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 & Ohahramani, 2005; Bellalta et al., 2010; Karbasi et al., 2012; Zheng et al., 2012). 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 $\alpha$ 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 $\alpha$: 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.
2 hypotheses
Similarly, for a condition \( C \) and function \( t \), we define \( \Pr \{ \argmin_{x \in X} |E_{h \in V}|h(x)| - 1/2| \}
if \( |V| = 1 \) then
return \( h \in V \)
end if
end for

Algorithm 2 Generalized Binary Search (F)
Input: \( H, X \), Previous test results \( \{(x_i, y_i)\}_{i=1}^{t-1} \),
\( V = \{ h \in H : h(x_i) = y_i, 1 \leq i \leq t-1 \} \),
\( x_t \leftarrow \arg\min_{x \in X} |E_{h \in V}|h(x)| - 1/2| \)
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
\( \Pr_{s \in S} [C(s)] = \frac{\sum_{s \in S} 1[C(s)]}{|S|} \).

2 Problem statement
Consider a set of \( n \) hypotheses \( H \) and tests \( X \), where each \( h \in H \) is a mapping from \( X \) to \( \{0, 1\} \). We assume that the hypotheses \( H \) are identifiable, meaning that any two hypotheses yield different results on at least one test. Assume there is a fixed but unknown hypothesis \( h^* \in H \) 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 \( H \) be linear classifiers separating 5 data points at the vertices of a regular pentagon. (see Figure 1). In this case, \(|H| = 20 \) (only including hypotheses with both + and - labels). The tests \( X \) 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.

Though we focus on the noiseless and well-specified setting, both conditions can be relaxed: Kaariainen (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 \( V \subseteq H \) 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 partitioning of the hypotheses into \( \{h \in H : h(x) = 0\} \) and \( \{h \in H : h(x) = 1\} \). 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 \( k-\)}
neighborly condition \cite{Nowak2011} 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. \cite{Nowak2011} 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 $\Delta(x, x') = \{ h \in H : h(x) = 0 \land h(x') = 1 \}$. Let the test graph contain undirected edges $(x, x')$ for which $|\Delta(x, x')| \leq k$. 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. \cite{Nowak2011} also defines the coherence parameter, which ensures an algorithm can easily find tests that both return 0 and 1 by choosing tests randomly.

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

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

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, \cite{Nowak2011} 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 $\frac{1}{\log(n)} \log(n)$ queries, where $\lambda = 1 - \min(c, \frac{1}{k+1})$.

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 $\alpha$) by some test.

**Definition 3.3** ($1/\alpha$-split-neighborly). Let $\alpha \in (0, \frac{1}{2}]$. \footnote{Our definition is a simple linear transformation of the definition in \cite{Nowak2011} to account for notational differences.} For any two tests, $x$ and $x'$, define $\Delta(x, x') = \{ h \in H : h(x) = 0 \land h(x') = 1 \}$. 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.

\footnote{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.}
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 $X' = X$ 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 $x \in X$ 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 $\alpha$, $\beta \approx \alpha$. 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 $\beta$ 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 $\beta$ split constant.

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 $\log n \over \log(1 - \beta)$, 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 $\log n \over \log(1 - \beta)$. The precise proof is in the appendix.

Similarly to the $k$-neighborly condition, for large enough coherence, the worst-case query cost is $O(\log n)$. Thus, for constant $\alpha$, 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

![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.](image)

![Figure 4: Relationship between the different conditions, where arrows represent logical implication.](image)
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.

\[ \Delta \]

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 $\alpha$, 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 $\alpha$, 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

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

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 $x'$, $|\Delta(x, x')| = m - 2\lfloor m/4 \rfloor + 1$. Further note that $n = |H| = m(m - 2\lfloor m/4 \rfloor + 1)$.

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,

\[ x_4 \lor x_7 \lor x_9 \]

\[ ^3 \text{We have 2 linear classifiers for each line, but we are double counting.} \]
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, $|\mathcal{H}| = \sum_{m=1}^{m} \binom{d}{2}$. However, the bit string $0^d \in \mathcal{X}$ disagrees with all other $x \in \mathcal{X}$ for $\sum_{m=1}^{m} \binom{d-1}{2}$ hypotheses. So for $m \geq 2, d \geq 2m$, then $k \geq \sqrt{m}$. 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 $x^+ = 1$ and $x^- = 0$ 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, $E_{h \in V}[h(x)] \in [q, 1-q]$, 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, $\Pr_{h \in V}[h(x') = 0 \land \forall j \in Z' : h(x^{(j)}) = 1] \geq 1 - (|Z'| + 1)q$. From the property of disjunctions, this implies $\Pr_{h \in V}[h \text{ has variables at } Z' \cup \{1\}] \geq 1 - (|Z'| + 1)q$. See row E of Figure 6.

If $|Z| \geq m$, then this means that we can choose a subset $Z'$ of size $m$. $\Pr_{h \in V}[h \text{ includes } m+1 \text{ variables}] \geq 1 - (m+1)q = 0$. 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, $1/|V| > 1 - (|Z| + 1)q \geq 1 - m + q \geq 1/(m+1)$. 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:

$$(x_1 \lor x_4 \lor x_5) \land (x_2 \lor x_7 \lor x_8).$$

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, $k \geq \sqrt{m}$ 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))-\text{split-neighborly}.

The proof is in the appendix. Note that the value of \(1/a\) 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 \cite{Chen2015} 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 \(\ell_p\) ball).

In this way, hypotheses and tests are both indexed by vectors of integers. For concreteness, for an \(\ell_2\) norm ball, a test \(x\) returns the result of \(||x-z||_2 \leq \ell\).

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

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

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 \(\ell_\infty\) 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 \(d\).](image)

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 \(y = 1[w \cdot x + b > 0]\). 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 \cite{Nowak2011} 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).

\[
\mathcal{H} \subseteq \{h_{b,w}\}_{b \in \mathbb{Z}, w \in \{-1,0,1\}^d} \\
\mathcal{X} = \{0,1\}^d \\
h_{b,w}(x) = 1[w \cdot x > b]
\]

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”.

Stephen Mussmann, Percy Liang
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 (Karbası 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 naïve approach). Kaariainen (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, Kaariainen (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.

Kiiäriäinen, Matti. Active learning in the non-realizable 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. In Advances in neural information processing systems, pp. 1366–1374, 2009.

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.
6 Appendix

6.1 Split-neighborly proofs

**Theorem 3.2** If a problem is $1/\alpha$-split-neighborly and has a coherence parameter of $c$, for

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

GBS has a worst case query cost of at most $\frac{\log n}{\log (1/\beta)}$ and GBS has an average query cost of at most $\frac{\log n}{H(\beta)}$ where $H(p)$ is the entropy of a Bernoulli($p$) random variable.

