Zheng et al., 2012; Jedynak et al., 2012; Luo et al., 2013; Maji et al., 2014; Sun et al., 2015). We seek to explain this performance by providing a condition under which GBS attains a query cost of $O(\log n)$, the information-theoretic optimal query cost.

While there has been much work on establishing that GBS attains an average cost within a $\log n$ factor of the optimal algorithm (Guillory & Bilmes, 2009; Kosaraju et al., 1999; Dasgupta, 2004; Chakaravarthy et al., 2007; Adler & Heeringa, 2008; Chakaravarthy et al., 2009; Gupta et al., 2010), establishing the asymptotically optimal query cost is a difficult and understudied problem. The few previous works have required somewhat stringent conditions. One such condition is "sample-rich" (Naghshvar et al., 2012), which states that every subset of the hypotheses has a test that returns true on exactly those hypotheses; this requires an exponential number of tests. Another line of work (Nowak, 2009, 2011) introduced the more lenient *k-neighborly* condition, which requires that every two tests be connected by a sequence of tests where neighboring tests disagree on at most k hypotheses. As we will show in this paper, many problems of a discrete nature do not satisfy this condition.

In this paper, building on the k-neighborly condition, we introduce a new, weaker condition called *$1/\alpha$-split-neighborly*, which requires that if neighboring tests disagree on a set of hypotheses V, then there exists a test that splits off an α fraction of V (note that $|V|$ could be quite large, whereas k-neighborly requires $|V| \leq k$). We prove that four natural problems satisfy the $1/\alpha$-split-neighborly condition for a constant α: pool-based linear classifiers, learning monotonic CNF formulas, discrete object localization, and discrete binary classification. Furthermore, we prove that the value of k in the k-neighborly analysis is at least $\sqrt{n}/2$ for all four problems, which yields nearly vacuous bounds. In summary, by using $1/\alpha$-split-neighborly, we show that Generalized Binary Search achieves an asymptotically optimal query cost of $O(\log n)$ in settings where the previous k-neighborly analysis tools fail.

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

Algorithm 1 Active querying algorithm template

Input: $\mathcal{H}$, $\mathcal{X}$, oracle access to h^*, method F
$V = \mathcal{H}$
for $t = 0, 1, \ldots$ **do**
  $x_t \leftarrow F(\{(x_i, y_i)\}_{i=1}^{t-1})$
  Query x_t and obtain $y_t = h^*(x_t)$
  Update $V \leftarrow \{h \in V : h(x_t) = y_t\}$
  if $|V| = 1$ **then**
    return $h \in V$
  end if
end for

Algorithm 2 Generalized Binary Search (F)

Input: $\mathcal{H}$, $\mathcal{X}$, Previous test results $\{(x_i, y_i)\}_{i=1}^{t-1}$
$V = \{h \in \mathcal{H} : h(x_i) = y_i, 1 \le i \le t-1\}$
$x_t \leftarrow \text{argmin}_{x \in \mathcal{X}} |\mathbb{E}_{h \in V}[h(x)] - 1/2|$
return x_t

1.1 Notation

In all cases, we use log to denote $\log_2$. For a set S and function f, we define $\mathbb{E}_{s \in S}[f(s)] = \frac{\sum_{s \in S} f(s)}{|S|}$. Similarly, for a condition C, we define $\Pr_{s \in S}[C(s)] = \frac{\sum_{s \in S} \mathbf{1}[C(s)]}{|S|}$.

2 Problem statement

Consider a set of n hypotheses $\mathcal{H}$ and tests $\mathcal{X}$, where each $h \in \mathcal{H}$ is a mapping from $\mathcal{X}$ to $\{0, 1\}$. We assume that the hypotheses $\mathcal{H}$ are identifiable, meaning that any two hypotheses yield different results on at least one test. Assume there is a fixed but unknown hypothesis $h^* \in \mathcal{H}$ that we wish to identify. An active querying algorithm performs a sequence of tests; on each iteration, it uses a method F to select a test x_t based on the results of previous tests and receives $y_t = h^*(x_t)$ (see Algorithm 1). We evaluate F based on the *worst-case* number of queries.

As an illustrative example, let the set of hypotheses $\mathcal{H}$ be linear classifiers separating 5 data points at the vertices of a regular pentagon. (see Figure 1). In this case, $|\mathcal{H}| = 20$ (only including hypotheses with both $+$ and $-$ labels). The tests $\mathcal{X}$ are data points, and the test output is an indicator for h^* classifying that point as $+$. Figure 1 shows the queries generated by GBS. We see that the size of the version space $|V|$ decreases exponentially, the hallmark of a $O(\log n)$ worst-case query cost. Later, we will prove that GBS indeed attains $O(\log n)$ worst-case query cost for this this problem class of linear classifiers on the vertices of convex polygons.

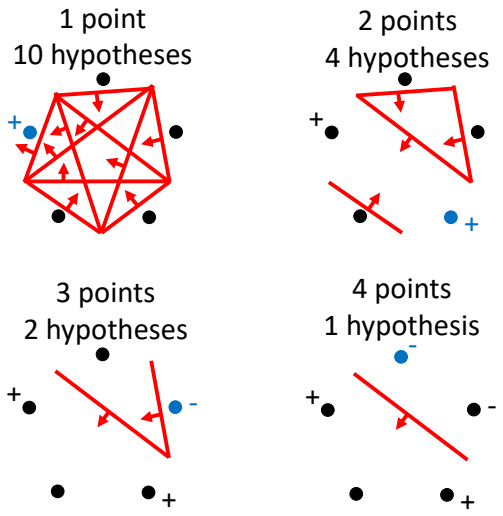

Figure 1: Problem of identifying a linear classifier in a pool-based active learning setting. Each linear classifier is represented by a red line with an arrow pointing towards the positive class. Each round, we select a new test (blue point), after four rounds, we have identified the true classifier.

