On the Evolvability of Monotone Monomials with a (1+1) Evolutionary Algorithm

Dimitrios I. Diochnos
Department of Computer Science
University of Virginia
diochnos@virginia.edu

Abstract

Diochnos in (2016) showed that Valiant's swapping algorithm for monotone monomials converges efficiently under binomial distributions characterized by any $0 < p < 1$, for a sufficiently large instance dimension n. We continue the study on the evolution of monotone monomials in Valiant's framework of evolvability. In particular we prove that a (1+1) evolutionary algorithm, in a *distribution-specific* sense, evolves, with probability at least $1 - \delta$, a monomial that is ε-optimal, for every binomial distribution characterized by a real algebraic number $p \in (0, 1/3] \cup \{1/2\}$ in $\mathcal{O}\left(n^2 q \delta^{-1}\right)$ generations, where $q = \lceil \log_{1/p}(3/\varepsilon) \rceil$ is a parameter called *frontier*. The sample size is $\widetilde{\mathcal{O}}\left(n^2 q \delta^{-1} \varepsilon^{-2} \left(\min\left\{4p\varepsilon/9, 1-3p\right\}\right)^{-2}\right)$ when $0 < p < 1/3$ and $\widetilde{\mathcal{O}}\left(n^2 q \delta^{-1} \varepsilon^{-4}\right)$ when $p = 1/3$ or $p = 1/2$, where $\widetilde{\mathcal{O}}(\cdot)$ ignores poly-log factors but not the parameter q itself. Further, a slight modification of the algorithm evolves, with probability at least $1 - \delta$, an ε-optimal hypothesis, in a *distribution-independent* sense, for a class of distributions. For a real algebraic number $0 < \alpha < 3/13$, letting $k = \lceil \log_2(1/\alpha) \rceil$, this is done for every binomial distribution characterized by an *arbitrary real* $p \in [\alpha, 1/3 - 4\alpha/9] \cup \{1/3\} \cup \{1/2\}$, in $\mathcal{O}\left(n^2 q \delta^{-1}\right)$ generations using total sample size $\widetilde{\mathcal{O}}\left(6^{4k} n^2 q \delta^{-1} \varepsilon^{-4k}\right)$ where $q = \lceil \log_2(3/\varepsilon) \rceil$ this time. Both algorithms are the first in the framework of evolvability that allow partial random walks and do not necessarily follow strictly beneficial steps until an ε-optimal hypothesis is formed.

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

1 Introduction

Valiant in (2009) introduced a framework for a quantitative approach to evolution, called *evolvability*. In this framework evolution is seen through the lens of computational learning. Roughly the idea is that there is an *ideal behavior* in every environment and the feedback that the various organisms receive during evolution indicates how close their behavior is to ideal. Ultimately, evolvability wants to model and explain mechanisms that allow near-optimal behavior of organisms while exploiting realistic computational resources. Due to a result by Feldman in (2008), evolvability is equivalent to learning in the *correlational statistical query* (CSQ) model; see (Bshouty and Feldman 2002). Thus, evolvability algorithms correspond to a special type of local search learning algorithms that fall under the umbrella of the *probably approximately correct* (PAC) model of learning (Valiant 1984). In fact Valiant in (2013) gives a broad exposition of such algorithms for evolution, which he calls *ecorithms*, and discusses them within the context of computational complexity and computational learning theory as well as identifies challenges that need to be addressed by such algorithms. See also (Watson and Szathmáry 2016) for a related interesting discussion on the connections between computational learning and evolution.

A key challenge for machine learning (and more broadly, artificial intelligence) algorithms, is that of *brittleness*. That is, typically many artificial intelligence systems *fail* when tested outside of some narrow domain for which they have been designed; such discussions go back to expert systems; see, e.g., (Duda and Shortliffe 1983). John Holland argues on the use of *genetic algorithms* in order to handle brittleness in (1986). Valiant also mentions in (2013) brittleness as a challenge that needs to be addressed by ecorithms. Both Holland and Valiant argue that learning is needed in order to tackle brittleness.