**Proof.** This theorem will follow from the next three lemmas.

**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 $\beta$ is defined as above.

**Proof.** Fix a subset $V \subseteq \mathcal{H}$. Assume $|V| > 1$, otherwise we are done.

From the assumption, we have a coherence parameter of

$$c \geq \beta$$

From the definition, this means that there exists a probability distribution on the tests $P$ such that for any hypothesis $h$,

$$\sum_{x \in \mathcal{X}} P(x)h(x) \in [\beta, 1 - \beta]$$

Since this is true for all $h \in \mathcal{H}$, this is also true for all convex combinations. Thus,

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

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

For simplicity, define the split constant $S(x) = \mathbb{E}_{h \in V}[h(x)]$. Thus,

$$\sum_{x \in \mathcal{X}} P(x)S(x) \in [\beta, 1 - \beta]$$

There are two possibilities, either there exists a test $x$ such that

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

in which case, this is the exact conclusion statement and we are done, or that there exists no test with a split constant in $[\beta, 1 - \beta]$. If there exists no test with a split constant in $[\beta, 1 - \beta]$ but the weighted combination is in $[\beta, 1 - \beta]$, then there exists tests $x$ and $x'$ such that $S(x) < \beta$ and $S(x') > 1 - \beta$

Since the problem is $1/\alpha$-split neighborly, there exists a graph over the tests that is strongly connected. Thus, there is a path from $x$ to $x'$. Since $S(x) < \beta$ and $S(x') > 1 - \beta$ and since $\forall x'' \in \mathcal{X} : S(x'') \notin [\beta, 1 - \beta]$, there must be an edge $(x_0, x_1)$ along the path where $S(x_0) < \beta$ and $S(x_1) > 1 - \beta$. Thus,

$$\mathbb{P}_{h \in V}[h(x_0) = 1] = \mathbb{E}_{h \in V}[h(x_0)] < \beta$$

$$\mathbb{P}_{h \in V}[h(x_1) = 1] = \mathbb{E}_{h \in V}[h(x_1)] > 1 - \beta$$

Combining these two yields,

$$\mathbb{P}_{h \in V}[h(x_0) = 0 \land h(x_1) = 1] > 1 - 2\beta$$

Recall $\Delta(x_0, x_1) = \{h \in \mathcal{H} : h(x_0) = 0, h(x_1) = 1\}$

$$\mathbb{P}_{h \in V}[h \in \Delta(x_0, x_1)] > 1 - 2\beta$$

Recall from the definition of $\beta$ that $\frac{1}{1/\alpha + 2} \geq \beta$. Thus

$$1 - 2\beta \geq 1 - 2 \frac{1}{1/\alpha + 2} = \frac{1/\alpha}{1/\alpha + 2} \geq \frac{\beta}{\alpha}$$

Thus,

$$\frac{|V \cap \Delta(x_0, x_1)|}{|V|} > \frac{\beta}{\alpha}$$

For brevity, define $\Delta = \Delta(x_0, x_1)$. Since there is an edge $(x_0, x_1)$ in the $1/\alpha$-neighborly graph, for any subset including $V \cap \Delta \subseteq \Delta$, either $|V \cap \Delta| \leq 1$ or there exists a test $\tilde{x}$ such that,
First, \( |V \cap \Delta| \neq 0 \), since \( |V| > 1 \) and \( \frac{|V \cap \Delta(x_0,x_1)|}{|V|} > \frac{\beta}{\alpha} \).

If \( |V \cap \Delta| = 1 \), then, \( \frac{|V \cap \Delta(x_0,x_1)|}{|V|} > \frac{\beta}{\alpha} \) and \( |V| > 1 \) so \( \frac{1}{2} \geq \frac{1}{|V|} > \frac{\beta}{\alpha} \geq \beta \). Since the hypotheses are identifiable, any pair of hypotheses yield a different result on some test, so we can always find a test with a split constant of at least \( \frac{1}{|V|} \), and this implies the result of the theorem.

In the other case, where \( |V \cap \Delta| > 1 \), we have all the necessary pieces and it’s just a matter of crunching the algebra.

\[
\mathbb{E}_{h \in V}[h(\hat{x})] = \frac{\sum_{h \in V} h(\hat{x})}{|V|} \\
\geq \frac{\beta}{\alpha} \frac{\sum_{h \in V \cap \Delta} h(\hat{x})}{|V \cap \Delta|} \\
\geq \frac{\beta}{\alpha} \mathbb{E}_{h \in V \cap \Delta}[h(\hat{x})] \\
\geq \frac{\beta}{\alpha} \mathbb{E}_{h \in V}[h(\hat{x})] \\
\geq \frac{\beta}{\alpha} \mathbb{E}_{h \in V}[h(\hat{x})] = \beta
\]

Additionally,

\[
\mathbb{E}_{h \in V}[h(\hat{x})] = \frac{\sum_{h \in V} h(\hat{x})}{|V|} \\
= \frac{\sum_{h \in V \cap \Delta} h(\hat{x}) + \sum_{h \in V \setminus \Delta} h(\hat{x})}{|V|} \\
\leq \frac{(1-\alpha)|V \cap \Delta| + \sum_{h \in V \setminus \Delta} h(\hat{x})}{|V|} \\
\leq \frac{(1-\alpha)|V \cap \Delta| + |V| - |V \cap \Delta|}{|V|} \\
\leq 1 - \alpha \frac{|V \cap \Delta|}{|V|} \\
\leq 1 - \alpha \frac{\beta}{\alpha} = 1 - \beta
\]

Thus, we have that

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

which is the conclusion of the lemma.

**Lemma 6.1.** If, for any \( V \subseteq \mathcal{H} \), \( |V| \leq 1 \) or there exists a test \( x \in \mathcal{X} \) such that

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

then GBS has a worst case query cost of at most \( \frac{\log n}{\log \log n} \).

**Proof.** After \( m \) queries, there are at most max(1, (1 − \( \beta \))\( mn \)) remaining hypotheses since greedy will choose a test with a split constant of at least \( \beta \) (a split with respect to the hypotheses without a prior) and will terminate when there is a single hypothesis. Thus, when \( (1 - \beta)^m n \leq 1 \), the algorithm must have terminated. Rearranging, we see that when \( m \geq \frac{\log n}{\log \log n} \), the algorithm must have terminated. This means that the worst case query cost must be at most \( \frac{\log n}{\log \log n} \).

**Lemma 6.2.** If, for any \( V \subseteq \mathcal{H} \), \( |V| \leq 1 \) or there exists a test \( x \in \mathcal{X} \) such that

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

then GBS has an average query cost of at most \( \frac{\log n}{H(\beta)} \) where \( H(p) \) is the entropy of a Bernoulli(p) random variable.