Though we focus on the noiseless and well-specified setting, both conditions can be relaxed: Kääriäinen (2006) reduces the non-persistent noisy setting (we can repeatedly query any test) to the noiseless setting, and Nowak (2011) adapts the GBS algorithm to the misspecified setting (see Section 5 for more details).

Generalized Binary Search (also known as the "splitting algorithm" and "maximal shrinkage") is a well-studied method (Garey & Graham, 1974; Nowak, 2011; Dasgupta, 2004). GBS maintains the set $V \subseteq \mathcal{H}$ of hypotheses consistent with test results thus far, and at each step, it chooses a test that splits the elements of V as evenly as possible. See Algorithm 2 for the pseudocode. The optimal worst-case number of queries is $\Omega(\log n)$ so if GBS attains $O(\log n)$ for a problem, it is asymptotically optimal.

3 General analysis

3.1 Splits

Intuitively, GBS works well when it can find tests that split the hypothesis space into roughly equal parts. The *split* induced by test x is a partition of the hypotheses into $\{h \in \mathcal{H} : h(x) = 0\}$ and $\{h \in \mathcal{H} : h(x) = 1\}$. Define the *split constant* of a test x for a set of hypotheses V as $\min_{y \in \{0,1\}} \Pr_{h \in V}[h(x) = y]$, the fraction of hypotheses in the smaller partition. Note that large split constants are preferred, and $1/2$ is the maximum split constant. As we will see, both the k-

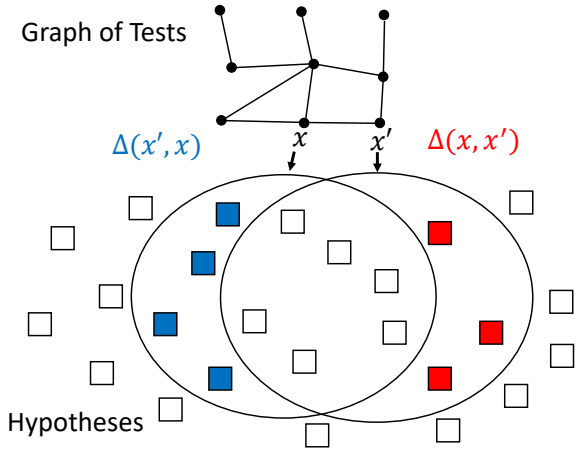

Figure 2: A test graph on the top and the action of two neighboring tests x, x' on the hypotheses on the bottom. The hypotheses h are represented by rectangles and the tests by circles that contain the hypotheses for which the test returns 1. For a k-neighborly edge to exist between two tests, the corresponding $\Delta(x, x')$ and $\Delta(x', x)$ must have cardinality $|\Delta(x, x') \cup \Delta(x', x)| \leq k$. If the resulting test graph is connected, we say the problem is k-neighborly.

neighborly condition (Nowak, 2011) and our new $1/\alpha$-split-neighborly condition imply that for any version space V, there is a test with a large split constant.

3.2 Earlier work: k-neighborly and coherence parameter

Tests often have a similarity structure. As an example, for the hypothesis class of linear classifiers, nearby input points (tests) yield the same result for most hypotheses. We therefore construct a similarity graph over tests, which will provide a useful analysis tool that allows us to only reason locally on the graph. Nowak (2011) defines two tests to be similar if they disagree on at most k hypotheses. k-neighborly is the condition that such a similarity graph is connected.

Definition 3.1 (k-neighborly)**.** *For any two tests, x and x', define $\Delta(x, x') = \{h \in \mathcal{H} : h(x) = 0 \wedge h(x') = 1\}$. Let the test graph contain undirected edges (x, x') for which $|\Delta(x, x') \cup \Delta(x', x)| \leq k$. A problem instance is k-neighborly if the test graph is connected.*

See Figure 2 for an illustration of the k-neighborly condition. Intuitively, the k-neighborly condition ensures that between any two tests, we can find a path where each pair of neighbors in the path are very similar. Nowak (2011) also defines the coherence parameter, which ensures an algorithm can easily find tests that both return 0 and 1 by choosing tests randomly.[1]

Definition 3.2 (Coherence parameter)**.** *The coherence parameter is the largest c such that*

$$\forall h \in \mathcal{H} : \mathbb{E}_{x \sim P}[h(x)] \in [c, 1 - c]$$

for some probability distribution P over tests.

This is a concept that will be used with our condition, $1/\alpha$-split-neighborly, as well. From these two definitions, Nowak (2011) showed the following result:

Theorem 3.1 (Nowak, 2011)**.** *If a problem has a coherence parameter c and is k-neighborly, then the worst-case cost of GBS is $\frac{1}{-\log(\lambda)} \log(n)$ queries, where $\lambda = 1 - \min(c, \frac{1}{k+2})$.*

For large enough c, the k-neighborly analysis yields worst-case query complexity of $O(k \log(n))$. Later, we show several examples where $k = \Omega(\sqrt{n})$, yielding the k-neighborly analysis very loose.

