

On the Evolution of Monotone Conjunctions: Drilling for Best Approximations

Dimitrios I. Diochnos¹

Department of Computer Science, University of Virginia, Charlottesville VA, USA

Abstract. We study the evolution of monotone conjunctions using local search; the fitness function that guides the search is correlation with Boolean loss. Building on the work of Diochnos and Turán [6], we generalize Valiant’s algorithm [19] for the evolvability of monotone conjunctions from the uniform distribution \mathcal{U}_n to *binomial distributions* \mathcal{B}_n .

With a drilling technique, for a frontier q , we exploit a structure theorem for best q -approximations. We study the algorithm using hypotheses from their natural representation ($\mathcal{H} = \mathcal{C}$), as well as when hypotheses contain at most q variables ($\mathcal{H} = \mathcal{C}_{\leq q}$). Our analysis reveals that \mathcal{U}_n is a very special case in the analysis of binomial distributions with parameter p , where $p \in \mathcal{F} = \{2^{-1/k} \mid k \in \mathbb{N}^*\}$. On instances of dimension n , we study approximate learning for $0 < p < 2^{-\frac{1}{n-1}}$ when $\mathcal{H} = \mathcal{C}$ and for $0 < p < \sqrt[n]{2/3}$ when $\mathcal{H} = \mathcal{C}_{\leq q}$. Thus, in either case, approximate learning can be achieved for any $0 < p < 1$, for sufficiently large n .

Keywords: evolution, evolvability, PAC learning, noise, evolutionary algorithms, optimization, local search, distribution-specific learning, binomial distributions, correlation, Boolean loss

1 Introduction

Valiant introduced in [19] a framework for analyzing evolution, called *evolvability*. The purpose is to allow and explain the evolution of complex mechanisms in realistic population sizes within realistic time periods. Evolution is treated as a form of computational learning from examples (experiences) and is a restricted form of the *probably approximately correct* (PAC) model of learning [18].

Noise was first studied in the framework of PAC learning by Angluin and Laird [3] and many subsequent results have been obtained in the statistical queries model which is due to Kearns [15]; see also [4, 17]. Apart from classification noise, noise on the attributes has also been considered [12]. Noise is natural in evolvability as the functionalities that evolve over time realize their fitness through interaction with the environment (sampling); not by interpreting tiny differences of the true fitness values given in some compact representation. In fact, Feldman showed in [8] that evolvability is equivalent to learning with *correlational statistical queries* [5]. However, as also pointed out by Feldman, this translation is not necessarily the most efficient or intuitive method in general. Hence, it is common to discuss *distribution-specific* results on the analysis

of intuitive algorithms in the framework of evolvability; e.g. [16, 14, 2]. Thus, the study of simple and intuitive evolvability algorithms using Valiant’s original Boolean loss is of interest for specific distributions. Our aim is to understand better such algorithms in the framework of evolvability as well as in the broader framework of *optimization* and *evolutionary algorithms* (EAs) [22, 7].

Previous work in evolvability includes [19, 21, 9, 11, 8, 10, 20, 13, 16, 14, 2]. In [19] Valiant introduced a swapping-type algorithm and proved the evolvability of monotone conjunctions under the uniform distribution (\mathcal{U}_n). The analysis was simplified by Diochnos and Turán in [6] and in fact it was shown that monotone conjunctions are evolvable in $\mathcal{O}(\log(1/\varepsilon))$ generations. The result was strengthened to general conjunctions under \mathcal{U}_n by Kanade, Valiant and Vaughan in [14] including target drift. Further, Feldman in [8] showed that conjunctions are evolvable for any fixed distribution in $\tilde{\mathcal{O}}(n)$ generations, where $\tilde{\mathcal{O}}(\cdot)$ ignores poly-log factors. Kanade in [13] extended Valiant’s model to include genetic recombination where it follows that conjunctions are evolvable in $\mathcal{O}\left((\log(n)/\varepsilon)^2\right)$ generations. On the other hand, one open question from [13] was whether the analysis of Diochnos and Turán could be generalized to distributions beyond \mathcal{U}_n .

In this paper we address this last question by considering binomial distributions \mathcal{B}_n with parameter p . We do so by exploiting a structure theorem for best approximations with a *drilling* technique. Drilling improves our estimates of the fitness function by increasing the sample size. In turn, we can discover any important variable for targets up to a certain size beyond the frontier q of our search. Hence, even if we have the power to form some targets precisely, the evolutionary mechanism only forms a best approximation for them. This way, targets with many variables are dealt in an easy way. Our analysis reveals the family $\mathcal{F} = \{2^{-1/k} \mid k \in \mathbb{N}^*\}$, where \mathcal{U}_n is the first member and is obtained for $k = 1$; i.e. $p = \frac{1}{2}$. As we consider larger values of p in the $(0, 1)$ interval, every time we encounter one more member of \mathcal{F} , we *drill deeper*, thus allowing evolution to identify variables from targets containing one more variable. Evolvability follows for any fixed distribution in $\mathcal{O}\left(\log_{\frac{1}{p}}(1/\varepsilon)\right)$ generations; the setup of [6] for \mathcal{U}_n is recovered as a special case. Our analysis reveals an interesting non-trivial connection between the parameters, which is captured in Figure 3.

The paper is structured as follows. Section 2 gives the definition of evolvability and Section 3 preliminaries specific to our setup. Section 4 lays the foundations for the evolvability of monotone conjunctions. Section 5 discusses adaptation. Section 6 discusses the convergence. Section 7 analyzes the complexity. We conclude with further remarks in Section 8. Due to space limitations some proofs are sketched or omitted in this version.

2 Definition of Evolvability

The truth values TRUE and FALSE are represented by 1 and -1 respectively. The fitness function that guides the search is called *performance*. For a target c and

a fixed distribution \mathcal{D}_n over $\{0, 1\}^n$, the performance of a hypothesis h is

$$\text{Perf}_{\mathcal{D}_n}(h, c) = \sum_{x \in \{0, 1\}^n} h(x) \cdot c(x) \cdot \Pr_{x \sim \mathcal{D}_n}(x), \quad (1)$$

called the *correlation* of h and c . Evolution starts with an initial hypothesis h_0 , and produces a sequence of hypotheses using a local-search procedure in \mathcal{H} . Similarity between h and c in an underlying distribution \mathcal{D}_n is measured by the *empirical performance* function $\text{Perf}_{\mathcal{D}_n}(h, c, |S|)$ which is evaluated approximately by drawing a random sample S and computing $\text{Perf}_{\mathcal{D}_n}(h, c, |S|) = \frac{1}{|S|} \sum_{x \in S} h(x) \cdot c(x)$. Valiant's original definition of evolvability treated the confidence parameter δ and the error parameter ε as one. Below, even if we draw the definitions from [19], we modify them slightly to also include δ explicitly.

Definition 1 (Modified from [19]). For a polynomial $p(\cdot, \cdot)$ and a representation class R a p -neighborhood N on R is a pair M_1, M_2 of randomized polynomial time Turing machines such that the numbers n (in unary), $\lceil 1/\varepsilon \rceil$ and a representation $r \in R_n$ act as follows: M_1 outputs all the members of a set $\text{Neigh}_N(r, \varepsilon) \subseteq R_n$, that contains r and may depend on random coin tosses of M_1 , and has size at most $p(n, 1/\varepsilon)$. If M_2 is then run on this output of M_1 , it in turn outputs one member of $\text{Neigh}_N(r, \varepsilon)$, with member r_1 being output with a probability $\Pr_N(r, r_1) \geq 1/p(n, 1/\varepsilon)$.