**Proof.** Define \( H(p) \) as the entropy of a Bernoulli random variable with probability \( p \).

\[
f(V) = \mathbb{E}[\text{average queries remaining while at subset } V]
\]

We will prove by induction on increasing subsets that

\[
f(V) \leq \frac{\log (|V|)}{H(\beta)}
\]

Note that the base case is that \( f(\{h\}) = 0 \) because we are done when there is just one hypothesis left. Note that this suffices to show that the total runtime is \( \log(n)/H(\beta) \) because \( |V| = n \) at the beginning of the algorithm.

Let \( A, B \) be a partition of \( V \) based on a test split. Without loss of generality, let \( |A| \leq |B| \), so \( |A| \leq \frac{1}{2}|V| \). Based on the recursive definition of cost and there is a test with a split constant of at least \( \beta \) (so GBS will choose a test with a split constant of at least \( \beta \),

\[
f(V) \leq \max_{A,B,|A|/|V| \in [\beta, 1/2]} \frac{|A|}{|V|} f(A) + \frac{|B|}{|V|} f(B) + 1
\]
From the induction hypothesis,
\[
\leq \max_{\beta} \frac{|A| \log |A| + |B| \log |B|}{H(\beta)} + H(\beta) + 1
\]
\[
\leq \frac{\max_{\beta} |A| \log |A| + |B| \log |B| + H(\beta)}{H(\beta)}
\]
\[
\leq \max_{\beta} -H(\frac{|A|}{|B|}) + H(\beta) + \log |V|
\]
\[
\leq \frac{-H(\frac{|A|}{|B|}) + H(\beta) + \log |V|}{H(\beta)}
\]
Note that since $|A|/|V| \in [\beta, 1/2]$ (the condition of the max), $H(\frac{|A|}{|B|}) \geq H(\beta)$. Thus, the max is non-positive, and thus,
\[
f(V) \leq \frac{\log(|V|)}{H(\beta)}
\]
Thus, we have proved the statement by induction and this suffices to show that the total runtime is at most $\log(n)/H(\beta)$.

\[\square\]

**Proposition 3.1** If a problem is $k$-neighborly and has a uniform prior, then the problem 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. Assume $k > 1$. 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 $|\Delta(x, x')| \leq k$, and thus either $|\Delta(x, x')| \leq 1$ 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. By a similar argument, there is also an edge from $x'$ to $x$. Since the $k$-neighborly graph is connected and each edge corresponds to a bidirectional edge in the $k$-split-neighborly graph, the $k$-split-neighborly graph is strongly connected and thus the problem is $k$-split-neighborly.

\[\square\]

6.2 Value of $k$

6.2.1 Disjunctions

For the disjunctions problem, for $m \geq 2, d \geq 2m$,
\[
n = \sum_{i=1}^{m} \binom{d}{i}
\]
\[
k \geq \sum_{i=1}^{m} \binom{d-1}{i-1}
\]
\[
k \geq 1 + \sum_{i=1}^{m-1} \binom{d-1}{i}
\]
\[
k^2 - n \geq 1 + 2 \sum_{i=1}^{m-1} \binom{d-1}{i} + \left( \sum_{i=1}^{m-1} \binom{d-1}{i} \right)^2
\]
\[
- \sum_{i=1}^{m-1} \binom{d}{i} - \binom{d}{m}
\]
Note that $2^{\binom{d-1}{i}} \geq \binom{d}{i}$ since $i \leq m - 1 \leq d/2$.

\[
k^2 - n \geq 1 + \left( \sum_{i=1}^{m-1} \binom{d-1}{i} \right)^2 - \binom{d}{m}
\]
\[
\geq \binom{d}{m} - \binom{d}{m}
\]
\[
\geq \binom{d-1}{m-1} \left( \binom{d-1}{m-1} - d/m \right)
\]
Since $m \geq 2$,
\[
\geq \binom{d-1}{m-1} - d/2
\]
\[
\geq d/2 - 1
\]
\[
\geq m - 1
\]
\[
\geq 0
\]
Thus, $k^2 - n \geq 0$ and so $k \geq \sqrt{n}$.

6.2.2 Monotonic CNF

Note that $n = |H| = \frac{1}{m!} \binom{d}{m,m,\ldots,m,d-lm}$. All of the bit strings with strictly less than $l$ ones will be trivially connected in the $k$-neighborly graph, because they yield 0 on all hypotheses. However, the closest test to connect them to the rest of the graph is the bit string $1^l0^{d-l} \in X$, which disagrees on $\binom{d-1}{m-1,\ldots,m-1,d-im} \leq k$ hypotheses. We examine the case where $d \geq 2ml$ and $m \geq 2$.

For the monotonic CNF formulas, recall that
\[
n = |H| = \frac{1}{m!} \binom{d}{m,m,\ldots,m,d-im}
\]
For simplicity, define $c$.

Thus, $d$.

Recall that we are in the special case where $d$.

\[ n = \frac{d}{m} \left( m, m, ..., m, d \right) \]

Since $d = 4m \geq 4l$,

\[ n \leq k \left( \frac{2l}{m} \right)^l \]

Since $d - l \geq 2l(m - 1)$ and $m \geq 2$

\[ n \leq k \left( \frac{d - l}{l(m - 1)} \right) \]

Thus, for $d \geq 4$, 

\[ n \leq k^2 \]

\[ k \geq \sqrt{n} \]

6.2.3 Discrete Linear Classifier

Recall that we are 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 $(d/2)$.

All tests with fewer than $d/4$ 1’s will yield a result of 0 for all hypotheses. The test with the next fewest hypotheses that yield 1 will be a test with exactly $d/4$ 1’s. Thus, $k$ is at least the number of such hypotheses that yield 1.

\[ n = \left( \frac{d}{d/2} \right) \]

\[ k \geq \left( \frac{3d/4}{d/4} \right) \]

For simplicity, define $c = d/4$.

\[ \frac{n}{k^2} \leq \frac{e^3}{\pi c^3} \leq 1 \]

\[ k \geq \sqrt{n} \]

6.3 Necessity of Dependencies

6.3.1 Linear classifiers on convex polygon data pool

For arbitrary data points where the points are not the vertices of a convex polygon, the linear classifier problem is not 1/4-split-neighborly for constant $a$. A counter-example is shown in Figure 8.

6.3.2 Disjunctions

The linear dependence on $m$ for the disjunctions is necessary because of the case where $d = m + 1$, and $|H| = d$ (each $h \in H$ lacking one variable). In this case, there are no tests with split constants of $1/m$, so the problem cannot be better than $(m-2)$-split-neighborly (recall coherence $c = 1/2$).
Generalized Binary Search For Split-Neighborly Problems

Figure 8: A counterexample that shows a non-convex data pool need not be split-neighborly. Note that we can at most split off 1 of the \( n \) hypotheses by querying one of the points from the lower half. However, the problem has coherence close to 1/2 and thus it cannot be \( \frac{1}{\alpha} \)-split-neighborly for constant \( \alpha \).