3.3 Split-neighborly

The k-neighborly condition is a rather strong condition since it requires tests that disagree on only k hypotheses. While this sometimes holds for problems with a continuous structure, such as linear classifiers or continuous object localization, it is often not satisfied for problems with a discrete nature. Later, in Section 4, we show a variety of problems of a discrete nature where k is at least $\sqrt{n}/2$. Motivated by these discrete problems, we will now introduce a weaker condition which we call $1/\alpha$-split-neighborly, the main contribution of this paper. In $1/\alpha$-split-neighborly, two tests are not only connected if there is a small number of hypotheses on which the tests differ, but also if any subset of the hypotheses that they differ on can be split evenly (with a split constant of at least α) by some test.

Definition 3.3 ($1/\alpha$-split-neighborly)**.** *Let $\alpha \in (0, \frac{1}{2}]$* [2]*. For any two tests, x and x', define $\Delta(x, x') = \{h \in \mathcal{H} : h(x) = 0 \wedge h(x') = 1\}$. Define a directed test graph to have a directed edge (x, x') if for any $V \subseteq \Delta(x, x')$, $|V| \leq 1$ or there exists a test $x \in \mathcal{X}$ such that*

$$\mathbb{E}_{h \in V}[h(x)] \in [\alpha, 1 - \alpha]$$

A problem is $1/\alpha$-split-neighborly if the test graph is strongly connected.

[1] Our definition is a simple linear transformation of the definition in Nowak (2011) to account for notational differences.

[2] As a special case, we say a problem is 1-split-neighborly if the graph generated by connecting nodes where $|\Delta(x, x')| \leq 1$ is strongly connected.

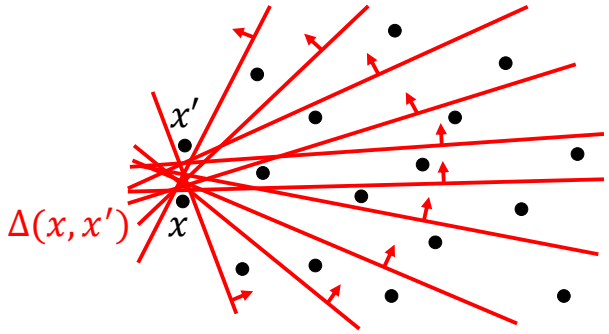

Figure 3: For the problem of identifying linear classifiers in a pool-based active learning setting, an example of two tests that are not connected in the k-neighborly graph for small k but are connected in the $1/\alpha$-split-neighborly graph for small $1/\alpha$. While the size of $\Delta(x, x')$ is large, we can still split any subset of $\Delta(x, x')$ because of the other points in the pool.

Although it can be more involved to show an edge between tests in the sense of $1/\alpha$-split-neighborly rather than k-neighborly, it is a more general condition which makes the similarity graph more connected (see Figure 3 for an example).

The coherence, k-neighborly, and $1/\alpha$-split-neighborly conditions are preserved when we restrict the hypotheses: create a problem with same tests $\mathcal{X}' = \mathcal{X}$ but with $\mathcal{H}' \subseteq \mathcal{H}$. This is because the conditions all are statements involving a universal quantification over the hypotheses, or subsets thereof.

Furthermore, and most importantly, constant coherence and the split-neighborly condition imply that GBS has $O(\log n)$ query cost. First, we prove the following lemma, showing that constant coherence and $1/\alpha$-split-neighborly imply that any subset of $\mathcal{H}$ has a test with a good split constant.

Lemma 3.1. *If a problem is $1/\alpha$-split-neighborly and has a coherence parameter of c, then for any $V \subseteq \mathcal{H}$, $|V| \leq 1$ or there exists a test $x \in \mathcal{X}$ such that*

$$\mathbb{E}_{h \in V}[h(x)] \in [\beta, 1 - \beta]$$

where the split constant is

$$\beta = \min\left(c, \frac{1}{1/\alpha + 2}\right).$$

Note that for large c and small α, $\beta \approx \alpha$. See appendix for the full proof; we only give a sketch here. Intuitively, the coherence parameter ensures that there is a good split of V or there is both a test that mostly yields 0 and a test that mostly yields 1 (for hypotheses V). If we examine a path of tests x between the two tests, either $\mathbb{E}_{h \in V}[h(x)]$ varies smoothly from close to 0 to close to 1, in which case there is a good split, or there is a large jump in the split constant between two neighboring tests x and x', which implies that $|V \cap \Delta(x, x')|/|V|$ is large. Finally, from the definition of $1/\alpha$-split-neighborly, we can find a test to have a β split constant of $V \cap \Delta(x, x')$. In summary, the split-neighborly condition and coherence condition allow us to conclude that for any subset of the hypotheses, there is a test with a β split constant.

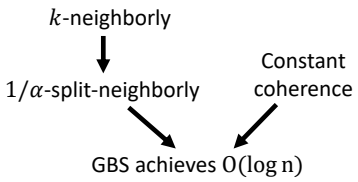

Figure 4: Relationship between the different conditions, where arrows represent logical implication.

From this lemma, we get the following theorem.