Definition 2 (Modified from [19]). For confidence parameter δ , error parameter ε , positive integers n and s , an ideal function $f \in \mathcal{C}_n$, a representation class R with $p(n, 1/\varepsilon)$ -neighborhood N on R , a distribution \mathcal{D} , a representation $r \in R_n$ and a real number t , the mutator $Mu(f, p(n, 1/\varepsilon), R, N, \mathcal{D}, s, r, t)$ is a random variable that on input $r \in R_n$ takes a value $r_1 \in R_n$ determined as follows: For each $r_1 \in \text{Neigh}_N(r, \varepsilon)$ it first computes an empirical value of $\nu(r_1) = \text{Perf}_{\mathcal{D}_n}(r_1, f, s)$. Let $Bene$ be the set $\{r_1 \mid \nu(r_1) > \nu(r) + t\}$ and $Neut$ be the set difference $\{r_1 \mid \nu(r_1) \geq \nu(r) - t\} \setminus Bene$. If $Bene \neq \emptyset$ then output $r_1 \in Bene$ with probability $\Pr_N(r, r_1) / \sum_{r_1 \in Bene} \Pr_N(r, r_1)$. Otherwise ($Bene = \emptyset$), output an $r_1 \in Neut$, the probability of a specific r_1 being $\Pr_N(r, r_1) / \sum_{r_1 \in Neut} \Pr_N(r, r_1)$.

Definition 3 (Modified from [19]). For a mutator $Mu(f, p(n, 1/\varepsilon), R, N, \mathcal{D}, s, r, t)$ a t -evolution step on input $r_1 \in R_n$ is the random variable $r_2 = Mu(f, p(n, 1/\varepsilon), R, N, \mathcal{D}, s, r_1, t)$. We then say $r_1 \rightarrow r_2$ or $r_2 \leftarrow Evolve(f, p(n, 1/\varepsilon), R, N, \mathcal{D}_n, s, r_1, t)$.

We say that polynomials $t_\ell(x, y)$ and $t_u(x, y)$ are *polynomially related* if for some $\eta > 1$ for all $x, y (0 < x, y < 1) (t_u(x, y))^\eta \leq t_\ell(x, y) \leq t_u(x, y)$.

Definition 4 (Modified from [19]). For a mutator $Mu(f, p(n, 1/\varepsilon), R, N, \mathcal{D}, s, r, t)$ a (t_ℓ, t_u) -evolution sequence for $r_1 \in R_n$ is a random variable that takes as values sequences r_1, r_2, r_3, \dots such that for all i $r_i \leftarrow Evolve(f, p(n, 1/\varepsilon), R, N, \mathcal{D}, s, r_{i-1}, t_i)$, where $t_\ell(1/n, \varepsilon) \leq t_i \leq t_u(1/n, \varepsilon)$, t_ℓ and t_u are polynomially related polynomials, and t_i is the output of a TM T on input r_{i-1}, n, ε and δ .

Definition 5 (Goal of Evolution; modified from [19]). For polynomials $p(n, 1/\varepsilon)$, $s(n, 1/\varepsilon, 1/\delta)$, $t_\ell(1/n, \varepsilon)$ and $t_u(1/n, \varepsilon)$, a representation class R and $p(n, 1/\varepsilon)$ -neighborhood N on R , the class \mathcal{C} is (t_ℓ, t_u) -evolvable by $(p(n, 1/\varepsilon), R, N, s(n, 1/\varepsilon, 1/\delta))$ over distribution \mathcal{D} if there is a polynomial $g(n, 1/\varepsilon, 1/\delta)$ and a Turing machine T , which computes a tolerance bounded between t_ℓ and t_u , such that for every positive integer n , every $f \in \mathcal{C}_n$, every $\delta > 0$, every $\varepsilon > 0$, and every $r_0 \in R_n$ it is the case that with probability at least $1 - \delta$, a (t_ℓ, t_u) -evolution sequence r_0, r_1, r_2, \dots , where $r_i \leftarrow \text{Evolve}(f, p(n, 1/\varepsilon), R, N, \mathcal{D}_n, s(n, 1/\varepsilon, 1/\delta), r_{i-1}, T(r_{i-1}, n, \varepsilon))$, will have $\text{Perf}_{\mathcal{D}_n}(r_{g(n, 1/\varepsilon, 1/\delta)}, f) \geq 1 - \varepsilon$.

The number of generations needed for evolution is upper bounded by $g(n, \frac{1}{\varepsilon}, \frac{1}{\delta})$.

Definition 6 (Modified from [19]). A class \mathcal{C} is evolvable by $(p(n, 1/\varepsilon), R, N, s(n, 1/\varepsilon, 1/\delta))$ over \mathcal{D} iff for some pair of polynomially related polynomials t_ℓ, t_u , \mathcal{C} is (t_ℓ, t_u) -evolvable by $(p(n, 1/\varepsilon), R, N, s(n, 1/\varepsilon, 1/\delta))$ over \mathcal{D} .

Definition 7 (Modified from [19]). A class \mathcal{C} is evolvable by R over \mathcal{D} iff for some polynomials $p(n, 1/\varepsilon)$ and $s(n, 1/\varepsilon, 1/\delta)$, and some $p(n, \frac{1}{\varepsilon})$ -neighborhood N on R , \mathcal{C} is evolvable by $(p(n, 1/\varepsilon), R, N, s(n, 1/\varepsilon, 1/\delta))$ over \mathcal{D} .

3 Preliminaries

Given a set of Boolean variables x_1, \dots, x_n , we assume that there is an unknown target $c \in \mathcal{C}$, a monotone conjunction of some of these variables. Let \mathcal{C} be the concept class of all possible conjunctions in their natural representation. For a threshold q , let $\mathcal{C}_{\leq q}$ be the set of monotone conjunctions from \mathcal{C} that contain at most q variables. Further, let $\mathcal{C}_{> q} = \mathcal{C} \setminus \mathcal{C}_{\leq q}$ be the set of conjunctions from \mathcal{C} that are not included in $\mathcal{C}_{\leq q}$.

By Definition 2, the neighborhood is split in 3 parts by the *increase* in performance that the hypotheses in the neighborhood offer. There are *beneficial*, *neutral*, and *deleterious* mutations. Thus, we need an oracle for computing

$$\Delta = \text{Perf}_{\mathcal{D}_n}(h', c) - \text{Perf}_{\mathcal{D}_n}(h, c), \quad (2)$$

and hence, for a given t , determine the set where $h' \in N$ lies. Now let

$$h = \bigwedge_{i=1}^m x_i \wedge \bigwedge_{\ell=1}^r y_\ell \quad \text{and} \quad c = \bigwedge_{i=1}^m x_i \wedge \bigwedge_{k=1}^u w_k. \quad (3)$$

The x 's are *mutual* variables, the y 's are called *redundant* and the w 's are called *undiscovered* or *missing*. Variables in the target c are called *good*, otherwise *bad*. With $|h|$ we denote the *size* (or *length*) of a conjunction; the number of variables that it contains. A *binomial distribution* over $\{0, 1\}^n$ is specified by the probability p of setting each variable x_i to 1. A truth assignment $(a_1, \dots, a_n) \in \{0, 1\}^n$ has probability $\prod_{i=1}^n p^{a_i} \cdot (1-p)^{1-a_i}$. We write \mathcal{B}_n to denote a fixed binomial distribution, omitting p for simplicity. On an instance of dimension