6.3.3 Monotonic CNF

For the monotonic CNF problem, the linear dependence on \( m \) is necessary because of the case where \( l = 1, d = m + 1 \), and \( |\mathcal{H}| = d \) (each \( h \in \mathcal{H} \) lacking one variable). In this case, there are no tests with split constants of \( \frac{1}{m} \), so the problem cannot be better than \((m - 2)\)-split-neighbourly (recall coherence \( c = 1/2 \)). Furthermore, the linear dependence on \( l \) is necessary because of the problem where \( m = 1, d = l + 1 \), and \( |\mathcal{H}| = d \) (each \( h \in \mathcal{H} \) lacking one variable). For this problem, there are no tests with split constants of \( \frac{l}{l+1} \), so the problem cannot be better than \((l - 2)\)-split-neighbourly. Thus, although the linear dependence on \( m \) and \( l \) is necessary, it may be possible to improve the constants.

6.3.4 Object Localization

For object localization with the axis-symmetric, axis-convex set \( S \), the dependence on \( d \) is necessary because if we use the set \( S = \{je_i : |j| \leq l, 1 \leq i \leq d \} \) and consider the set of hypotheses, \( \{\pm le_i : 1 \leq i \leq d \} \), the problem has no test with split constant of \( \frac{l}{d-l+1} \) but has coherence \( c = 1/2 \), so it can’t be \((2d-3)\)-split-neighbourly.

6.4 Monotonic CNF

Theorem 4.2 The Conjunction of Disjunctions problem is \((m + 1 + 3(l - 1))\)-split-neighborly.

Proof. We prove this theorem by induction on \( l \). First, for the base case \( l = 1 \).

The test graph has an edge from \( x \) to \( x' \) if \( ||x - x'||_1 = 1 \) (the bit strings differ in one location).

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

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 \( \forall i > 1 : x^+_i = x^-_i \).

For a proof by contradiction, the problem is not \((m + 1)\)-split-neighbourly so that \( |V| > 1 \) and there is no test \( x \) such that \( \mathbb{E}_{h \in V} [h(x)] \in [q, 1 - q] \), where \( q = 1/(m + 1) \).

Let

\[ X^+ = \{ x \in \mathcal{X} : \mathbb{P}_{h \in V} [h(x) = 1] > 1 - q \}, \]

\[ X^- = \{ x \in \mathcal{X} : \mathbb{P}_{h \in V} [h(x) = 1] < q \} = \mathcal{X} - X^+. \]

Let \( x' \) be the element of \( X^- \) with the fewest 0’s and let the 0’s be at indices \( Z \) (note 1 \( \in Z \)). If \( |Z| < m \), then \( h(x') = 1 \) for all \( h \) since the disjunctions have \( m \) variables. But since \( x' \in X^- \), which is a contradiction.

Define \( \{x^{(j)}\}_{j \in Z} \) to be the test resulting changing the \( j^{th} \) bit of \( x' \) to a 1. By the minimal definition of \( x' \), \( \forall j \in Z : x^{(j)} \in X^+ \).

Suppose \( |Z| > m \). Take a subset \( Z' \subseteq Z \) such that \( |Z'| = m + 1 \). Then, from the definition of \( X^+ \) and \( X^- \), \( \mathbb{P}_{h \in V} [h(x') = 0 \land \exists j \in Z' : h(x^{(j)}) = 1] > 1 - (m + 1)q \geq 0 \), which means \( \mathbb{P}_{h \in V} [h \text{ includes variables } Z' \}] > 0 \). Therefore, there is a disjunction with at least \( m + 1 \) variables, which is a contradiction.

Thus, \( |Z| = m \), so there is only one hypothesis such that \( h(x') = 0 \), the hypothesis with variables at \( Z \). So \( 1/|V| > 1 - q \) (by definition of \( X^- \)), which implies \( |V| = 1 \) since \( q \leq 1/2 \), which is a contradiction. Thus, by contradiction, the problem with \( l = 1 \) is \((m + 1)\)-split-neighbourly. For \( l > 1 \), we proceed by induction.

We can define the graph as above, define \( X^- \) and \( X^+ \) as above, and \( x' \) and \( Z \) as above. The same argument goes through that \( |Z| = m \). Thus, \((1 - q)\) proportion of the hypotheses have a disjunction with variables at the indices \( Z \). These hypotheses are simply another copy of the problem with \( l - 1 \) conjuctions and \( d - m \) variables. Since that problem has \( 1/2 \) coherence and is \( m + 1 + 3(l - 2)\)-splittable (by induction hypothesis), there exists some test with a split constant of \( \frac{1}{m+1+3(l-2)+2} \) for a total split constant on the original problem of

\[ \frac{1}{m+1+3(l-2)+2} = \frac{1}{m+1+3(l-1)} \]
Thus, the problem is \( m + 1 + 3(l - 1) \)-split-neighborly by induction.

6.5 Box Object Localization

**Theorem 4.3.** The object localization problem where \( S \) is a box is 4-split-neighborly.

Notationally, refer to \( z_h \) as the integer vector for the hypothesis \( h \) and \( z_{h,i} \) to be its \( i \)th component.

We begin by fixing two tests \( x \) and \( x' \) such that \( ||x - x'||_1 = 1 \). Without loss of generality, let \( x' - x = e_1 \) where \( e_1 \) is the 1st elementary vector. Since the box is axis symmetric, there exists radii \( r_i \geq 0 \) such that \( x - z_h \in S \Leftrightarrow \forall i : |x_i - z_{h,i}| \leq r_i \). Without loss of generality, assume \( x = (r_1, 0, 0, \ldots, 0) \) and \( x' = (r_1 + 1, 0, 0, \ldots, 0) \). Recall \( \Delta(x, x') = \{ h : h(x) = 0 \land h(x') = 1 \} \), this implies that \( \Delta(x, x') = \{ h : z_{h,1} = 0 \land \forall i > 1 : |z_{h,i}| \leq r_i \} \). We will begin by fixing a subset \( V \subseteq \Delta(x, x') \). As in all the application proofs, we will start by assuming by contradiction that there is no test with a split constant in the range \( [q, 1 - q] \) where \( q = 1/4 \).

We will use this contradiction to show that the size of \( V \) is small, so that there is in fact a test with a split constant \( q \) which is a contradiction.

6.5.1 Majority Element

Fix a dimension \( i \). Examine the tests \( X_i = \{ je_i : j = 0, \ldots, 2r_i + 1 \} \) and note that for \( h \in V \subseteq \Delta(x, x') \), \( h(je_i) = 1 \) if \( z_{h,i} \geq j - r_i \).