Theorem 3.2. *If a problem is $1/\alpha$-split-neighborly and has a coherence parameter of c, then GBS has a worst case query cost of at most $\frac{\log n}{-\log(1-\beta)}$, where*

$$\beta = \min\left(c, \frac{1}{1/\alpha + 2}\right).$$

From Lemma 3.1, it is clear that after m queries, we have at most $n(1-\beta)^m$ hypotheses left. Thus, the worst-case query cost (to reach one hypothesis) is $\frac{\log n}{-\log(1-\beta)}$. The precise proof is in the appendix.

Similarly to the k-neighborly condition, for large enough coherence, the worst-case query cost is $O(\frac{1}{\alpha}\log n)$. Thus, for constant α, we get $O(\log n)$ worst-case query cost, but for $\alpha \to 0$, we do not.

In fact, k-neighborly implies k-split-neighborly ($1/\alpha$-split-neighborly, $\alpha = 1/k$). Thus, our split-neighborly condition is a generalization of k-neighborly, and comparison between our theorems shows our condition is strictly more powerful than the k-neighborly condition. See Figure 4 for a diagram.

Proposition 3.1. *If a problem is k-neighborly, then it is k-split-neighborly.*

Proof. In the case that $k = 1$, $|\Delta(x, x')| = 1$ so $|V| \leq 1$ so the problem is 1-split-neighborly. Note that any set of hypotheses must have a test that distinguishes at least one of the hypotheses (otherwise the hypotheses are the same). If two points x and x' in the k-neighborly graph have an edge between them, then $|\Delta(x, x') \cup \Delta(x', x)| \leq k$, which implies $|V| \leq |\Delta(x, x')| \leq k$, and thus either $|V| \leq 1$ or there

is a test with a $1/k$ split constant and thus there is an edge from x to x' in the k-split-neighborly graph. Similarly, there is an edge from x' to x, the k-split-neighborly graph is strongly connected, and the problem is k-split-neighborly. □

4 Application of analysis

In this work, we establish the $1/\alpha$-split-neighborly condition for four problems: two-dimensional linear classifiers on the vertices of a convex polygon, learning monotonic disjunctions and CNF formulas, discrete object localization (under two different conditions), and discrete linear classifiers. We show that GBS achieves $O(\log n)$ cost on these problems under conditions on $\mathcal{H}$ by showing that the problems are $1/\alpha$-split-neighborly and have constant coherence. Further, we show the inadequacy of the k-neighborly analysis for each of these problems.

All of the proofs have a similar structure for proving $1/\alpha$-split-neighborly. First, fix a subset of hypotheses $V \subseteq \Delta(x, x')$. Then, by assuming there is no test with a good split constant α, we can leverage the structure of the problems to conclude that the size of V is small. Since any two hypotheses disagree on at least one test (identifiability of hypotheses), we can always split off one of the hypotheses for a split constant of $1/|V|$ which is a good split if $|V| \leq 1/\alpha$.

Since there is no test with a split constant α, any test either yields 1 on the vast majority of hypotheses in V or yields 0 on the vast majority of hypotheses in V. Thus, we partition the tests into two sets,

$$\mathcal{X}^+ = \{x \in \mathcal{X} : \Pr_{h \in V}[h(x) = 1] > 1 - \alpha\}$$

$$\mathcal{X}^- = \{x \in \mathcal{X} : \Pr_{h \in V}[h(x) = 1] < \alpha\} = \mathcal{X} - \mathcal{X}^+$$

Several of the arguments will leverage the structure of this partition and use union bound to show that the probability of a single hypothesis is high, and thus V is small.

4.1 Two-dimensional linear classifiers with convex polygon data pool

Suppose we have a pool of unlabeled data points and our set of hypotheses is linear classifiers in the transductive setting (we group all hypotheses with the same output on all unlabeled data points together). We examine the case of two dimensions.

Problem 1 (Linear classifiers on convex polygon data pool)**.** *Let $\mathcal{X}$ be a set of m points $x \in \mathbb{R}^2$ such that the points are the vertices of a convex polygon. Let $\mathcal{H}$ be equivalence classes of linear classifiers that have the same output on $\mathcal{X}$ and such that*

$$\frac{\sum_{x \in \mathcal{X}} h(x)}{|\mathcal{X}|} \in \left[\frac{1}{4}, \frac{3}{4}\right].$$

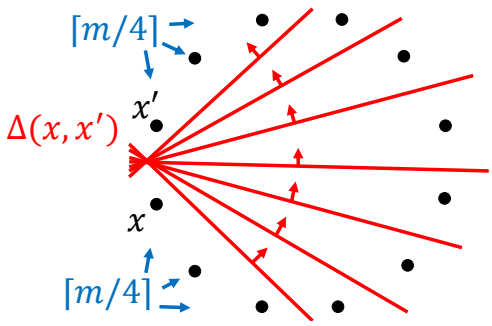

Figure 5: An illustration of $\Delta(x, x')$ for linear classifiers on a data pool forming the vertices of a convex polygon. Note that we can split any subset of $\Delta(x, x')$ with a split constant of at least $1/3$ because the tests are interleaved in a sequence with the hypotheses.

This last constraint restricts the classifiers to those with balanced labels. This ensures a good coherence parameter; otherwise, no algorithm can perform better than $\Theta(m) = \Theta(\sqrt{n})$.