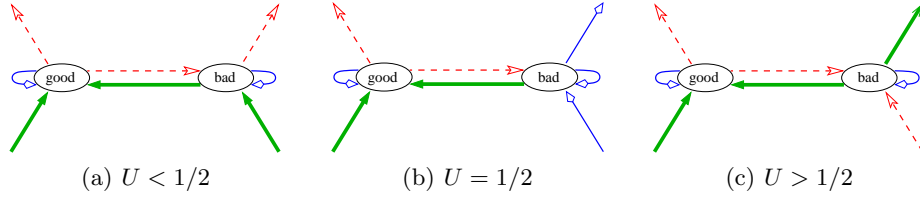


Fig. 1. Arrows pointing towards the nodes indicate addition of one variable and arrows pointing away from a node indicate removal of one variable. This is consistent with arrows indicating swapping a pair of variables. Thick solid lines indicate $\Delta > 0$. Simple lines indicate $\Delta = 0$. Dashed lines indicate $\Delta < 0$. Let U be the weight of the undiscovered variables. Figure 1(a) holds when $U < 1/2$, Figure 1(b) holds when $U = 1/2$ and Figure 1(c) holds when $U > 1/2$.

n we say that \mathcal{B}_n has *low density* when $0 < p < \frac{1}{2}$, *medium density* when $\frac{1}{2} \leq p \leq 2^{-\frac{1}{n}}$, *high density* when $2^{-\frac{1}{n}} < p < \sqrt[n]{2/3}$, and *very high density* when $\sqrt[n]{2/3} \leq p < 1$. Consider a target c and a hypothesis h as in (3). Then (1) gives

$$\text{Perf}_{\mathcal{B}_n}(h, c) = 1 - 2p^{m+r} - 2p^{m+u} + 4p^{m+r+u}. \quad (4)$$

Figure 1 presents the sign of Δ that guides the search. Note that while the sign of an arrow may be fully determined, it is the value of the tolerance t that defines the two sets of interest (Bene and Neut) that guide the search. Figure 1(a) refers to the *expansion phase*, Figure 1(b) to the *identification phase* and Figure 1(c) to the *shrinking phase*.

3.1 The Swapping Algorithm

The swapping algorithm for monotone conjunctions was introduced by Valiant in [19] and was also analyzed in [6]. The neighborhood N of a conjunction h is the set of conjunctions that arise by *adding* a variable (neighborhood N^+), *removing* a variable (neighborhood N^-), or *swapping* a variable with another one (neighborhood N^{+-}), plus the conjunction itself¹. Thus, $N = N^- \cup N^+ \cup N^{+-} \cup \{h\}$. As an example, let $h = x_1 \wedge x_2$, and $n = 3$. Then, $N^- = \{x_1, x_2\}$, $N^+ = \{x_1 \wedge x_2 \wedge x_3\}$, and $N^{+-} = \{x_3 \wedge x_2, x_1 \wedge x_3\}$. Note that $|N| = \mathcal{O}(n |h|)$ in general. Algorithm 1 presents the mutator function for the swapping algorithm.

Compute-q uses Table 2 or (11) to set q depending on the hypothesis class \mathcal{H} that is used for evolution. (Table 2 used for **Compute-q**, already incorporates a modified ε when needed.) Line 6 computes the minimum non-zero value A of $\mathcal{A}(u) = |1 - 2p^u|$ for $u \in \{0, \dots, n\}$ using Table 1 from Section 4.1. Tolerance t is normally t_ℓ ; however, when $\mathcal{H} = \mathcal{C}$ and $|h| > q$ then $t = t_u$. We discuss tolerance in Sections 4.2 and 7. **Performance** computes the empirical performance of h w.r.t. c over the distribution \mathcal{B}_n with parameter p , within ε_s of its true value,

¹ As h will be clear from the context, we write N instead of $N(h)$.

Algorithm 1: Mutator function for a binomial distribution

Input: dimension n , $p \in (0, 1)$, $\delta \in (0, 1)$, $\varepsilon \in (0, 2)$, $\mathcal{H} \in \{\mathcal{C}_{\leq q}, \mathcal{C}\}$, $h \in \mathcal{H}$
Output: a new hypothesis h'

- 1 $q \leftarrow \text{Compute-}q(p, \varepsilon, \mathcal{H})$; $\vartheta \leftarrow \lfloor \log_{\frac{1}{p}}(2) \rfloor$;
- 2 **if** $|h| > 0$ **then** Generate N^- **else** $N^- \leftarrow \emptyset$;
- 3 **if** $|h| < q$ **then** Generate N^+ **else** $N^+ \leftarrow \emptyset$;
- 4 **if** $|h| \leq q$ **then** Generate N^{+-} **else** $N^{+-} \leftarrow \emptyset$;
- 5 $Bene \leftarrow \emptyset$; $Neutral \leftarrow \{h\}$;
- 6 $A \leftarrow \min_{\neq 0} \{1 - 2p^u\}$; $\mu = \min \{2p^{q+\vartheta}, A\}$;
- 7 $t_\ell \leftarrow p^{q-1}\mu(1-p)$; $t_u \leftarrow 4p^q(1-p)$;
- 8 **if** $(\mathcal{H} = \mathcal{C})$ **and** $(|h| > q)$ **then** $t \leftarrow t_u$; $\varepsilon_s \leftarrow t_u/4$; $\delta_s \leftarrow \delta/4$;
- 9 **else** $t \leftarrow t_\ell$; $\varepsilon_s \leftarrow t_\ell$; $\delta_s \leftarrow \delta/2$;
- 10 $\text{SetWeight}(h, h, N^-, N^+, N^{+-})$; $\nu_h \leftarrow \text{Perf}(p, h, \varepsilon_s, \delta_s)$;
- 11 **for** $x \in N^+, N^-, N^{+-}$ **do**
- 12 $\text{SetWeight}(x, h, N^-, N^+, N^{+-})$; $\nu_x \leftarrow \text{Perf}(p, x, \varepsilon_s, \delta_s)$;
- 13 **if** $\nu_x > \nu_h + t$ **then** $Bene \leftarrow Bene \cup \{x\}$;
- 14 **else if** $\nu_x \geq \nu_h - t$ **then** $Neutral \leftarrow Neutral \cup \{x\}$;
- 15 **if** $Bene \neq \emptyset$ **then** **return** $\text{Select}(Bene)$ **else** **return** $\text{Select}(Neutral)$;

with probability at least $1 - \delta_s$; see Section 7. **SetWeight** assigns the same weight to all members of $\{h\} \cup N^- \cup N^+$ so that they add up to $\frac{1}{2}$, and the same weight to all the members of N^{+-} so that they add up to $\frac{1}{2}$. **Select** computes the sum of weights W of the conjunctions in the set passed as argument, and returns a hypothesis h' with probability $w_{h'}/W$, where $w_{h'}$ is the weight of h' .