By the contradiction assumption,

\[
\mathbb{E}_{h \in V} [h(je_i)] \notin [q, 1 - q]
\]

\[
\Pr_{h \in V} [z_{h,i} \geq j - r_i] \notin [q, 1 - q]
\]

Since \( \Pr_{h \in V} [z_{h,i} \geq -r_i] = 1 \) and \( \Pr_{h \in V} [z_{h,i} \geq r_i + 1] = 0 \), there must be some integer \( m_i \) such that

\[
\Pr_{h \in V} [z_{h,i} \geq m_i] > 1 - q
\]

\[
\Pr_{h \in V} [z_{h,i} \geq m_i + 1] < q
\]

which implies that

\[
\Pr_{h \in V} [z_{h,i} = m_i] > 1 - 2q
\]

Define thus, there exists a vector \( m \) such that there is a \( 1 - 2q \) probability that an hypothesis’ \( i \)th component matches \( m \).

6.5.2 Side Splits

Intuitively, we will create a sequence of tests that each remove at least half of the elements with the \( i \)th component not equal to \( m \). For each test in the sequence, the probability that the test yields 1 over the hypotheses in \( V \) must be greater that \( 1 - q \) so we can prove that there aren’t many elements that disagree with \( m \) at any component.

Here we recursively define sets \( S_i, B_i, \) and \( A_i \). \( S_i \) will be defined in terms of \( B_i \) and \( B_i \) will be defined in terms of \( S_{i-1} \).

Define \( S_0 = V \) and for \( i > 1 \), \( S_i = S_{i-1} - B_i \). Noting that we could reflect the \( i \)th component about \( m_i \), without loss of generality, suppose that

\[
\Pr_{h \in S_i} [z_{h,i} > m_i] \geq \Pr_{h \in S_i} [z_{h,i} < m_i]
\]

Define \( B_i = \{ h \in S_{i-1} : z_{h,i} > m_i \} \) and \( A_i = \{ h \in S_{i-1} : z_{h,i} < m_i \} \)

Note that \( |B_i| \geq |A_i| \).

Further, there is a test \( x^{(i)} = (-r_1, \ldots, -r_i, 0, \ldots, 0) \) such that \( h(x^{(i)}) = 1 \leftrightarrow h \in S_i \) and thus by the contradiction assumption,

\[
\frac{|S_i|}{|S|} \notin [q, 1 - q]
\]

However, since \( \Pr_{h \in V} [z_{h,i} = m_i] > 1 - 2q \), \( |B_i|/|V| < 2q \). We now prove by induction that \( |S_i|/|V| > 1 - q \).

The base case is that \( |S_1|/|V| = 1 > 1 - q \). As long as \( q \leq 1/4 \), since \( |S_{i-1}|/|V| > 1 - q \) and \( |B_i|/|V| < 2q \), \( |S_i|/|S| > 1 - 3q \geq q \) (since \( q = 1/4 \)) and thus by the contradiction assumption \( |S_i|/|S| > 1 - q \).

Note that the \( B_i \) are disjoint because

\[
B_i \subseteq S_i = V - B_1 - B_2 - \ldots - B_{i-1}
\]

\[
|S_d| > (1 - q)|V|
\]

\[
|V - \bigcup_{i=1}^{d} B_i| > (1 - q)|V|
\]

\[
|V| - \sum_{i=1}^{d} |B_i| > (1 - q)|V|
\]

\[
q|V| > \sum_{i=1}^{d} |B_i|
\]
Define the set of elements $M' \subseteq V$ as the points with a component not equal to $m$. This is the union of all $A_i$ and $B_i$,

$$|M'| = \left| \bigcup_{i=1}^{d} A_i \cup \bigcup_{i=1}^{d} B_i \right| \leq \sum_{i=1}^{d} |A_i| + \sum_{i=1}^{d} |B_i| \leq 2 \sum_{i=1}^{d} |B_i| < 2q|V|$$

Also note that $|M'| \geq |V| - 1$ since there can only be one element that doesn’t disagree with any element of $m$. Thus,

$$|V| - 1 < 2q|V| \quad \Rightarrow \quad |V| < \frac{1}{1 - 2q}$$

Since $q \leq 1/3$, then this implies $|V| < 3$ so there is a test with a split of 1/3, which is a contradiction. So in a proof by contradiction, the problem is 4-split-neighborly.

### 6.6 Convex, axis-symmetric Shape Object Localization

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

**Proof.** Let the test graph has an edge from $x$ to $x'$ if $|x - x'| = 1$.

Fix a subset $V \subseteq \Delta(x, x')$. Without loss of generality, let $x' = 0^d$. $V \subseteq \Delta(x, x') \subseteq \{h : h(x') = 1\} = \{h : z_h - x' \in S\} = \{h : z_h \in S\}$

For a proof by contradiction, the problem is not 4d+1-split-neighborly so that $|V| > 1$ and there is no test $x$ such that $E_{h \in V}[h(x)] \in [q, 1 - q]$, where $q = 1/(4d + 1)$. Let

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

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

Note that $x' = 0^d \in X^+$ since $V \subseteq \{h : h(x') = 1\}$.

Fix a dimension $i$. Examine the set of tests $\{je_i : j \in Z\}$. From above, $0e_i \in X^+$. Further, since $V \subseteq \{h : z_h \in S\}$ and since $S$ is bounded, there exists some $B \in \mathbb{Z}$ such that $\pm B e_i \in X^-$. Thus there exists some $c_1 \leq 0, c_2 \geq 0$ such that $(c_1 - 1)e_i \in X^-, c_1 e_i \in X^+, c_2 e_i \in X^+, (c_2 + 1)e_i \in X^-$. From the definition of $X^+$ and $X^-$,

$$\Pr_{h \in V}[h((c_1 - 1)e_i)] = 0, h(c_1 e_i) = 1, ...$$

$$h(c_2 e_i) = 1, h((c_2 + 1)e_i) = 0 > 1 - 4q$$

Define $S_l = \{s_{-i} : s_i = l, s \in S\}$ to be the slices of $S$ along axis $i$ at location $l$. Therefore, $h(je_i) = 1 \iff z_{h,-i} \in S_{z_{h,-i} - j}$.

Note that $S_{-i} = S_i$ since the shape $S$ is axis symmetric. Combining these three facts,

$$\Pr_{h \in V}[z_{h,-i} \in S_{z_{h,-i} - 0} \cap S_{z_{h,-i} - 1}] \ldots$$

$$\\setminus \{ S_{z_{h,-i} - 0} \cup S_{z_{h,-i} - 1} \} > 1 - 4q$$

Note that for $l' > l \geq 0$, $S_{l,l'} \subseteq S_{l,l}$ because of axis-convexity. To see this, suppose there was $t \in S_{l,l'} \setminus S_{l,l}$, then there would be three elements $s^{(-1)}, s^{(0)}, s^{(1)}$ such that $s^{(-1)} \equiv t$ and $s^{(1)} \equiv l'$, which would imply $s^{(-1)} \not\in S^{(0)} \not\in S^{(1)} \in S$ which contradicts axis-convexity.