We will now show that the k-neighborly analysis for this problem is poor. See Figure 5 for a diagram. For adjacent points x and x', $|\Delta(x, x')| = m - 2\lceil m/4 \rceil + 1$. Further note that $n = |\mathcal{H}| = m(m - 2\lceil m/4 \rceil + 1)$ [3]. Thus, $|\Delta(x, x')| \geq \frac{\sqrt{n}}{2}$ (for $m \geq 4$) and so the k for the k-neighborly analysis is at least $\frac{\sqrt{n}}{2}$.

However, it is clear from Figure 5 that $\Delta(x, x')$ is a sequence of hypotheses with tests interleaved. Thus, we can split $\Delta(x, x')$ with at least a split constant of $1/3$ and to get the following proposition,

Proposition 4.1. *The problem of learning a linear classifier on a convex polygon data pool is 3-split-neighborly.*

Note that because of the constraint that the minority label is at least $1/4$, the coherence parameter is at least $c = 1/4$. Thus, the worst case query complexity is at most $\frac{\log n}{-\log(1-1/5)} \leq 3.2 \log n$.

4.2 Monotonic CNF formulas

In this section, we examine the problem of learning monotonic CNF formulas from function evaluations. To begin, we study the case of a single disjunction, such as the following,

$$x_4 \vee x_7 \vee x_9$$

[3] We have 2 linear classifiers for each line, but we are double counting.

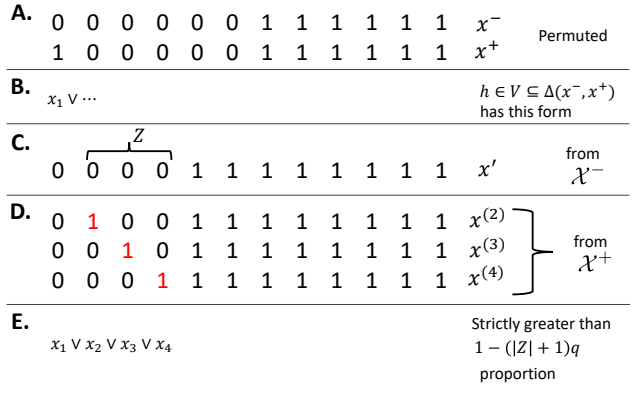

Figure 6: A proof illustration for the disjunction problem being split-neighborly.

Problem 2 (Disjunction)**.** *Let the elements of $\mathcal{H}$ be a disjunction over d variables without any negations where the disjunction has at most m variables. Let $\mathcal{X}$ be the set of length d bit assignments.*

First, note that $h(0^d) = 0$ and $h(1^d) = 1$ for all $h \in \mathcal{H}$ and thus the coherence parameter is $c = 1/2$. The k-neighborly analysis is lacking for this problem. Note, $|\mathcal{H}| = \sum_{i=1}^{m} \binom{d}{i}$. However, the bit string $0^d \in \mathcal{X}$ disagrees with all other $x \in \mathcal{X}$ for $\sum_{i=1}^{m} \binom{d-1}{i-1}$ hypotheses. So for $m \geq 2, d \geq 2m$, then $k \geq \sqrt{n}$. See the appendix for details. On the other hand, our split neighborly analysis achieves the optimal rate in the case where m is constant and d goes to infinity.

Theorem 4.1. *The single disjunction problem is $(m+1)$-split-neighborly.*

Proof. A graphic for the proof is shown in Figure 6.

We will show that there are edges between tests that differ by just one bit. This will suffice since such a graph is strongly connected. In particular, we show that the test graph has a bidirectional edge from x to x' if $||x - x'||_1 = 1$.

Let x^+ be the value of x or x' with more 1's (and let x^- be the other one). Note that from monotonicity, $|\Delta(x^+, x^-)| = 0$ so there is a directed edge from x^+ to x^-.

For the other direction, fix a subset $V \subseteq \Delta(x^-, x^+)$. Without loss of generality, let x^+ and x^- differ in the first coordinate so $x_1^+ = 1$ and $x_1^- = 0$ and $\forall i > 1 : x_i^+ = x_i^-$. See row A of Figure 6. Because $V \subseteq \Delta(x^-, x^+)$, all hypotheses in V include x_1 in the disjunction. See row B of Figure 6.

For ease of notation, let $q = 1/(m+1)$. We will proceed by showing that if there is no test with a good split, $\mathbb{E}_{h \in V}[h(x)] \in [q, 1-q]$, then $|V| \leq m+1$. Then, either $|V| \leq 1$ or there is a test with split constant at least $1/(m+1)$ and the proof is complete.

Now, if there are no tests with a good split, each test must either yield 1 or 0 for the vast majority of hypotheses in V. Thus, we can define the following two sets.

$$\mathcal{X}^+ = \{x \in \mathcal{X} : \Pr_{h \in V}[h(x) = 1] > 1 - q\}$$

$$\mathcal{X}^- = \{x \in \mathcal{X} : \Pr_{h \in V}[h(x) = 1] < q\} = \mathcal{X} - \mathcal{X}^+$$

Let x' be the the element of $\mathcal{X}^-$ with the fewest 0's. Since $x' \in \mathcal{X}^-$, $x'_1 = 0$. Let Z be the other indices of the 0's. If $|Z| = 0$, then $|\Delta(x^-, x^+)| = |\{x_1\}| = 1$ so $|V| \leq 1$ and we are done. Define $\{x^{(j)}\}_{j \in Z}$ to be the test resulting from x' and changing the j^{th} bit to a 1. By the minimal definition of x', $\forall j \in Z : x^{(j)} \in \mathcal{X}^+$. See rows C and D of Figure 6.