4 Foundations for Evolvability

Let $\log_{\frac{1}{p}}(x)$ be the logarithm of x in base $\frac{1}{p}$. Given a size q and an extension ϑ , a hypothesis h is called *short* when $|h| \leq q$, *medium* when $q < |h| \leq q + \vartheta$ and *long* when $|h| > q + \vartheta$. Given a target c and a size q , we will be interested in the best size q approximation of c . The reason is Theorem 1 below, first proved in [6] for \mathcal{U}_n . Note that the best approximation is not necessarily unique.

Definition 8 (Best q -Approximation). *Let h be a hypothesis such that $|h| \leq q$ and $\forall h' \neq h, |h'| \leq q : \text{Perf}_{\mathcal{D}_n}(h', c) \leq \text{Perf}_{\mathcal{D}_n}(h, c)$. We call h a best q -approximation of c .*

Theorem 1 (Best Approximations under Binomial Distributions; [6]).
The best q -approximation of a target c is c if $|c| \leq q$, or any hypothesis formed by q good variables if $|c| > q$.

Lemma 1 (Performance Lower Bound, Medium Target). *Let \mathcal{B}_n be a binomial distribution. Let c be a medium target. A best q -approximation h has $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - 2p^q$.*

Lemma 2 (Performance Lower Bound, Long Target). *Let \mathcal{B}_n be a binomial distribution. Let h be a hypothesis such that $|h| \geq q$ and consider a long target c . Then, $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - 2p^q (1 + p^{1+\vartheta})$.*

We now examine the difference Δ between the current hypothesis h and a hypothesis h' that is generated in each neighborhood.

Comparing $h' \in N^+$ with h . We introduce a variable z in the hypothesis h . If z is good, $\Delta = 2p^{|h|}(1-p) > 0$. If z is bad, $\Delta = 2p^{|h|}(1-2p^u)(1-p)$.

Comparing $h' \in N^-$ with h . We remove a variable z from the hypothesis h . If z is good, $\Delta = -2p^{|h|-1}(1-p) < 0$. If z is bad, $\Delta = -2p^{|h|-1}(1-2p^u)(1-p)$.

Comparing $h' \in N^{+-}$ with h . Replacing a good with a bad variable gives $\Delta = -4p^{|h|+u}(1-p)$. Replacing a good with a good, or a bad with a bad variable gives $\Delta = 0$. Replacing a bad with a good variable gives $\Delta = 4p^{|h|+u-1}(1-p)$.

Our aim for short and medium targets is to have the ability to determine the signs of the differences Δ in every case. For long targets, we want to determine the signs of the Δ 's for the mutations that arise in the N^+ and N^- neighborhoods; not necessarily for those in the N^{+-} neighborhood. We denote

$$\mathcal{A}(u) = |1 - 2p^u|, \quad u \in \{0, \dots, n\}. \quad (5)$$

As $\mathcal{A}(u)$ appears in the Δ 's for the mutations in the N^+ and N^- neighborhoods, we need to study the minimum non-zero value that $\mathcal{A}(u)$ can attain for $u \in \{0, 1, \dots, n\}$ under an arbitrary \mathcal{B}_n . The zeros of $\mathcal{A}(u)$ are found in the family

$$\mathcal{F} = \left\{ 2^{-\frac{1}{k}} \mid k \in \mathbb{N}^* \right\}. \quad (6)$$

4.1 On the Minimum Non-Zero Value of $\mathcal{A}(u)$, $u \in \{0, \dots, n\}$

Lemma 3. *Consider the polynomials $f_k(p) = p^{k+1} + p^k - 1$ defined respectively in the intervals $\mathcal{J}_k = [2^{-1/k}, 2^{-1/(k+1)}]$ with $k \in \mathbb{N}^*$. Then, each f_k is monotone increasing in \mathcal{J}_k and has a (unique) root ξ_k in the open interval \mathcal{J}_k .*

Table 1 and Figure 2 present $\min_{\neq 0} \{\mathcal{A}(u)\}$ as p ranges in $(0, 1)$.

4.2 On Tolerance and Design Requirements

The critical part of the evolution will be evolving short hypotheses. In this part we want to identify swaps precisely for short and medium targets and thus $|\Delta| \geq 4p^{2q+\vartheta-1}(1-p)$. Regarding additions and removals we want to be able to identify the sign of Δ precisely, regardless of the target; thus, using (5), for the non-zero values of Δ , $|\Delta| = 2p^{|h|-1} \cdot \mathcal{A}(u) \cdot (1-p) \geq 2p^{q-1} \cdot \min_{\neq 0} \{\mathcal{A}(u)\} \cdot (1-p)$. Therefore, in order to determine the tolerance, we want to determine

$$2p^{q+\vartheta} < \min_{\neq 0} \{\mathcal{A}(u)\} = \min_{\neq 0} \{|1 - 2p^u|\} = A. \quad (7)$$

Table 1. $\min_{\neq 0} \{\mathcal{A}(u)\}$, attained for specific u by some target c , as p ranges in $(0, 1)$. When $2^{-1/k} < p < 2^{-1/(k+1)}$, then ξ_k is the root from Lemma 3.

density	p	$\min_{\neq 0} \{\mathcal{A}(u)\}$	for u	obtained by target c
low	$0 < p < 1/2$	$1 - 2p$	1	$1 \leq c \leq \min\{n, q + 1\}$
medium	$2^{-1/k}$, with $1 \leq k \leq n - 1$	$1 - p$	$k + 1$	$k + 1 \leq c \leq \min\{n, q + k + 1\}$
	$2^{-1/n}$	$(1 - p)/p$	$n - 1$	$n - 1 \leq c \leq n$
	$2^{-1/k} < p \leq \xi_k$ with $1 \leq k = \lfloor \log_{\frac{1}{p}}(2) \rfloor \leq n - 1$	$2p^k - 1$	k	$k \leq c \leq \min\{n, q + k\}$
	$\xi_k \leq p < 2^{-1/(k+1)}$ with $1 \leq k = \lfloor \log_{\frac{1}{p}}(2) \rfloor \leq n - 1$	$1 - 2p^{k+1}$	$k + 1$	$k + 1 \leq c \leq \min\{n, q + k + 1\}$
(very) high	$2^{-1/n} < p < 1$	$2p^n - 1$	n	$ c = n$

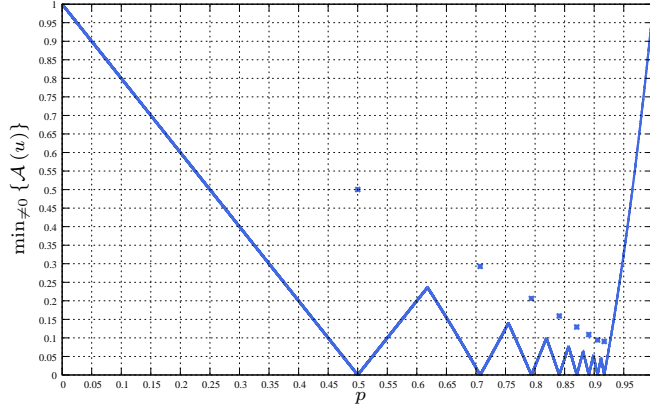


Fig. 2. $\min_{\neq 0} \{\mathcal{A}(u)\}$ for $n = 8$, as presented in Table 1.