Toward more robust artificial intelligence algorithms, a typical challenge in machine learning is the design of algorithms that can cope with noise. Such algorithms are more desirable as they are more realistic for practical purposes. One of the first wide experimental studies on various types of noise was conducted by Quinlan in (1986). However noise has been studied extensively within the framework of PAC learning. In particular, noise, without any assumptions on its nature and thus potentially malicious, was first discussed in the framework of PAC learning by Valiant in (1985) and subsequently Kearns and Li provided several related results in that model of noise in (1993). A broad discussion on noise is found in (Laird 1988). Sloan has an overview of different kinds of noise models in (1995) as well as discusses malicious classification noise. Random classification noise due to Angluin and Laird in (1987), led Kearns to the development of the *statistical query* model in (1998); see also (Decatur 1993; Aslam and Decatur 1998; Szörényi 2009; Simon 2014; Feldman et al. 2017) for some related results. It is within this framework that we find

the CSQ model and due to Feldman's result, evolvability.[1] Ecorithms, by the very nature of the framework of evolvability, have to deal with noisy estimates. Such estimates represent the *goodness of fit* of individuals within certain environments and are computed by a fairly limited amount of interaction that individuals have with the environment.

Ecorithms also fit well within the framework of *learning by distances* due to (Ben-David, Itai, and Kushilevitz 1995). Finally, a somewhat different approach to studying evolution has been initiated in (Livnat et al. 2008; 2010; 2014); see also (Livnat and Papadimitriou 2016).

Related Work in Computational Learning Theory

Previous work in the framework of evolvability includes (Valiant 2009; Feldman 2008; 2009; 2011; 2012; Diochnos and Turán 2009; Kanade, Valiant, and Vaughan 2010; Kanade 2011; Michael 2012; Angelino and Kanade 2014; Valiant 2014; Diochnos 2016). Regarding monomials, their evolvability follows by a result in (Feldman 2008) for every fixed distribution within $\widetilde{\mathcal{O}}(n)$ generations; $\widetilde{\mathcal{O}}(\cdot)$ ignores poly-log factors. As also pointed out by Feldman, this translation is not necessarily the most efficient or intuitive method in general; thus, there is still interest in different evolution mechanisms. The evolvability of monotone monomials under the uniform distribution $\mathcal{U}_n$ with a swapping-type algorithm was initially shown in (Valiant 2009). The analysis was simplified in (Diochnos and Turán 2009) and the result was strengthened to general monomials[2] under $\mathcal{U}_n$ including target drift in (Kanade, Valiant, and Vaughan 2010). Kanade extended Valiant's model to include recombination in (2011), where it follows that monomials are evolvable in $\mathcal{O}\left((\log(n)/\varepsilon)^2\right)$ generations. In (Diochnos 2016) it was shown that monotone monomials are evolvable under binomial distributions in $\mathcal{O}(\log(1/\varepsilon))$ generations by generalizing the swapping-type mechanism for $\mathcal{U}_n$.

In general, *distribution-specific* results are common in evolvability, as for example, in (Kanade, Valiant, and Vaughan 2010; Michael 2012; Angelino and Kanade 2014). Broadly, studying the learnability of certain concept classes on certain classes of distributions, or even specific distributions, has an independent interest and also arises in contexts outside of evolvability; e.g., (Benedek and Itai 1991; Hancock and Mansour 1991; Linial, Mansour, and Nisan 1993; Blum et al. 1994; Khardon 1994; Bshouty and Tamon 1996; Mansour and Parnas 1998; Verbeurgt 1998; Reischuk and Zeugmann 1999; Servedio 1999; Sakai and Maruoka 2000; Jackson, Klivans, and Servedio 2002; Klivans, O'Donnell, and Servedio 2004; Servedio 2004; Jackson and Servedio 2006; Sellie 2008; 2009; Simon 2009; Balcan et al. 2013; Hanneke, Kanade, and Yang 2015).

[1] There are additional models of noise that have been studied within PAC learning. For example, noise on the attributes as in (Shackelford and Volper 1988; Goldman and Sloan 1995), or the *nasty noise* model of (Bshouty, Eiron, and Kushilevitz 2002). Further, some recent work revists such noise models as relationships between machine learning algorithms and security mechanisms are explored; e.g., (Mahloujifar, Diochnos, and Mahmoody 2017).

[2] Evolving general monomials under the uniform distribution is attributed to B. Jacobson in (Kanade, Valiant, and Vaughan 2010).

Connections to Black-Box Optimization

At a very high level, evolvability is a local search method. Therefore, it makes sense to study the framework from the point of view of combinatorial optimization using stochastic local search (SLS) methods as in (Hoos and Stützle 2004), evolutionary algorithms (EAs) (Wegener 2001; Droste, Jansen, and Wegener 2002), genetic algorithms (Mitchell 1998) and genetic programming (Koza 1993).

Perhaps the key characteristic difference between evolvability and such traditional black-box optimization frameworks is the fact that 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. However, contrary to evolvability, the requirement is usually stronger under black-box optimization as one wants to identify precisely the ideal behavior; not just an ε-approximation.