We now derive a useful equation. Note that for any subset $Z' \subseteq Z$, from the definition of $\mathcal{X}^+$ and $\mathcal{X}^-$ and union bound, $\Pr_{h \in V}[h(x') = 0 \wedge \forall j \in Z' : h(x^{(j)}) = 1] > 1 - (|Z'| + 1)q$. From the property of disjunctions, this implies $\Pr_{h \in V}[h \text{ has variables at } Z' \cup \{1\}] > 1 - (|Z'| + 1)q$. See row E of Figure 6.

If $|Z| \geq m$, then this means that we can choose a subset Z' of size m. $\Pr_{h \in V}[h \text{ includes } m+1 \text{ variables}] > 1 - (m+1)q = 0$. This means there is a non-zero probability of a hypothesis with $m+1$ variables which is impossible, since our disjunctions don't have more than m variables. So $|Z| \leq m - 1$.

We are nearly done. Note that the left side of the useful equation is exactly $1/|V|$. Therefore, $1/|V| > 1 - (|Z| + 1)q \geq 1 - mq \geq 1/(m+1)$. Rearranging, we find that $|V| < m + 1$ and we are done. □

We now examine the more general monotonic CNF problem from function evaluations. An example of such a monotonic formula is:

$$(x_1 \vee x_4 \vee x_5) \wedge (x_2 \vee x_7 \vee x_8).$$

Problem 3 (Conjunction of disjunctions)**.** *Let $\mathcal{H}$ be a conjunction of ℓ m-disjunctions over d variables without any negations, and where each variable does not appear in multiple disjunctions. Let $\mathcal{X}$ be the set of length d bit assignments.*

Note that there is an isomorphism between conjunctions of disjunctions and disjunctions of conjunctions by flipping the test bits and the result bit.

Additionally, for a general setting shown in the appendix, $k \geq \sqrt{n}$ which renders the k-neighborly analysis very poor.

However, the split-neighborly analysis suffices,

Theorem 4.2. *The conjunction of disjunctions problem is* $(m + 1 + 3(l-1))$*-split-neighborly.*

The proof is in the appendix. Note that the value of $1/\alpha$ does not depend on the number of variables, so GBS is efficient even for very large d when m, l are constant.

4.3 Object localization in $\mathbb{Z}^d$

Consider the problem of object localization (Chen et al., 2015) where we try to locate an object based on spatial queries. In this work, we wish to find the location z of an object in space or in an image by asking queries of the form "Is z close to point x?". We can discretize the space into the grid of integers and define "closeness" by as whether $z - x$ is in some set S (e.g., an ℓ_p ball).

In this way, hypotheses and tests are both indexed by vectors of integers. For concreteness, for an ℓ_p norm ball, a test x returns the result of $\|x - z\|_p \leq \ell$.

Problem 4 (Object localization)**.** *Fix a set* $S \subseteq \mathbb{Z}^d$ *representing the sensing field*

$$\mathcal{H} \subseteq \{h_z\}_{z \in \mathbb{Z}^d} \quad \mathcal{X} = \mathbb{Z}^d \quad h_z(x) = \mathbf{1}[x - z \in S]$$

Note that there are infinitely many integer vectors, but we can take a bounded region as the hypothesis space. Note that if the bounded region of the hypothesis space is too large, the coherence parameter would be very small, and greedy would not perform well (and any algorithm for that matter, since the algorithm would have to do a linear search). One way to make the coherence parameter $c = 1/2$ is to choose a x^* and ensure that $\mathcal{H} \subseteq \{h_z : z - x^* \in S\}$.

For cases where S is an axis-symmetric box, or equivalently where we use a weighted ℓ_∞ norm, the problem is split-neighborly.

Theorem 4.3. *The object localization problem where* S *is an axis-symmetric box is* 4*-split-neighborly.*

The proof is in the appendix. See Figure 7 for some intuition. Note that the value of k for this problem is the largest cross-sectional volume of the box, which for $d \geq 2$, $k \geq \sqrt{n}$.

For axis-symmetric, axis-convex (weaker than convex) sets S, we have a dimension-dependent bound. By an axis-convex set S, we mean that if two points in S differ in only one dimension, then all integral points between them are also in S.

Theorem 4.4. *If* S *is a bounded, axis-symmetric, axis-convex set, the object localization problem is* $(4d + 1)$*-split-neighborly.*

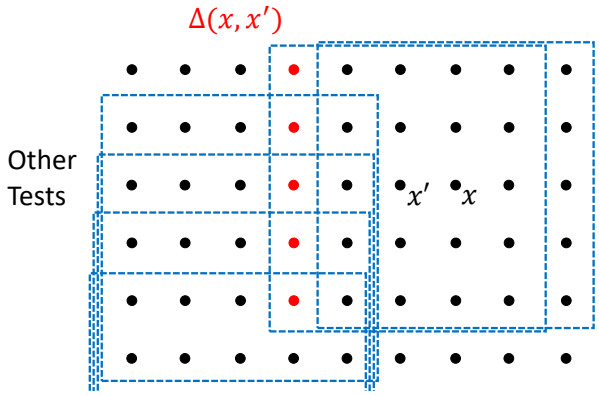

Figure 7: For the object localization problem where S is an axis-symmetric box, the $1/\alpha$-split-neighborly graph has edges between adjacent points, for example x and x' in this figure. Thus, $\Delta(x, x')$ will be a flat box. This figure shows that for $d = 2$, the problem is 3-split-neighborly. In fact, the problem is 4-split-neighborly for all d.