We now let

$$\mu = \min \left\{ 2p^{q+\vartheta}, \min_{\neq 0} \{\mathcal{A}(u)\} \right\} \quad \text{and} \quad t_\ell = p^{q-1} \mu (1 - p). \quad (8)$$

Tolerance is set by (8) when evolution takes place in $\mathcal{C}_{\leq q}$. When $\mathcal{H} = \mathcal{C}_{\leq q}$, $t_\ell = t_u$ and this is a special case, *fixed-tolerance* evolvability. On the other hand, if $\mathcal{H} = \mathcal{C} = \mathcal{C}_{\leq q} \cup \mathcal{C}_{> q}$, the approach in $\mathcal{C}_{> q}$ relies on setting the tolerance t_u large enough so that a random walk can be performed and eventually form a hypothesis in $\mathcal{C}_{\leq q}$. The neighborhood in $\mathcal{C}_{> q}$ is $N^- \cup \{h\}$; see Algorithm 1. Thus, $|\Delta| \leq 2p^{|h|-1}(1 - p) \leq 2p^q(1 - p)$ and t_u is set to be $t_u = 2 \cdot \max\{|\Delta|\}$, that is,

$$t_u = 4p^q(1 - p). \quad (9)$$

Thus, requiring $(t_u)^\eta \leq t_\ell \leq t_u < 1$ for some $\eta > 1$, we get the constraints

$$\begin{array}{c|c|c|c} p \leq \frac{1}{4} & \frac{1}{4} < p < \frac{1}{2} & p = \frac{1}{2} & \frac{1}{2} < p \\ \hline q \geq 1 & q \geq 2 & q > 1 & q \geq \log_{\frac{1}{p}}(2) \end{array} \quad (10)$$

5 Adaptation

Corollary 1. $q \geq \log_{\frac{1}{p}} \left(\frac{3}{\varepsilon} \right), \vartheta \geq 0, |h| = q < |c| \leq q + \vartheta \Rightarrow \text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \frac{2\varepsilon}{3}$.

Corollary 2. $q \geq \log_{\frac{1}{p}} \left(\frac{3}{\varepsilon} \right), \vartheta \geq \log_{\frac{1}{p}}(2p), |h| \geq q, |c| > q + \vartheta \Rightarrow \text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$.

5.1 Evolution when $\mathcal{H} = \mathcal{C}$

In light of Corollaries 1 and 2, setting $q = \left\lceil \log_{\frac{1}{p}} \left(\frac{3}{\varepsilon} \right) \right\rceil$ and $\vartheta = \left\lceil \log_{\frac{1}{p}}(2) \right\rceil$ would also satisfy the requirements in (10) for every $0 < \varepsilon < 2$. However, we can improve the frontier q . Depending on p , let q and ϑ be defined from Table 2.

Table 2. Definition of q and ϑ depending on p when evolving on $\mathcal{H} = \mathcal{C} = \mathcal{C}_{\leq q} \cup \mathcal{C}_{> q}$.

p	q	ϑ
$p \leq \frac{1}{4}$	$\left\lceil \log_{\frac{1}{p}}(3/\varepsilon) \right\rceil$	0
$\frac{1}{4} < p < \frac{1}{2}$	$\max \left\{ \left\lceil \log_{\frac{1}{p}}(3/\varepsilon) \right\rceil, 2 \right\}$	0
$p = \frac{1}{2}$	$\max \left\{ \left\lceil \log_{\frac{1}{p}}(3/\varepsilon) \right\rceil, 2 \right\}$	1
$p > \frac{1}{2}$	$\max \left\{ \left\lceil \log_{\frac{1}{p}}(3/\varepsilon) \right\rceil, \left\lceil \log_{\frac{1}{p}}(2) \right\rceil \right\}$	$\left\lceil \log_{\frac{1}{p}}(2) \right\rceil$

Learnability on a Fixed Dimension. Let $\lambda > 0$. Then, $\log_{\frac{1}{p}} \left(\frac{3}{\varepsilon} \right) \leq \lambda \Rightarrow \varepsilon \geq 3p^\lambda$. Approximate learning degenerates to exact, when $\log_{\frac{1}{p}} \left(\frac{3}{\varepsilon} \right) > n - 1$, as due to rounding $q \geq n$. However, we also need to be able to achieve $q = n$ for a value of ε in the $(0, 2)$ interval. Thus, on any fixed dimension n , it makes sense to discuss about approximate learning ($\lambda \leq n - 1$) when $0 < p < \sqrt[n-1]{2/3}$ and about exact ($\lambda = n$) when $p < \sqrt[n]{2/3}$. That is, when $p \geq \sqrt[n]{2/3}$ then the dimension is *too low* to allow even exact learning with our method. Regarding $\left\lceil \log_{\frac{1}{p}}(2) \right\rceil$ from Table 2, approximate learning can be done when $0 < p < 2^{-\frac{1}{n-1}}$ and exact when $0 < p < 2^{-\frac{1}{n}}$. Hence, on an instance of dimension n , we study approximate learning when $0 < p < 2^{-\frac{1}{n-1}}$ and exact learning when $0 < p < 2^{-\frac{1}{n}}$.

Adaptation for Large Input Error. For $\frac{1}{4} < p < \frac{1}{2}$, when $\varepsilon \geq 3p$, evolution will reset ε to $\varepsilon' = 3p^2 \geq \frac{3}{16}$. In the end it will return a hypothesis that has accuracy $1 - \varepsilon' > 1 - \varepsilon$ but through a (t_ℓ, t_u) -evolutionary sequence.

Let $\mathcal{I}_k = [2^{-\frac{1}{k}}, 2^{-\frac{1}{k+1}})$. When $p \in \mathcal{I}_1$, if $\varepsilon \geq 3p$, it will be reset to $\varepsilon' = 3p^2 > \frac{3}{4}$. When $p \in \mathcal{I}_k$ with $\left\lceil \log_{\frac{1}{p}}(2) \right\rceil = k \in \{2, \dots, n - 1\}$, if $\varepsilon \geq 3p^k$, then setting $\varepsilon' = 3p^{k+1} \geq 3 \cdot 2^{-\frac{k+1}{k}} > 1$ implies $q = \left\lceil \log_{\frac{1}{p}}(3/\varepsilon') \right\rceil = \left\lceil \log_{\frac{1}{p}}(2) \right\rceil = k + 1 \leq n$.

Thus, we will treat q as if it is defined *solely* by $q = \lceil \log_{\frac{1}{p}}(\frac{3}{\varepsilon}) \rceil$ in Table 2. If the input ε is too large, evolution will adapt it to an appropriate constant.

5.2 Evolution when $\mathcal{H} = \mathcal{C}_{\leq q}$

Working strictly on $\mathcal{H} = \mathcal{C}_{\leq q}$, one need no longer respect the requirements in (10) as we have fixed-tolerance evolvability; see Section 4.2. Hence, we let

$$q = \lceil \log_{\frac{1}{p}}(3/\varepsilon) \rceil \quad \text{and} \quad \vartheta = \lceil \log_{\frac{1}{p}}(2) \rceil. \quad (11)$$

By restricting the hypothesis class, on an instance of dimension n , evolution can now take place even when p belongs to the high density region, contrasting Section 5.1. Also, no adaptation is needed for any feasible (p, ε) pair.