Noise models have also been studied in black-box optimization methods. Droste in (2004) discusses noise in the *prior noise model* while optimizing the ONEMAX function. In the prior noise model, the response of the fitness oracle to queries may, with some small probability, return the true fitness value but for an instance that is in the neighborhood of the original input instance (e.g., in Hamming distance 1). In the *posterior noise model* the fitness value of the input instance is always computed correctly but is subsequently corrupted by noise; e.g., by adding a small random value drawn from some fixed distribution. Interestingly, it is only in the last few years that focus has shifted toward noisy fitness functions in the framework of EAs and dealing with noisy estimates has been identified as a hot topic in the field; (Friedrich and Neumann 2017). Such recent results on either noise model are (Astete-Morales, Cauwet, and Teytaud 2015; Dang and Lehre 2015; Prugel-Bennett, Rowe, and Shapiro 2015; Gießen and Kötzing 2016; Qian et al. 2017). There have also been studies on noise in SLS methods but these appear to be more scarce; e.g., (Gutjahr and Pflug 1996) discusses additive noise in the posterior model.

Finally, we note that (Kötzing, Neumann, and Spöhel 2011) is concerned with a swapping-type mechanism for linear functions similar to the mechanisms that we discuss.

Contributions

We extend the work that deals with simple, swapping-type mechanisms for the evolution of monotone monomials. During the course of evolution (learning), we manipulate hypotheses that correspond to monotone monomials in every step. For some integer q defined by the analysis, the distributions that we examine have the property that one can find optimal, or ε-optimal, approximations of any ideal (target) behavior, when restricting the search only among monomials that are composed by at most q variables. We exploit this structural property and our mechanisms search only among such solutions. Hence, we call q the *frontier* of our search. Below, with $\widetilde{\mathcal{O}}(\cdot)$ we will ignore poly-log factors in $n, 1/\varepsilon, 1/\delta$ and q, but not the frontier q itself.

Theorem 1 summarizes our distribution-specific results.

Theorem 1. *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p \in \mathbb{R}_{alg}$ such that $p \in (0, 1/3] \cup \{1/2\}$. Let $q = \lceil \log_{1/p}(3/\varepsilon) \rceil$. Then, the (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, will evolve a hypothesis h such that $\mathbf{Pr}\left(Perf_{\mathcal{B}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$. The total sample size is $\widetilde{\mathcal{O}}\left(\frac{n^2 q}{\delta \cdot \varepsilon^2 \cdot (\min\{4p\varepsilon/9, 1-3p\})^2}\right)$ when $0 < p < 1/3$ and $\widetilde{\mathcal{O}}\left(\frac{n^2 q}{\delta \cdot \varepsilon^4}\right)$ when $p = 1/3$ or $p = 1/2$.*

First of all we use a *fitness-level* technique (see, e.g., (Wegener 2003; Wegener and Witt 2005; Sudholt 2010; Lehre 2011; Corus et al. 2014)) and prove convergence on binomial distributions characterized by $p \in (0, 1/3] \cup \{1/2\}$.

Second, our algorithm by default uses the underlying distribution as a hint in order to adapt to an appropriate representation and related parameters (sample size, tolerance). To this end, most of our results refer to the parameter p of the distribution as a real algebraic number[3] and leave open the exact bit complexity of required related calculations. A different point of view would be to assume that the algorithms are endowed with appropriate values that are the integers or some appropriate fractions near the true values implied by operations on arbitrary real numbers (referring to the distribution, or the input ε and δ), and then one happens to examine the mechanism on a setup where these values correspond to. In any case, our algorithm also *converges with a uniform setup over a class of distributions*; see Theorem 2 below. This result also includes values of p that can be *transcendental*, which is not the case of Theorem 1.

Theorem 2 (Evolution with a Uniform Setup)**.** *Let $\alpha \in \mathbb{R}_{alg}$ with $0 < \alpha < 3/13$. Let $k = \lceil \log_2(1/\alpha) \rceil$. Let $q = \lceil \log_2(3/\varepsilon) \rceil$. Let $\mathcal{I} = [\alpha, 1/3 - 4\alpha/9] \cup \{1/3\} \cup \{1/2\}$. Let $p \in \mathcal{I}$. Consider a binomial distribution $\mathcal{B}_n$ over $\{0,1\}^n$ that is characterized by p. Then, the (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(6^{4k} n^2 q/(\delta \varepsilon^{4k})\right)$, will evolve a hypothesis h such that $\mathbf{Pr}\left(Perf_{\mathcal{B}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.*

Third, Feldman in (2012) showed that monomials are evolvable distribution independently with quadratic loss, while in (2011) it was shown that monomials are not CSQ learnable distribution-independently using Boolean loss; the loss function of the current paper. In this context, Theorem 2 has an added value as the result lies somewhere between distribution-specific and distribution-independent learning.