The proof is in the appendix and uses union bound with the partition of $\mathcal{X}$ into $\mathcal{X}^-$ and $\mathcal{X}^+$. For this problem, k must be at least the largest, axis-aligned "shadow" which for $d \geq 2$, $k \geq \sqrt{n}$.

4.4 Discrete binary linear classifier

Linear classifiers are a classic type of function where the output label is $y = \mathbf{1}[w \cdot x + b > 0]$. We consider the active learning setting where $\mathcal{H}$ is a set of (w, b) pairs and $\mathcal{X}$ are points x in the feature space. In fact, this is the problem covered previously by Nowak (2011) where w and x take continuous values.

Here, we consider the setting of discrete linear classifiers, where $x \in \{0, 1\}^d$ and $w \in \{-1, 0, 1\}^d$.

Problem 5 (Discrete Binary Linear Classifier)**.**

$$\mathcal{H} \subseteq \{h_{b,w}\}_{b \in \mathbb{Z}, w \in \{-1,0,1\}^d}$$

$$\mathcal{X} = \{0, 1\}^d$$

$$h_{b,w}(x) = \mathbf{1}[w \cdot x > b]$$

with the following holding for all hypotheses ($w^{(+)}$ *and* $w^{(-)}$ *are the number of positive and negative vector components for* w*, respectively):*

$$w^{(+)} - b \leq r(w^{(-)} + b) - \frac{d}{8}$$

$$w^{(-)} + b \leq r(w^{(+)} - b - 1) - \frac{d}{8}$$

Intuitively, $w^{(+)} - b$ is the maximum "overshoot" of the threshold and $w^{(-)} + b$ is the maximum "undershoot".

We require that the ratio between these quantities is at most r with the addition of an additive constant.

Theorem 4.5. *The discrete binary linear classifier problem is* $\max(16, 8r)$*-split-neighborly.*

The coherence parameter will be constant c if there is some data distribution such that each hypothesis yields a balanced label distribution (at least probability c of minority label). Intuitively, this is necessary since if the labels are very unbalanced, it may take many queries to even find a minority label. With this condition, GBS achieves $O(\log n)$ on the discrete binary linear classifiers problem.

On the other hand, the k-neighborly analysis does not work here, as before. In the special case where d is divisible by 4, $b = d/4 - 1$ and there are an equal number of 1 and 0 weights (just for making the calculation simpler), for $d \geq 4$, $k \geq \sqrt{n}$. See the Appendix for more details.

5 Discussion and related work

The GBS algorithm, or more generally, choosing the test that maximizes the information gain, has several approximations and variants. The greedy information gain technique was introduced in MacKay (1992) and used or extended in active learning (Jedynak et al., 2012), ranking learning (Chu & Ghahramani, 2005), comparison based search (Karbasi et al., 2012), image segmentation (Maji et al., 2014), structured prediction (Sun et al., 2015; Luo et al., 2013), group identification (Bellala et al., 2010), and graphical models (Zheng et al., 2012). As the active learning survey of Settles (2012) notes, "all of the general query frameworks we have looked at contain a popular utility function that can be viewed as an approximation to [information gain] under certain conditions." Thus, greedy information gain is seen as the gold standard, and there has been significant work finding approximations and extensions. Our work examines the other side of GBS and tries to understand that gap between GBS and the optimal solution.

A large body of literature exists on the analysis of GBS and close relatives in the noiseless and well-specified version of the sequential hypothesis testing problem, known as the optimal decision tree problem (Guillory & Bilmes, 2009; Kosaraju et al., 1999; Dasgupta, 2004; Chakaravarthy et al., 2007; Adler & Heeringa, 2008; Chakaravarthy et al., 2009; Gupta et al., 2010). These analyses, which borrow ideas from submodular analysis, yield an *average* cost *ratio* of $O(\log n)$, where the average cost ratio for a method is defined as the ratio between the expected cost of the method and the expected optimal cost (note that this is significantly worse than an average query cost of $O(\log n)$). Furthermore, there exists a problem where GBS achieves a average cost ratio of $\Theta(\log n / \log\log n)$ (optimal is $\Theta(\log n)$ but GBS is $\Theta(\log^2 n / \log\log n)$), so the general upper bound for GBS is very close to tight (Dasgupta, 2004). In our work, we show that GBS achieves a *constant factor* cost ratio, that is, within a constant factor of the optimal cost. In many natural settings such as linear classification, the hypothesis space is exponentially large in the dimension (i.e. $n = 2^{O(d)}$), so existing guarantees are $O(d)$ times the optimal, which itself is $O(d)$ for many problems. In our work, we prove in multiple settings that GBS achieves the asymptotically optimal query cost.

Other works extend the noiseless and well-specified assumptions to more general frameworks. Nowak (2011) provides a way to adapt GBS to the mis-specified case with only a constant factor increase in the query complexity that ensures GBS never performs worse than randomly querying tests (the naive approach). Kääriäinen (2006) provides a reduction from the noisy case to the noiseless case. There are two different noise settings which are handled separately in the literature, persistent noise (tests are not repeatable) and non-persistent noise (tests are repeatable). Earlier work (Nowak, 2009, 2011; Naghshvar et al., 2012) that has handled noise has addressed *i.i.d.* noise with repeatable tests, where the outputs of the deterministic problem are flipped with a constant probability p. In the case of non-persistent *i.i.d.* noise, Kääriäinen (2006) presents a technique to reduce the noisy case to the deterministic case by repeatedly querying tests and using the majority vote, so that with high probability we attain the uncorrupted test result. Thus, while our work might appear to only handle the noiseless case, it actually handles the non-persistent noise case as well.