5.3 Determining $\mu = \min \{2p^{q+\vartheta}, \min_{\neq 0} \{\mathcal{A}(u)\}\}$.

For a specific p , we need to identify the minimum q_m such that $2p^{q_m+\vartheta} < A$. Then, for $\varepsilon < 3p^{q_m-1}$ swaps are more expensive. Thus, q_m satisfies²

$$q_m > \log_{\frac{1}{p}}(2p^\vartheta/A) = \zeta. \quad (12)$$

A *Smooth Frontier* for $\mu = \min \{2p^{q+\vartheta}, \min_{\neq 0} \{\mathcal{A}(u)\}\}$. As q involves rounding, $2p^{q+\vartheta} = 2p^{\lceil \log_{1/p}(3/\varepsilon) \rceil + \lceil \log_{1/p}(2) \rceil} \geq 2p^{1+\log_{1/p}(3/\varepsilon)+\log_{1/p}(2)} = \frac{p\varepsilon}{3}$. Thus, by overestimating the required accuracy for swaps, determining μ can be reduced to the simpler $\frac{p\varepsilon}{3} < A \Leftrightarrow \varepsilon < \frac{3A}{p}$. In other words, μ could also be defined as $\mu = \min \{\frac{p\varepsilon}{3}, \min_{\neq 0} \{\mathcal{A}(u)\}\}$ in (8) and in line 6 of Algorithm 1.

Figure 3 presents all the above relationships between p and ε .

6 Convergence

6.1 Short Initial Hypothesis and Short Target

If $U < \frac{1}{2}$, Figure 1(a) applies. Beneficial mutations can only *add* or *swap* variables. Swaps or additions of good variables increase m . Thus after at most $|c|$ such mutations and at most $q - |h_0|$ additions of bad variables, $U \geq \frac{1}{2}$.

If $U = \frac{1}{2}$, Figure 1(b) applies. $U = \frac{1}{2} \Rightarrow p \in \mathcal{F}$ for some k . Further, $U = \frac{1}{2} \Rightarrow p^u = \frac{1}{2} \Rightarrow u = \log_{\frac{1}{p}}(2) = k$. Also, $(k = u) \wedge (u \leq |c|) \wedge (|c| \leq q) \Rightarrow k \in \{1, \dots, q\}$ ³.

² As p ranges in $(0, 1)$, a natural question in (12) is whether $\zeta \in \mathbb{Z}$; then $q_m = \zeta + 1$, otherwise $q_m = \lceil \zeta \rceil < \zeta + 1$. Equivalently, does $2p^{\zeta+\vartheta} - A = 0$ hold for $\zeta \in \mathbb{Z}$? By Table 1 and the definition of ϑ , for \mathcal{U}_n , $\zeta = 1$. Hence, in \mathcal{U}_n , when $\frac{3}{2} \leq \varepsilon < 3$ then the two quantities for μ in (8) have the same value *for a range of ε values*. Regardless if there are additional integer solutions, q_m can be computed efficiently.

³ A clarification comment is in order here. When $\mathcal{H} = \mathcal{C}$, by Table 2, $q \geq \lceil \log_{\frac{1}{p}}(2) \rceil \geq \log_{\frac{1}{p}}(2) = k$ *always*, and thus, on an instance of dimension n , as p increases in \mathcal{F} for successive values of k , then q increases at least that fast.

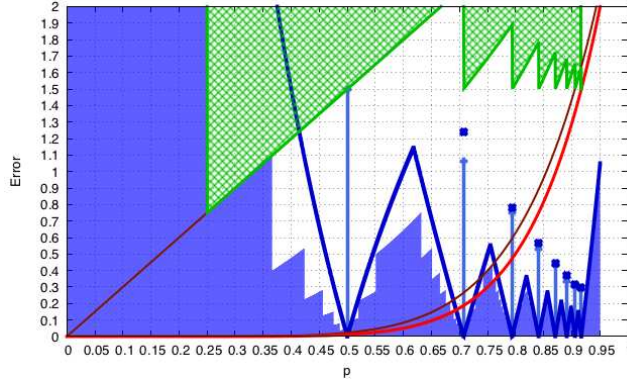


Fig. 3. $\mathcal{H} = \mathcal{C}$ and $n = 8$. Along the line $3p$, $q = 1$ for the lowest possible error at every p . Similarly, the curves $3p^{n-1}$ and $3p^n$ are also drawn. In the top part of the plot, the triangle and the region with the jigsaw frontier that are shaded indicate (p, ε) pairs where evolution needs to adapt a large input ε to a suitable smaller constant ε' ; see Section 5.1. The shaded region in the lower part of the plot, as well as the individual spikes for the members of \mathcal{F} , indicate (p, ε) pairs where swaps determine μ in (8). (When $p < \frac{1}{2}$, the critical p 's are obtained by solving numerically $2p^\zeta + 2p - 1 = 0$ for $\zeta \in \mathbb{N}^*$. For $p \geq \frac{1}{2}$, we use stepsize $\Delta p = 10^{-5}$ and for every such p we compute the turning point ε ; see Section 5.3.) Finally, the smooth boundary that is discussed in Section 5.3 is also shown; the separation point for \mathcal{U}_n is $(\frac{1}{2}, 3)$ and it is not drawn.

In one step, the first beneficial swap or addition of good variable brings one more good variable in the hypothesis and $U > \frac{1}{2}$.

When $U > \frac{1}{2}$, corresponding to Figure 1(c), then beneficial mutations are those that add potentially missing good variables, swap bad variables for good ones, or finally delete bad variables. Each swap or addition increases the number of good variables in the hypothesis and thus there can be $|c|$ of those. Further, there can be at most q removals of bad variables. After we get to the target, there are no beneficial mutations; the only neutral mutation is the target itself.

In the above process, until we reach the target, the number m of good variables that appear in h is non-decreasing. Thus there can be at most $|c|$ additions of good variables and swaps combined. Further, there can be at most $q + q = 2q$ beneficial additions or deletions of bad variables. Hence, overall, after at most $|c| + 2q \leq 3q$ steps the target will be identified and that formation is stable⁴.

The above is not necessarily true when $\mathcal{H} = \mathcal{C}_{\leq q}$. By (11), when $p \in \mathcal{F}$ with $k \geq 3$ (i.e. $p \in \mathcal{F}$ and $p \geq 2^{-1/3}$), for input ε such that $2 > \varepsilon \geq 3p^{k-1} = \frac{3}{2p}$, then $q = \lceil \log_{\frac{1}{p}}(\frac{3}{\varepsilon}) \rceil < \lfloor \log_{\frac{1}{p}}(2) \rfloor = \log_{\frac{1}{p}}(2) = k$. However, these distributions and input errors are irrelevant to our discussion as for $|c| \leq q$, $U = p^u \geq p^q \geq p^{k-1} = \frac{1}{2p} > \frac{1}{2}$.

⁴ Diochnos and Turán in [6] gave a bound of $2q$ for \mathcal{U}_n . \mathcal{U}_n is once again special, because $p = \frac{1}{2}$ is the *unique* member of \mathcal{F} where in the shrinking phase (Figure 1(c)), $U > \frac{1}{2} \Rightarrow U = 1 \Rightarrow u = 0$; that is, one needs to argue only about *specializations*

6.2 Short Initial Hypothesis and Medium Target

Medium targets make sense when $p \geq \frac{1}{2}$ and only when we perform *approximate* learning. Hence, the input error satisfies $\varepsilon \geq 3p^{n-1}$ *always*. Also, for a medium target, $q < |c| = q + j \leq q + \vartheta$, a hypothesis h is a best q -approximation if $m = q$. Then, $u = j \leq \vartheta \Rightarrow U = p^u = p^j \geq p^\vartheta = \frac{1}{2}$.