Thus, in order for the set composed of slices of $S$ in the equation above to be non-empty,

$$|z_{h,i} - c_1|, |z_{h,i} - c_2| < |z_{h,i} - c_1 + 1|, |z_{h,i} - c_2 - 1|$$

it must be the case that $z_{h,i} = \frac{c_1 + c_2}{2} \in \mathbb{Z}$ which we define to be $m_i$. So,

$$\Pr_{h \in V}[z_{h,i} = m_i] > 1 - 4q$$

Repeating this argument for all dimensions and combining,

$$\Pr_{h \in V}[\forall i : z_{h,i} = m_i] > 1 - 4dq$$

There is only one such element $z_h = m$ so

$$\frac{1}{|V|} > 1 - 4dq = 1 - 4d \frac{1}{4d + 1} = \frac{1}{4d + 1}$$

So $|V| < 4d + 1$ so there must be a split of at least $q$ which is a contradiction. 

\[\Box\]
6.7 Discrete Binary Linear Classifiers

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

Define \( q = \min\left(\frac{1}{16}, \frac{1}{8}\right) \)

Recall that for the Discrete Binary Linear Classifier case, we have hypotheses as a pair of vectors and threshold \( h = (w_h, b_h) \in \{-1, 0, 1\}^d \times \mathbb{Z} \) and tests as vectors \( \{0, 1\}^d \). Recall \( h(x) = 1[w_h \cdot x > b_h] \).

From the problem setting of Discrete Binary Linear Classifiers, we know that,

\[
\begin{align*}
  w_h^{(+) - b} & \leq r(w_h^{(-)} + b) - \frac{d}{8} \\
  w_h^{(-)} + b & \leq r(w_h^{(+)} - b - 1) - \frac{d}{8}
\end{align*}
\]

Recall \( w_h^{(+)} \) is the number of positive elements of \( w \) and \( w_h^{(-)} \) is the number of negative elements. Notationally \( w_{h,i} \) refers to the \( i \)th component of \( w_h \).

6.7.1 Key Lemma and its Sufficiency

We will first state a lemma and then prove that it implies the problem stated stated.

**Lemma 6.3.** Define

\[
\begin{align*}
  x^{(0)} &= (0, 0, ..., 0) \\
  x^{(1)} &= (1, 0, ..., 0) \\
  H' &= \{h \in H : h(x^{(0)}) = 0 \land h(x^{(1)}) = 1 \land ... \land w_h^{(+)} \leq r(w_h^{(-)} - \frac{d}{8} \land ...) \\
  &\quad \ldots \land w_h^{(-)} \leq r(w_h^{(+)} - 1) - \frac{d}{8}\}
\end{align*}
\]

For any subset \( V \subset H' \), there exists a test \( x \) such that \( \mathbb{E}_{h \in V}[h(x)] \in [q, 1 - q] \)

6.7.2 Proof of Theorem 4.5 from Lemma 6.3

We will prove Theorem 4.5 by a reduction to Lemma 6.3. To show that the problem is \( \frac{1}{\alpha} \)-split-neighborly, we need to show that for two tests with \( x \) and \( x' \) with \( |x - x'|_1 = 1 \) that for any subset \( V \subseteq \Delta(x, x') = \{h : h(x) = 0 \land h(x') = 1\} \), that \( |V| \leq 1 \) or there exists a test \( \tilde{x} \) such that

\[
\Pr_{h \in V}[h(\tilde{x}) = 1] \in [q, 1 - q]
\]

Note that by permuting the indices of \( x \) and \( x' \), we can make the first index the one that is different between \( x \) and \( x' \). Additionally, for the remaining indices we can flip the 0’s and 1’s of the test so long as we flip the non-zero entries of \( w_h \) at that same position, and change \( b_h \) accordingly. We flip the bits so that \( x \) becomes \( x^{(0)} \) and \( x' \) becomes \( x^{(1)} \).

Note that \( h(x^{(0)}) = 0 \) implies that \( 0 \leq b_h \). Further note that, \( h(x^{(1)}) = 1 \) implies that \( w_{h,1} > b_h \). Thus, the only possibility is that \( w_{h,1} = 1 \) and \( b_h = 0 \).

Let \( T_{+-} \) denote the number of flips from positive to negative weights and let \( T_{-+} \) denote the number of flips from negative to positive. Then, the weights for the new (reduction) problem will be

\[
\begin{align*}
  w_{new}^{(+)} &= w^{(+)} + T_{+-} - T_{-+} \\
  w_{new}^{(-)} &= w^{(-)} + T_{-+} - T_{+-} \\
  0 &= b_{new} = b - T_{+-} + T_{-+}
\end{align*}
\]

From the last equation, \( b = T_{+-} - T_{-+} \). Thus,

\[
\begin{align*}
  w_{new}^{(+)} &= w^{(+)} - b \\
  w_{new}^{(-)} &= w^{(-)} + b
\end{align*}
\]

Since,

\[
\begin{align*}
  w^{(+)} - b & \leq r(w^{(-)} + b) - \frac{d}{8} \\
  w^{(-)} + b & \leq r(w^{(+)} - b - 1) - \frac{d}{8}
\end{align*}
\]

then,

\[
\begin{align*}
  w_{new}^{(+)} &\leq r(w_{new}^{(-)} - \frac{1}{8}d) \\
  w_{new}^{(-)} &\leq r(w_{new}^{(+)} - 1) - \frac{1}{8}d
\end{align*}
\]

We can see that the hypothesis conditions for the original theorem imply that \( \Delta(x^{(0)}, x^{(1)}) \) is a subset of the hypotheses that satisfy the conditions based on \( w_{new}^{(-)} \) and \( w_{new}^{(+)} \) so Lemma 6.3 implies the binary linear classifier is \( \frac{1}{q} \)-split-neighborly which means \( \max(16, 8r) \)-split-neighborly.

6.7.3 Proof of Lemma 6.3

The remainder of this is devoted to proving Lemma 6.3.

We begin by fixing a subset \( V \subseteq H' \). As in all the application proofs, we will start by assuming by contradiction that there is no test with a split constant in the range \( [q, 1 - q] \). We will use this contradiction to show that the size of \( V \) is small.
Recall that $b_h = 0$ for all hypotheses in the reduced problem and $w_{h,1} = 1$. This follows from the fact that $h(x(0)) = 0$ and $h(x(1)) = 1$.

### 6.7.4 Majority Vector

Let $e_i$ be an elementary vector with all entries 0 except for the $i^{th}$ entry which is 1.