Finally, in (Kalai and Vempala 2006) it was shown that within simulated annealing it is important to allow descendents with worse performance than that of the parent. Paul Valiant asks in (2014) whether similar phenomena can arise in evolvability. While our algorithm does not fully achieve this goal, nevertheless it is the first ecorithm that allows partial random walks and does not necessarily follow strictly beneficial steps until a near optimal solution is formed.

Omitted or sketched proofs are available in the appendix.

[3] One typical encoding of real algebraic numbers is the *isolating interval representation*. In this encoding, a real algebraic number ρ can be characterized by a polynomial f with rational coefficients and an interval $[\alpha, \beta]$ such that $\alpha, \beta \in \mathbb{Q}$ and in such a way so that ρ is the unique root of f in the interval $[\alpha, \beta]$. Usually, it is assumed that f is square-free; that is, $f(\alpha) f(\beta) < 0$. Ultimately, the bit complexity results of the computational problems at hand, are given with respect to the input degree of the polynomials and the maximum bitsizes of the rationals that describe the various real algebraic numbers that are part of the input. For example, see (Diochnos, Emiris, and Tsigaridas 2009).

2 Informal Description of Evolvability

We consider Boolean functions. 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 distribution $\mathcal{D}_n$ over $\{0,1\}^n$, the performance of a hypothesis h, called the *correlation* of h and c, is,

$$\begin{aligned} \text{Perf}_{\mathcal{D}_n}(h, c) &= \sum_{x \in \{0,1\}^n} h(x) \cdot c(x) \cdot \mathbf{Pr}_{x \sim \mathcal{D}_n}(x) \quad (1) \\ &= 1 - 2 \cdot \mathbf{Pr}_{x \sim \mathcal{D}_n}(h(x) \neq c(x)) \,. \quad (2) \end{aligned}$$

Note that by (2), achieving an approximation error ε for correlation, implies misclassification error $\varepsilon/2$.

Evolution starts with some initial hypothesis and produces a sequence of hypotheses using a local-search procedure in the hypothesis space $\mathcal{H}$. Similarity between h and c in an underlying distribution $\mathcal{D}_n$ is measured by the *empirical performance* $\text{Perf}_{\mathcal{D}_n}(h, c, S)$ which is evaluated approximately by drawing a random sample S (of size $|S|$) and computing

$$\text{Perf}_{\mathcal{D}_n}(h, c, S) = \frac{1}{|S|} \sum_{x \in S} h(x) \cdot c(x) \,. \quad (3)$$

The mutator function is responsible for generating the neighborhood $N(h)$ and selecting one hypothesis from $N(h)$ as the output for the next generation. For each hypothesis $h' \in N(h)$, the mutator first computes an empirical value of $\nu(h') = \text{Perf}_{\mathcal{D}_n}(h', c, S)$ and also associates each hypothesis h' with a weight $\mathbf{Pr}_N(h, h')$. Then, based on the value of a real constant t called *tolerance* we obtain,

$$\begin{cases} \text{Bene} &= \{h' \in N(h) \mid \nu(h') > \nu(h) + t\} \\ \text{Neut} &= \{h' \in N(h) \mid \nu(h') \geq \nu(h) - t\} \setminus \text{Bene} \end{cases}$$

The output of the mutator function is based on the rule[4]:

- if Bene $\neq \emptyset$ then output $h_1 \in$ Bene with probability $\mathbf{Pr}_N(h, h_1) / \sum_{h' \in \text{Bene}} \mathbf{Pr}_N(h, h')$,
- if Bene $= \emptyset$ then output $h_1 \in$ Neut with probability $\mathbf{Pr}_N(h, h_1) / \sum_{h' \in \text{Neut}} \mathbf{Pr}_N(h, h')$.

Ultimately, the goal of the evolution is to produce, within a realistic time period (i.e., within $poly(1/\varepsilon, 1/\delta, n)$ generations), a hypothesis $h \in \mathcal{H}$ such that

$$\mathbf{Pr}\left(\text{Perf}_{\mathcal{D}_n}(h, c) < \text{Perf}_{\mathcal{D}_n}(c, c) - \varepsilon\right) < \delta \,. \quad (4)$$

3 Definition of Evolvability

We use the definitions of performance and empirical performance as described in (1) and (3) respectively. Below we draw the definitions from (Valiant 2009); however, we include the failure probability δ explicitly.