Thus, starting with a hypothesis h such that $U < \frac{1}{2}$, we have that $m \leq q - 1$. Hence, either $|h| \leq q - 1 \Rightarrow N^+ \neq \emptyset$, or $|h| = q \Rightarrow r \geq 1$. In either case, there is at least one beneficial mutation in the neighborhood. As long as $U < \frac{1}{2}$, there can be at most q beneficial additions of variables and at most q beneficial swaps. Therefore, $2q$ generations are enough to form a hypothesis with $U \geq \frac{1}{2}$.

If $U = \frac{1}{2}$, Figure 1(b) applies. $U = \frac{1}{2} \Rightarrow p \in \mathcal{F}$ for some k . Further, $U = \frac{1}{2} \Rightarrow p^u = \frac{1}{2} \Rightarrow u = \log_{\frac{1}{p}}(2) = k = \vartheta$. In other words, as k increases, we *drill deeper* and thus $\vartheta = k = u$. We distinguish cases.

- If $m = q$, then a best q -approximation is already formed; by the selection of tolerance this formation is stable. By Corollary 1, $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$. This case refers to the *longest medium* target; that is, $|c| = q + \vartheta$. For all other medium targets, $m = q$ implies $u < \vartheta$ and thus, $U = p^u > p^\vartheta = \frac{1}{2}$.
 - If $\mathcal{H} = \mathcal{C}$, as medium targets make sense only for approximate learning, by Section 5.1, $p < 2^{-\frac{1}{n-1}}$. If $\mathcal{H} = \mathcal{C}_{\leq q}$, then $p \in \mathcal{F}$ and $p < 2^{-\frac{1}{n}}$. To see this, note that $q = \lceil \log_{\frac{1}{p}}(\frac{3}{\varepsilon}) \rceil \Rightarrow q \geq 1$ for any $0 < \varepsilon < 2$. Hence, as $|c| = q + \vartheta \leq n$, it follows that $\vartheta \leq n - 1$ and as a consequence $p < 2^{-\frac{1}{n}}$. Note that $p = 2^{-\frac{1}{n-1}}$ can arise^{5,6} under \mathcal{U}_n , for $\varepsilon \geq \frac{3}{2}$ and $|c| = n = 2$.
- If $m < q$, since $u = \vartheta$, we are dealing with targets such that $|c| \in \{q + 1, \dots, q + \vartheta - 1\}$. Hence, this case can arise when $p \in \mathcal{F}$ for $k \geq 2$. Since $m < q$, either $|h| = m \Rightarrow N^+ \neq \emptyset$, or $m < |h| \leq q \Rightarrow r \geq 1 \Rightarrow N^{+-} \neq \emptyset$. In either case, in one step, evolution will proceed to the case where $U > \frac{1}{2}$.
 - If $\mathcal{H} = \mathcal{C}$, then again by Section 5.1, $p < 2^{-\frac{1}{n}}$. If $\mathcal{H} = \mathcal{C}_{\leq q}$, then $p \leq 2^{-\frac{1}{n}}$; not even the full conjunction can achieve $U = \frac{1}{2}$ for $p > 2^{-\frac{1}{n}}$.

If $U > \frac{1}{2}$, Figure 1(c) applies. Beneficial mutations either increase good variables with additions or swaps, or redundant bad variables are removed. However,

of the target. For $p < \frac{1}{2}$, Figure 1(b) never applies, Figure 1(c) is again about specializations of the target, and then we can match their $2q$ bound. However, we use $3q$ throughout for uniformity in the analysis.

⁵ This example reveals another aspect of our approach. There are cases where $q + \vartheta \geq n$, even when $\mathcal{H} = \mathcal{C}$. Then, our method is powerful enough to perform exact learning (there are no long targets). However, only an approximation of the target will be returned, satisfying $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$. On the other hand, one can improve the definitions of ϑ in Table 2 and in (11) by setting $\vartheta = \min\{n - q, \lceil \log_{1/p}(2) \rceil\}$; we did not do so for simplicity in the presentation.

⁶ Also, p can be arbitrarily close to 1. For $k \in \mathbb{N}^*$, $p = 2^{-\frac{1}{k}} \Rightarrow \vartheta = k$. Let, $\varepsilon = \frac{3}{4} \Rightarrow q = \lceil \log_{\frac{1}{p}}(4) \rceil = 2k$. Then, for $n \geq 3k$, we look at the conjunction with size $3k$.

there can be at most q removals of bad variables. Further, the set of good variables can be augmented at most q times through beneficial mutations. Thus, a best q -approximation is formed within at most $2q$ generations.

As a summary, in the above process m is non-decreasing. Thus, there can be at most q additions of good variables and swaps combined. Further, there can be at most $q + q = 2q$ beneficial additions or deletions of bad variables. Hence, after at most $q + 2q \leq 3q$ generations, a best q -approximation of a medium target will be formed. That formation is stable. By Corollary 1, $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$.

6.3 Short Initial Hypothesis and Long Target

$\vartheta = \lfloor \log_{\frac{1}{p}}(2) \rfloor \Rightarrow \vartheta > \log_{\frac{1}{p}}(2p)$. For long targets, $u \geq 1 + \vartheta \Rightarrow U = p^u \leq p^{1+\vartheta} < p^{\log_{1/p}(2)} = 1/2$. Thus, we have $U < \frac{1}{2}$, corresponding to Figure 1(a). Beneficial mutations are additions of variables or swaps. As long as $|h| < q$, then $N^+ \neq \emptyset$. Hence, after at most $2q$ generations a hypothesis of size q will be formed. By the selection of tolerance, the mutations in N^- are deleterious. Thus, evolution will wander among hypotheses of size precisely q . By Corollary 2, $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$.

6.4 Medium or Long Initial Hypothesis

With $\tilde{\mathcal{O}}(\cdot)$ we ignore polylogarithmic terms; however, we do not ignore q , as q is the frontier of our search and the maximum size of the shortest explanation. As long as $|h| > q$ the neighborhood is $N = N^- \cup \{h\}$. Tolerance is t_u from (9); every hypothesis in the neighborhood is neutral. Thus, with probability at least $1 - \delta/4$, in $\tilde{\mathcal{O}}(n)$ generations we arrive at a hypothesis of size q .

7 Sketch of Complexity Analysis for Evolution

Evolution in $\mathcal{C}_{\leq q}$. Evolution lasts for $3q$ generations. $|N| = \mathcal{O}(nq) \Rightarrow cnq^2$ queries are enough, for some $c > 0$. Table 1 computes $A = \min_{u \neq 0} \{\mathcal{A}(u)\}$; by (8), $\mu = \min\{2p^{q+\vartheta}, A\}$. By (8), tolerance is $t = t_\ell$. Requiring $\mathcal{O}(\frac{1}{t^2} \cdot \ln(n/\delta))$ samples per hypothesis tested, it follows by Hoeffding's bound and a union bound argument that the performance of each hypothesis in this phase is computed within $\varepsilon_s = t$ of its exact value with probability at least $1 - \delta/2$.

Theorem 2. *Let \mathcal{B}_n be a binomial distribution with $0 < p < 1$. Starting with a short initial hypothesis and considering hypotheses in $\mathcal{C}_{\leq q}$, the swapping algorithm, using total sample size $\tilde{\mathcal{O}}(nq^2/t_\ell^2)$, in at most $3q$ generations, will evolve a hypothesis h such that $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$, with probability at least $1 - \delta/2$.*

Evolution in $\mathcal{C}_{> q}$. With a Chernoff bound argument, selecting from the neutral set when all hypotheses are present there, for $\tilde{\mathcal{O}}(n)$ generations, then with probability at least $1 - \delta/4$, evolution will form a hypothesis of size q .