**Lemma 6.4.** There exists a vector $m \in \{-1, 0, 1\}^d$ such that $\forall i : m_i = 0 : \Pr_{h \in V}[w_{h,i} = m_i] \geq 1 - 2q$ and $\forall i : m_i \neq 0 : \Pr_{h \in V}[w_{h,i} = m_i] \geq 1 - q$

**Proof.** By the contradiction assumption, there isn’t a test with a split constant greater than $q$, \[
\Pr_{h \in V}[w_h \cdot e_i > b_h] \notin [q, 1 - q]
\]
\[
\Pr_{h \in V}[w_{h,i} > 0] \notin [q, 1 - q]
\]
\[
\Pr_{h \in V}[w_{s,i} = 1] \notin [q, 1 - q]
\]

Also, by the contradiction assumption,
\[
\Pr_{h \in V}[w_h \cdot (e_0 + e_i) > b_h] \notin [q, 1 - q]
\]
\[
\Pr_{h \in V}[1 + w_{h,i} > 0] \notin [q, 1 - q]
\]
\[
\Pr_{h \in V}[w_{h,i} \neq -1] \notin [q, 1 - q]
\]
\[
\Pr_{h \in V}[w_{h,i} = -1] \notin [q, 1 - q]
\]

Since $\Pr_{h \in V}[w_{h,i} = 1] + \Pr_{h}[w_{h,i} = 0] + \Pr_{h}[w_{h,i} = 1] = 1$,
\[
\Pr_{h \in V}[w_{h,i} = 0] \notin [q, 1 - 2q]
\]

Thus, each index is either mostly 1, mostly 0, or mostly $-1$ for elements in $S$ (since $q < 1/3$). Define $m \in \{-1, 0, 1\}^d$ such that
\[
m_i = \arg \max_c \Pr_{h \in V}[w_{h,i} = c]
\]

\[\Box\]

### 6.7.5 Ratio between $m(-)$ and $m(+)\]

Note,
\[
\mathbb{E}_{h \in V}[w_h^{(+)}] = \sum_{i=1}^d \Pr[w_{h,i} = 1]
\]
\[
\leq (q)(d - m(+) + (1)m(+)) = qd + (1 - q)m(+)
\]
\[
m(+) \geq \frac{1}{1 - q} \mathbb{E}_{h \in V}[w_h^{(+)}] - qd
\]

Further note,
\[
\mathbb{E}_{h \in V}[w_h^{(+)\]} = \sum_{i=1}^d \Pr[w_{h,i} = 1]
\]
\[
\geq (0)(d - m(+) + (1 - q)m(+) = (1 - q)m(+)
\]
\[
m(+) \leq \frac{1}{1 - q} \mathbb{E}_{h \in V}[w_h^{(+)}]
\]

We have similar equations for $m(-)$ and $\mathbb{E}_{h \in V}[w_h^{(-)}]$

Let $\bar{m}$ be the vector of $m$ without the first component.

Recall that we have
\[
\forall h \in V : w_h^{(+)} \leq r w_h^{(-)} - \frac{1}{8}d
\]
\[
\mathbb{E}_{h \in V}[w_h^{(+)}] \leq r \mathbb{E}_{h \in V}[w_h^{(-)}] - qr d
\]
\[
\frac{1}{1 - q} \mathbb{E}_{h \in V}[w_h^{(+)}] \leq r \frac{1}{1 - q} \mathbb{E}_{h \in V}[w_h^{(-)}] - qr d
\]
\[
m(+) \leq r \bar{m}(+)
\]
\[
\bar{m}(+) \leq r \bar{m}(-)
\]

Also, recall,
\[
\forall h \in V : w_h^{(-)} \leq r (w_h^{(+)} - 1) - \frac{1}{8}d
\]
\[
\mathbb{E}_{h \in V}[w_h^{(-)}] \leq r \mathbb{E}_{h \in V}[w_h^{(+)}] - qr d - r
\]
\[
\frac{1}{1 - q} \mathbb{E}_{h \in V}[w_h^{(-)}] \leq r \frac{1}{1 - q} (\mathbb{E}_{h \in V}[w_h^{(+)}] - qr) - \frac{r}{1 - q}
\]
\[
m(-) \leq r \bar{m}(+) - \frac{r}{1 - q}
\]
\[
m(-) \leq r \bar{m}(+)
\]

Note that $m_1 = 1$. 

---

**Generalized Binary Search For Split-Neighborly Problems**

---
6.7.6 Partition

Let \( \bar{w} \) be the vector of \( w \) without the first component.

**Definition 6.1.** Let

- \( \mathcal{X}^+ = \{ x : \Pr_{h \in V}[\bar{w}_h \cdot x \geq 1] > 1 - q \} \)
- \( \mathcal{X}^0 = \{ x : \Pr_{h \in V}[\bar{w}_h \cdot x = 0] > 1 - 2q \} \)
- \( \mathcal{X}^- = \{ x : \Pr_{h \in V}[\bar{w}_h \cdot x \leq -1] > 1 - q \} \)

**Lemma 6.5.** \( \mathcal{X}^+, \mathcal{X}^0, \mathcal{X}^- \) is a partition of \( \{0, 1\}^{d-1} \)

**Proof.** Since \( q \leq 1/4 \) and the three defining events are mutually exclusive. It is clear that \( \mathcal{A}^+, \mathcal{A}^0, \mathcal{A}^- \) are disjoint. Next we show that every point is in at least one of the sets. Suppose a point \( x \) is in neither \( \mathcal{A}^+ \) or \( \mathcal{A}^- \).

Using the contradiction assumption on the test \((0, x)\),

\[
\Pr_{h \in V}[w_h \cdot (0, x) > 0] \not\in [q, 1 - q]
\]

\[
\Pr_{h \in V}[\bar{w}_h \cdot x > 0] \not\in [q, 1 - q]
\]

\[
\Pr_{h \in V}[\bar{w}_h \cdot x > 0] < q
\]

Using the contradiction assumption on the test \((1, x)\),

\[
\Pr_{h \in V}[w_h \cdot (1, x) > 0] \not\in [q, 1 - q]
\]

\[
\Pr_{h \in V}[1 + \bar{w}_h \cdot x > 0] \not\in [q, 1 - q]
\]

\[
\Pr_{h \in V}[\bar{w}_h \cdot x \geq 0] \not\in [q, 1 - q]
\]

\[
\Pr_{h \in V}[\bar{w}_h \cdot x < 0] < q
\]

Combining these,

\[
\Pr_{h \in V}[\bar{w}_h \cdot x = 0] = 1 - \Pr_{h \in V}[\bar{w}_h \cdot x > 0] - \Pr_{h \in V}[\bar{w}_h \cdot x < 0] \geq 1 - 2q
\]

Thus, \( x \not\in \mathcal{X}^+ \) and \( x \not\in \mathcal{X}^- \) imply \( x \in \mathcal{A}^0 \) so the three sets are a partition. \( \square \)

**Definition 6.2.** Define \( \mathcal{X}^* \) to be every \( x \in \{0, 1\}^d \) that \((\bar{m} = 1) \cdot x = (\bar{m} = -1) \cdot x \), where \((\bar{m} = 1)\) is the element-wise boolean function.

Intuitively, this means that there are as many ones of \( x \) in positions where \( \bar{m} = 1 \) as there are places where \( \bar{m} = -1 \).