[4] Additional rules can be defined and are discussed in (Valiant 2009); they are however, beyond the scope of this paper.

Definition 1. *For a polynomial $p(\cdot,\cdot,\cdot)$ 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), $\lceil 1/\varepsilon \rceil$, $\lceil 1/\delta \rceil$ and a representation $r \in R_n$ act as follows: M_1 outputs all the members of a set $Neigh_N(r,\varepsilon,\delta) \subseteq R_n$, that contains r and may depend on random coin tosses of M_1, and has size at most $p(n, 1/\varepsilon, 1/\delta)$. If M_2 is then run on this output of M_1, it in turn outputs one member of $Neigh_N(r,\varepsilon,\delta)$, with member r_1 being output with a probability $\mathbf{Pr}_N(r, r_1) \geq 1/p(n, 1/\varepsilon, 1/\delta)$.*

Definition 2. *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 $p(n, 1/\varepsilon, 1/\delta)$-neighborhood N on R, a distribution $\mathcal{D}$, a representation $r \in R_n$ and a real number t, the mutator $Mu(f, p(n, 1/\varepsilon, 1/\delta), R, N, \mathcal{D}, s, r, t)$ is a random variable that on input $r \in R_n$ takes a value $r_1 \in R_n$ determined as follows: For each $r_1 \in Neigh_N(r,\varepsilon,\delta)$ it first computes an empirical value of $\nu(r_1) = Perf_{\mathcal{D}_n}(r_1, f, s)$. 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 $\mathbf{Pr}_N(r, r_1) / \sum_{r_1 \in Bene} \mathbf{Pr}_N(r, r_1)$. Otherwise (Bene $= \emptyset$), output an $r_1 \in Neut$, the probability of a specific r_1 being $\mathbf{Pr}_N(r, r_1) / \sum_{r_1 \in Neut} \mathbf{Pr}_N(r, r_1)$.*

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

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

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

Definition 5. *For polynomials $p(n, 1/\varepsilon, 1/\delta)$, $s(n, 1/\varepsilon, 1/\delta)$, $t_\ell(1/n, \varepsilon, \delta)$ and $t_u(1/n, \varepsilon, \delta)$, a representation class R and $p(n, 1/\varepsilon, 1/\delta)$-neighborhood N on R, the class $\mathcal{C}$ is (t_ℓ, t_u)-evolvable by $(p(n, 1/\varepsilon, 1/\delta), R, N, s(n, 1/\varepsilon, 1/\delta))$ 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 $r_0, r_1, r_2, \ldots$, where $r_i \leftarrow Evolve(f, p(n, 1/\varepsilon, 1/\delta), R, N, \mathcal{D}_n, s(n, 1/\varepsilon, 1/\delta), r_{i-1}, T(r_{i-1}, n, \varepsilon))$, will have $Perf_{\mathcal{D}_n}(r_{g(n,1/\varepsilon,1/\delta)}, f) \geq 1 - \varepsilon$.*

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

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

4 Preliminaries

Given a set of Boolean variables $x_1, \ldots, x_n$, we assume that there is an unknown target $c \in \mathcal{C}_n$, a monotone monomial (conjunction) of some of these variables. Let $\mathcal{C}_n$ be the concept class of all possible monotone monomials in their natural representation. For a threshold q, let $\mathcal{C}_n^{\leq q}$ be the set of monomials from $\mathcal{C}_n$ that contain at most q variables. Further, let $\mathcal{C}_n^{>q} = \mathcal{C}_n \setminus \mathcal{C}_n^{\leq q}$. Our hypothesis space will be $\mathcal{H} = \mathcal{C}_n^{\leq q}$.

By Definition 2, the neighborhood N is split in 3 parts. There are *beneficial*, *neutral*, and *deleterious* mutations. Thus, we need an oracle for computing,

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

and hence for a given tolerance t, determine the set where $h' \in N$ lies. In other words, we imply two kinds of oracles: *a fitness oracle* that returns a compact representation of the value of the fitness function with a call to $\text{Perf}_{\mathcal{D}_n}(\cdot,\cdot)$ and *a fitness comparison oracle* that compares two such values.