On the other hand, by (9), tolerance $t = t_u$. Requiring $\mathcal{O}(\frac{1}{t^2} \ln(n/\delta))$ samples per hypothesis tested, with a combination of the Hoeffding bound and a union

bound argument, the performance of each hypothesis is computed within $\epsilon_s = t/4$ of its exact value, with probability at least $1 - \delta/4$.

Theorem 3. *Let \mathcal{B}_n be a binomial distribution with $0 < p < 1$. Starting with a long initial hypothesis and considering hypotheses in \mathcal{C} , the swapping algorithm, using total sample size $\tilde{O}(nq^2/t_\ell^2 + n^2/t_u^2)$, in $\tilde{O}(n)$ generations, will evolve a hypothesis h such that $\text{Perf}_{\mathcal{B}_n}(h, c) > 1 - \epsilon$, with probability at least $1 - \delta$.*

8 Further Remarks

With a drilling technique, we examined a local search algorithm for the evolution of monotone conjunctions under binomial distributions. We identified differences between $\mathcal{H} = \mathcal{C}$ and $\mathcal{H} = \mathcal{C}_{\leq q}$ that had to do with the sample size as well as with the overall design and adaptation of the method. Also, on an instance of dimension n , using $\mathcal{H} = \mathcal{C}_{\leq q}$, we are able to cover a wider spectrum of distributions.

Our analysis assumed rational p but can be extended to any real value. We outline the extension; details will be given in the full version. For example, let $p > \frac{1}{2}$ and $p \in [l, r] \subset (2^{-\frac{1}{k}}, 2^{-\frac{1}{k+1}})$, for rational l and r . Setting $q = \lceil \log_{\frac{1}{r}}(3/\epsilon) \rceil$ and $\theta = \lfloor \log_{\frac{1}{r}}(2) \rfloor = k$, Corollaries 1 and 2 hold for the p of the distribution. Considering $\mathcal{H} = \mathcal{C}_{\leq q}$, we need a lower bound for t_ℓ that works for all $p \in [l, r]$. Notice that $A = \min_{u \neq 0} \{\mathcal{A}(u)\} = \min \{|1 - 2l^k|, |1 - 2l^{k+1}|, |1 - 2r^k|, |1 - 2r^{k+1}|\}$. Hence, μ in (8) is the minimum between A and $2p^{q+\theta}$. Note also that $l = r^b$ for $b \leq 1 + \frac{1}{k}$. Then, for any $p \in [l, r]$, $p^q \geq l^q = r^{bq} \geq (r\frac{\epsilon}{3})^b \geq (\frac{\epsilon}{6})^2$ and $p^\theta \geq l^\theta = r^{b\theta} \geq 2^{-b} \geq \frac{1}{4}$. Thus, $\mu = \min\{\frac{\epsilon^2}{72}, A\}$. Further, $t_\ell \geq l^{q-1}\mu(1-r) \geq \frac{\epsilon^2}{36l}\mu(1-r)$. Similar arguments can be made if $p \in \mathcal{F}$ or if $p < \frac{1}{2}$ thus treating uniformly all real values of p in an appropriate interval with rational endpoints.

Concluding, Valiant's model for evolution poses interesting questions even for concept classes that have been studied extensively in learning theory. Perhaps the most distinctive difference between evolvability on one hand and traditional optimization and EAs on the other hand, is that the evolutionary mechanism has access to fitness comparison oracles that have bounded and unbounded precision respectively. Such a distinction on comparison oracles can have independent interest, as for example in [1]. In the case of evolvability, having bounded precision on the comparisons is an artifact of sampling. By trying to understand evolution using local search when the fitness values are corrupted by noise, we have additional results that we will explore in subsequent papers. Finally, studying the method in different computation models or by restricting the parameters on real algebraic numbers might be a problem of independent interest.

Acknowledgement. The author would like to thank György Turán for fruitful discussions on earlier versions of the paper. The author would also like to thank Yanjun Qi and Elias Tsigaridas for some additional interesting discussions.

References

1. Miklós Ajtai, Vitaly Feldman, Avinatan Hassidim, and Jelani Nelson. Sorting and selection with imprecise comparisons. *ACM Transactions on Algorithms*, 12(2):19, 2016.
2. Elaine Angelino and Varun Kanade. Attribute-efficient evolvability of linear functions. In *ITCS*, pages 287–300, 2014.
3. Dana Angluin and Philip D. Laird. Learning From Noisy Examples. *Machine Learning*, 2(4):343–370, 1987.
4. Javed A. Aslam and Scott E. Decatur. Specification and Simulation of Statistical Query Algorithms for Efficiency and Noise Tolerance. *Journal of Computer and System Sciences*, 56(2):191–208, 1998.
5. Nader H. Bshouty and Vitaly Feldman. On Using Extended Statistical Queries to Avoid Membership Queries. *Journal of Machine Learning Research*, 2:359–395, 2002.
6. Dimitrios I. Diochnos and György Turán. On Evolvability: The Swapping Algorithm, Product Distributions, and Covariance. In *SAGA*, pages 74–88, 2009.
7. Stefan Droste, Thomas Jansen, and Ingo Wegener. On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science*, 276(1-2):51–81, 2002.
8. Vitaly Feldman. Evolvability from learning algorithms. In *STOC*, pages 619–628, 2008.
9. Vitaly Feldman. Robustness of Evolvability. In *COLT*, pages 277–292, 2009.
10. Vitaly Feldman. Distribution-Independent Evolvability of Linear Threshold Functions. In *COLT*, pages 253–272, 2011.
11. Vitaly Feldman. A Complete Characterization of Statistical Query Learning with Applications to Evolvability. *Journal of Computer and System Sciences*, 78(5):1444–1459, 2012.
12. Sally A. Goldman and Robert H. Sloan. Can PAC Learning Algorithms Tolerate Random Attribute Noise? *Algorithmica*, 14(1):70–84, 1995.
13. Varun Kanade. Evolution with Recombination. In *FOCS*, pages 837–846, 2011.
14. Varun Kanade, Leslie G. Valiant, and Jennifer Wortman Vaughan. Evolution with Drifting Targets. In *COLT*, pages 155–167, 2010.
15. Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. In *STOC*, pages 392–401, 1993.
16. Loizos Michael. Evolvability via the Fourier transform. *Theoretical Computer Science*, 462:88–98, 2012.
17. Balázs Szörényi. Characterizing Statistical Query Learning: Simplified Notions and Proofs. In *ALT*, pages 186–200, 2009.
18. Leslie G. Valiant. A Theory of the Learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.
19. Leslie G. Valiant. Evolvability. In *MFCS*, pages 22–43, 2007.
20. Paul Valiant. Distribution free evolvability of polynomial functions over all convex loss functions. In *ITCS*, pages 142–148, 2012.
21. Paul Valiant. Evolvability of Real Functions. *ACM Transactions on Computation Theory*, 6(3):12:1–12:19, 2014.
22. Ingo Wegener. Theoretical Aspects of Evolutionary Algorithms. In *ICALP*, pages 64–78, 2001.