**Lemma 6.6.** \( \mathcal{X}^* \subseteq \mathcal{X}^0 \)

**Proof.** We prove this by induction on the number of 1’s in \( x \) for \( x \in \mathcal{X}^* \).

The base case is \( x = 0^d \) which is trivially in \( \mathcal{X}^0 \).

For other \( x \), suppose \( x_i = 1 \) at a location where \( m_i = 0 \). Then we know \( x - e_i \in \mathcal{X}^0 \) by the induction hypothesis.

\[
Pr_{h \in V}[w_h \cdot (x - e_i) = 0] > 1 - 2q
\]

\[
Pr_{h \in V}[w_{h,i} = 0] > 1 - 2q
\]

From these,

\[
Pr_{h \in V}[w_h \cdot x = 0] > 1 - 4q \geq q
\]

for \( q \leq 1/5 \). So \( x \not\in \mathcal{X}^+ \cup \mathcal{X}^- \) and thus \( x \in \mathcal{X}^0 \).

The only other case is where \( x_i = x_j = 1 \) at locations where \( m_i = 1 \) and \( m_j = -1 \). Then we know \( x - e_i - e_j \in \mathcal{X}^0 \) from the induction hypothesis.

\[
Pr_{h \in V}[w_h \cdot (x - e_i - e_j) = 0] > 1 - 2q
\]

\[
Pr_{h \in V}[w_{h,i} = 1] > 1 - q
\]

\[
Pr_{h \in V}[w_{h,j} = -1] > 1 - q
\]

From these,

\[
Pr_{h \in V}[w_h \cdot x = 0] > 1 - 4q \geq q
\]

and similarly, \( x \in \mathcal{X}^0 \). \( \square \)

6.7.7 Probability Distribution

We now define a probability distribution over \( x \in \mathcal{X}^* \).

Without loss of generality, suppose \( \bar{m}^{(+)} \geq \bar{m}^{(-)} \).

- Randomly draw an injection \( f : \{i : \bar{m}_i = -1\} \rightarrow \{i : \bar{m}_i = 1\} \).
- Initialize \( x = 0^{d-1} \)
- For indices \( \{i : \bar{m}_i \leq 0\} \), draw \( x_i \sim \text{bernoulli}(1/2) \).
- For \( \{i : \bar{m}_i = -1\} \), set \( x_{f(i)} = x_i \)

Note that the result \( x \in \mathcal{X}^* \) because of the pairing \( f \), there will be a 1 where \( \bar{m}_i = 1 \) for each 1 where \( \bar{m}_i = -1 \).
Generalized Binary Search For Split-Neighborly Problems

6.7.8 Set $T$

Definition 6.3. For the probability distribution, $$Q(h) = \Pr_{x \in X^* \mid w_h \cdot x = 0}$$

Lemma 6.7. Let $T = \{h \in V : Q(h) > 1 - 4q\}$, then $|T| > 5|T|$.

Proof. For $x \in X^*$, since $X^* \subseteq X^0$,

$$\Pr_{h \in V \mid w_h \cdot x = 0 > 1 - 2q} \sum_{x \in X^0} P(x) \Pr_{h \in V \mid w_h \cdot x = 0} > 1 - 2q \sum_{x \in X^0} P(x) \Pr_{h \in V \mid w_h \cdot x = 0} > 1 - 2q$$

Lemma 6.8. $|T| \leq 3$

Proof. Recall that $\bar{m}^+(x) \leq r\bar{m}^(-) + r\bar{m}^(-)$ as well.

Also $1 - 4q \geq 1 - \min(\frac{1}{16}, \frac{1}{2r})$ since $q \leq \min(\frac{1}{16}, \frac{1}{2r})$

For any $t \in T$, $Q(t) > 1 - 4q \geq 1 - \min(\frac{1}{16}, \frac{1}{2r})$. Define $\text{Ber}(1/2)$ to be a Bernoulli random variable.

$$\Pr_{x \in X^* \mid w_h \cdot x = 0 > 1 - \min(\frac{1}{4}, \frac{1}{2r})} \sum_{x \in X^0} P(x) \Pr_{h \in V \mid w_h \cdot x = 0} > 1 - \min(\frac{1}{4}, \frac{1}{2r})$$

Note that

$$\Pr_{x \in X^* \mid w_h \cdot x = 0 > 1 - \min(\frac{1}{4}, \frac{1}{2r})} \sum_{x \in X^0} P(x) \Pr_{h \in V \mid w_h \cdot x = 0} > 1 - \min(\frac{1}{4}, \frac{1}{2r})$$

unless $\forall i : m_i = 0 : w_{t,i} = 0$ and $\forall i : m_i = -1 : w_{t,i} + w_{t,f(i)} = 0$, call this condition $(t, f)$.

$$\frac{1}{2}(1 - \min(\frac{1}{4}, \frac{1}{2r})) > 1 - \min(\frac{1}{4}, \frac{1}{2r})$$

If $\bar{m}^(-) = 0$, then $\bar{m}^+ = 0$, and thus $\forall i : m_i = 0 : w_{t,i} = 0$ so $t = 0$ and $|T| = 1 \leq 3$.

Note that $\Pr_{f \mid w_{t,i} = -w_{t,f(i)}} \geq 1/2$ implies that $\forall i : m_i = 0 : w_{t,i} = 0$.

Lemma 6.9. If there exists $i, j$ such that $m_i = m_j = -1$, then $w_{t,i} = w_{t,j}$.

Proof. $\Pr_{f \mid w_{t,i} = -w_{t,f(i)}} > 1/2$ means that

$$\Pr_{f \mid w_{t,i} = -w_{t,f(i)}} > 1/2$$

so

$$\frac{\{l : m_i = 1 \wedge w_{t,l} = -w_{t,i}\}}{\{l : m_i = 1\}} > 1/2$$

which is only possible if $w_{t,i} = w_{t,j}$.

Thus, there is some $c \in \{-1, 0, 1\}$ such that $\forall i : m_i = 1 : w_{t,i} = c$.

$$\Pr_{f \mid \text{condition}(t, f)} > \frac{1}{r}$$

$$\Pr_{f \mid \text{condition}(t, f)} > \frac{1}{r}$$

$1 - \Pr_{f \mid \text{condition}(t, f)} > \frac{1}{r}$

Suppose $\exists j : m_j = 1 : w_{t,j} = -c$.

$$\Pr_{f \mid \text{condition}(t, f)} > \frac{1}{r}$$

which is a contradiction. So $\forall j : m_j = 1 : w_{t,j} = -c$. 


Thus, $c$ completely determines $t$. Since there are three options for $c$, there are three options for $t$, and $|T| \leq 3$.

Since $|T| \leq 3$ and $2|T| \geq |V|$, $|V| \leq 6$. Thus, there is a split of $1/6$ which is a contradiction since $q \leq \frac{1}{5}$.

Thus, the lemma is proved. And thus the binary linear classifier problem is split-neighborly.