Definition 8 (Bounded-/Unbounded- Precision Evolution)**.** *The unbounded-precision model occurs for $t_\ell = 0$ in the definitions of evolvability. The bounded-precision model occurs for $t_\ell > 0$. Further, the tolerances t_ℓ and t_u may not be polynomially related.*

The bounded precision model allows intermediate setups between black-box optimization (unbounded-precision) and evolvability, where in evolvability one can determine the sign of Δ if the two fitness values differ significantly; that is, Δ is $poly(1/n, \varepsilon, \delta)$. Bounded-precision oracles are of interest in other domains as well; see e.g., (Ajtai et al. 2016). Now let,

$$c = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{k=1}^{u} y_k \quad \text{and} \quad h = \bigwedge_{i=1}^{m} x_i \wedge \bigwedge_{\ell=1}^{r} w_\ell \,. \tag{5}$$

The x's are *mutual* variables, the y's are called *undiscovered* or *missing*, and the w's are called *redundant* or *wrong*. Variables in the target c are called *good*, otherwise *bad*. With $|h|$ we denote the *size* (or *length*) of a monomial h; the number of variables that it contains. Given a size q and an *extension* ϑ, a hypothesis h is *short* when $|h| \leq q$, *medium* when $q < |h| \leq q + \vartheta$ and *long* when $|h| > q + \vartheta$.

Definition 9 (Best q-Approximation)**.** *A hypothesis h is called a best q-approximation of c if $|h| \leq q$ and $\forall h' \neq h, |h'| \leq q : Perf_{\mathcal{D}_n}(h', c) \leq Perf_{\mathcal{D}_n}(h, c)$.*

We represent monotone monomials as sets of indices; the indices correspond to the variables that appear in the monomials, in some fixed ordering. However, it is also convenient to think that the representation of a hypothesis h is a binary string of length n, where n is the dimension of the instances, and a 1 in the i-th position indicates that the i-th variable appears in h. For two binary strings σ and σ' of length n, let $d_n(\sigma, \sigma') = \sum_{i=1}^{n} |\sigma_i - \sigma'_i|$ be their *Hamming distance*,

where σ_i and σ'_i is their i-th bit respectively. Thinking of the representation of hypotheses as bitstrings of length n, for any $h_1, h_2 \in \mathcal{H}$ we will be able to compute $d_n(h_1, h_2)$ even if technically we are passing as parameters sets of indices.

We will consider product distributions in which *each variable follows the same Bernoulli p distribution*. In a truth assignment drawn from this distribution, the number of 1's are binomially distributed with parameters p and n and we thus call such distributions, *binomial*. Hence, on an instance of dimension n, a *binomial distribution* over $\{0,1\}^n$ is specified by the probability p of setting each variable x_i equal to 1. A truth assignment $(a_1, \ldots, a_n) \in \{0,1\}^n$ 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. The uniform distribution $\mathcal{U}_n$ is a a binomial distribution where $p = 1/2$. Consider a target c and a hypothesis h as in (5). For a binomial distribution with parameter p, (1) reduces to,

$$\text{Perf}_{\mathcal{B}_n}(h, c) = 1 - 2p^{m+r} - 2p^{m+u} + 4p^{m+r+u}. \quad (6)$$

We will use $U = p^u$ for the weight of the subcube of the undiscovered variables.

For a target c, we partition the hypothesis space in three parts; that is, we set $\mathcal{H} = \mathcal{H}_{<1/2} \cup \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$. $\mathcal{H}_{<1/2}$ refers to hypotheses for which $U < 1/2$, $\mathcal{H}_{1/2}$ refers to hypotheses for which $U = 1/2$ and $\mathcal{H}_{>1/2}$ refers to hypotheses for which $U > 1/2$. Under a distribution $\mathcal{D}_n$, for a target c, and two sets of hypotheses Φ and Ψ, we write $\Phi \not\rightsquigarrow \Psi$ to indicate that

$$(\forall h_1 \in \Phi)(\forall h_2 \in \Psi)[\text{Perf}_{\mathcal{D}_n}(h_1, c) > \text{Perf}_{\mathcal{D}_n}(h_2, c)].$$

With $\log_{1/p}(x)$ we denote the logarithm of x in base $1/p$, where $0 < p < 1$. With $\mathbb{Q}$ and $\mathbb{R}_{alg}$ we denote respectively the fields of rational and real algebraic numbers. H_j denotes the j-th harmonic number; that is, $H_j = \sum_{i=1}^{j} 1/i$.

Notions of Monotonicity

Feldman first discussed monotonicity. We draw the following notions from (Kanade, Valiant, and Vaughan 2010).