Theoretical explanations for the effectiveness of GBS are still incomplete. Although GBS always achieves a cost ratio of $O(\log n)$ (Guillory & Bilmes, 2009), in the large hypothesis space regime, this factor could be very large. Furthermore, there do exist problems for which GBS performs much worse than the optimal (Dasgupta, 2004). These examples, however, tend to be contrived. Anecdotally, from the sample problems in this paper, we found that GBS is effective for most "natural" problems. In conclusion, we have made progress on characterizing this observation by introducing the $1/\alpha$-split-neighborly condition, which provably ensures that GBS achieves the asymptotically optimal query cost.

Acknowledgments

This research was supported by NSF grant DGE-1656518.

References

Adler, Micah and Heeringa, Brent. Approximating optimal binary decision trees. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pp. 1–9. Springer, 2008.

Bellala, Gowtham, Bhavnani, Suresh, and Scott, Clayton. Extensions of generalized binary search to group identification and exponential costs. In *Advances in Neural Information Processing Systems*, pp. 154–162, 2010.

Chakaravarthy, Venkatesan T, Pandit, Vinayaka, Roy, Sambuddha, Awasthi, Pranjal, and Mohania, Mukesh. 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*, pp. 53–62. ACM, 2007.

Chakaravarthy, Venkatesan T, Pandit, Vinayaka, Roy, Sambuddha, and Sabharwal, Yogish. Approximating decision trees with multiway branches. In *International Colloquium on Automata, Languages, and Programming*, pp. 210–221. Springer, 2009.

Chen, Yuxin, Javdani, Shervin, Karbasi, Amin, Bagnell, J Andrew, Srinivasa, Siddhartha S, and Krause, Andreas. Submodular surrogates for value of information. In *AAAI*, pp. 3511–3518, 2015.

Chu, Wei and Ghahramani, Zoubin. Extensions of gaussian processes for ranking: semisupervised and active learning. *Learning to Rank*, pp. 29, 2005.

Dasgupta, Sanjoy. Analysis of a greedy active learning strategy. In *Advances in neural information processing systems*, pp. 337–344, 2004.

Garey, Michael R and Graham, Ronald L. Performance bounds on the splitting algorithm for binary testing. *Acta Informatica*, 3(4):347–355, 1974.

Guillory, Andrew and Bilmes, Jeff. Average-case active learning with costs. In *International Conference on Algorithmic Learning Theory*, pp. 141–155. Springer, 2009.

Gupta, Anupam, Nagarajan, Viswanath, and Ravi, R. Approximation algorithms for optimal decision trees and adaptive tsp problems. In *International Colloquium on Automata, Languages, and Programming*, pp. 690–701. Springer, 2010.

Jedynak, Bruno, Frazier, Peter I, and Sznitman, Raphael. Twenty questions with noise: Bayes optimal policies for entropy loss. *Journal of Applied Probability*, 49(01):114–136, 2012.

Kääriäinen, Matti. Active learning in the nonrealizable case. In *International Conference on Algorithmic Learning Theory*, pp. 63–77. Springer, 2006.

Karbasi, Amin, Ioannidis, Stratis, et al. Comparison-based learning with rank nets. *arXiv preprint arXiv:1206.4674*, 2012.

Kosaraju, S Rao, Przytycka, Teresa M, and Borgstrom, Ryan. On an optimal split tree problem. In *Workshop on Algorithms and Data Structures*, pp. 157–168. Springer, 1999.

Luo, Wenjie, Schwing, Alex, and Urtasun, Raquel. Latent structured active learning. In *Advances in Neural Information Processing Systems*, pp. 728–736, 2013.

MacKay, David JC. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.

Maji, Subhransu, Hazan, Tamir, and Jaakkola, Tommi S. Active boundary annotation using random map perturbations. In *AISTATS*, pp. 604–613, 2014.

Naghshvar, Mohammad, Javidi, Tara, and Chaudhuri, Kamalika. Noisy bayesian active learning. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pp. 1626–1633. IEEE, 2012.

Nowak, Robert. Generalized binary search. In *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*, pp. 568–574. IEEE, 2008.

Nowak, Robert. Noisy generalized binary search. In *Advances in neural information processing systems*, pp. 1366–1374, 2009.

Nowak, Robert D. The geometry of generalized binary search. *IEEE Transactions on Information Theory*, 57(12):7893–7906, 2011.

Settles, Burr. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1): 1–114, 2012.

Sun, Qing, Laddha, Ankit, and Batra, Dhruv. Active learning for structured probabilistic models with histogram approximation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3612–3621, 2015.

Young, Linda J and Young, Jerry H. Sequential hypothesis testing. In *Statistical Ecology*, pp. 153–190. Springer, 1998.

Zheng, Alice X, Rish, Irina, and Beygelzimer, Alina. Efficient test selection in active diagnosis via entropy approximation. *arXiv preprint arXiv:1207.1418*, 2012.