Definition 10 (Monotonic Evolution)**.** *An evolution algorithm $\mathcal{A}$ monotonically evolves a class $\mathcal{C}_n$ over a distribution $\mathcal{D}$ if $\mathcal{A}$ evolves $\mathcal{C}_n$ over $\mathcal{D}$ and with probability at least $1-\delta$, for all $i \leq g\left(n, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$, $\text{Perf}_{\mathcal{D}}(r_i, c) \geq \text{Perf}_{\mathcal{D}}(r_0, c)$, where $g\left(n, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$ and $r_0, r_1, \cdots$ are defined as in Section 3.*

Definition 11 (Strict Monotonic Evolution)**.** *An evolution algorithm $\mathcal{A}$ strictly monotonically evolves a class $\mathcal{C}_n$ over a distribution $\mathcal{D}$ if $\mathcal{A}$ evolves $\mathcal{C}_n$ over $\mathcal{D}$ and, for a polynomial m, with probability at least $1-\delta$, for all $i \leq g\left(n, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$, either $\text{Perf}_{\mathcal{D}}(r_{i-1}, c) \geq 1-\varepsilon$, or $\text{Perf}_{\mathcal{D}}(r_i, c) \geq \text{Perf}_{\mathcal{D}}(r_{i-1}, c) + 1/m(n, \frac{1}{\varepsilon}, \frac{1}{\delta})$, where $g\left(n, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$ and $r_0, r_1, \cdots$ are defined as in Section 3.*

The Algorithm

Algorithm 1 presents the mutator function. For a current hypothesis h, the evolutionary operator `Mutate` generates one candidate hypothesis h' by first initializing h' to h and then flipping each bit with probability $1/n$. h' is accepted as a viable neighbor only if $|h'| \leq q$. `USelect` picks uniformly at random among the elements of the set passed as argument.

Algorithm 1: Mutator function of the (1+1) EA for binomial distributions with $p \in (0, 1/3] \cup \{1/2\}$.

Input: dimension n, $p \in (0, 1/3] \cup \{1/2\}$, $\delta \in (0,1)$, $\varepsilon \in (0,2)$, $h \in \mathcal{H} = \mathcal{C}_n^{\leq q}$
Output: a new hypothesis

$q \leftarrow \lceil \log_{1/p}(3/\varepsilon) \rceil$;
$h' \leftarrow$ `Mutate(h)`;
if $|h'| \leq q$ **then** $N \leftarrow \{h'\}$; **else return** h;
if $0 < p < 1/3$ **then**
  $t \leftarrow p^{q-1} \cdot \min\{4p^q/3, 1-3p\}$; $\delta_s \leftarrow \delta^2/(126en^2q)$;
else if $p = 1/3$ **then**
  $t \leftarrow 2 \cdot 3^{-1-2q}$; $\delta_s \leftarrow \delta^2/(126en^2q)$;
else
  $t \leftarrow 2^{-2q}$; $\delta_s \leftarrow \delta^2/(142en^2q)$; `/* p = 1/2 */`
$\epsilon_s \leftarrow t/2$;
$\nu_h \leftarrow$ `Perf(p, h, ε_s, δ_s)`; $\nu_{h'} \leftarrow$ `Perf(p, h', ε_s, δ_s)`
if $\nu_{h'} > \nu_h + t$ **then return** h';
else if $\nu_{h'} \geq \nu_h - t$ **then return** `USelect({h} ∪ {h'})`;
else return h;

5 Foundations for Evolvability

For a current hypothesis h, of particular interest will be the hypotheses that have: Hamming distance 1, or Hamming distance 2 and same size, with respect to h. The set of hypotheses that is obtained from h by *adding* a variable is denoted as N^+. The set of hypotheses that is obtained from h by *removing* a variable is denoted as N^-. The set of hypotheses that is obtained from h by *swapping* a variable with another one is denoted as $N^\pm$. 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 $N^\pm = \{x_3 \wedge x_2, x_1 \wedge x_3\}$.

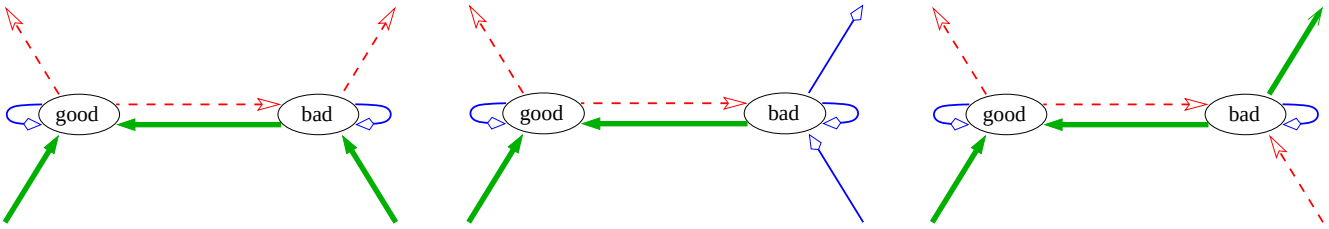
Even if we have to take into account mutations of h that are wilder than those described by N^+, N^- and $N^\pm$, nevertheless, such simple mutations are important for proving Theorem 3, Lemmata 1 and 2, as well as for some arguments related to the convergence of the (1+1) EA. Figure 1 presents the sign of the difference Δ for such mutations that give rise to a hypothesis in N^+, N^- or $N^\pm$. In particular Figure 1 is obtained by the following quantities.

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 $h' \in N^\pm$ 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 $\Delta = 4p^{|h|+u-1}(1-p)$.

While the sign of an arrow in Figure 1 may be fully determined, it is the value of the tolerance t that defines the sets Bene and Neut that guide the search. A critical quantity in the above calculations is $\mathcal{A}(u) = |1-2p^u|$, $u \in$



(a) $U < 1/2$ (b) $U = 1/2$ (c) $U > 1/2$

Figure 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. Let $\Delta = \text{Perf}_{\mathcal{B}_n}(h', c) - \text{Perf}_{\mathcal{B}_n}(h, c)$. Thick solid lines indicate $\Delta > 0$. Simple lines indicate $\Delta = 0$. Dashed lines indicate $\Delta < 0$. Figure 1(a) holds when $U < 1/2$; Figure 1(b) when $U = 1/2$; Figure 1(c) when $U > 1/2$.

$\{0, \ldots, n\}$; its minimum non-zero value $\min_{\neq 0}\{\mathcal{A}(u)\}$ is discussed in (Diochnos 2016) for every $p \in (0, 1)$. However, in our context, $p \in (0, 1/3] \Rightarrow \min_{\neq 0}\{\mathcal{A}(u)\} = 1 - 2p \geq 1/3$, while $p = 1/2 \Rightarrow \min_{\neq 0}\{\mathcal{A}(u)\} = 1/2$.

Theorem 3 ((Diochnos 2016)). *Let $\mathcal{B}_n$ be a binomial distribution with parameter p. The best q-approximation of a target c is c if $|c| \leq q$, or any hypothesis formed by q good variables if $|c| > q$.*

Lemmata 1 and 2 below are taken from (Diochnos 2016). Lemma 1 is relevant in our context only under $\mathcal{U}_n$, where $\vartheta = 1$. While Algorithm 1 does not mention ϑ, it is however taken into account under $\mathcal{U}_n$ in Lemmata 4, 5, 8 and 9, in order to compute the tolerance and the sample size.

Lemma 1 (Medium Targets). *Let $\mathcal{B}_n$ be a binomial distribution, h a hypothesis and c be the target. Then, $q \geq \log_{\frac{1}{p}}\left(\frac{3}{\varepsilon}\right), \vartheta \geq 0, |h| = q < |c| \leq q + \vartheta$, all variables in h are good $\Rightarrow$ Perf$_{\mathcal{B}_n}(h, c) > 1 - 2\varepsilon/3$.*

Lemma 2 (Long Targets). *Let $\mathcal{B}_n$ be a binomial distribution, h a hypothesis and c be the target. Then, $q \geq \log_{\frac{1}{p}}\left(\frac{3}{\varepsilon}\right)$, $\vartheta \geq \log_{\frac{1}{p}}(2p)$, $|h| \geq q, |c| > q + \vartheta \Rightarrow$ Perf$_{\mathcal{B}_n}(h, c) > 1 - \varepsilon$.*

6 Analysis of the (1+1) EA

We start with some coarse characterizations under $\mathcal{U}_n$ for the three sets into which the hypothesis space $\mathcal{H} = \mathcal{H}_{<1/2} \cup \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$ is partitioned (corresponding to Figures 1(a), 1(b) and 1(c)). Lemma 3 further refines the hypotheses in $\mathcal{H}_{<1/2}$ and expresses the fact that among such hypotheses, the larger the size of the hypothesis the better its performance. For binomial distributions with $p \in (0, 1/3]$, we provide equivalent results in Appendix C. In such cases, $\mathcal{H}_{1/2} = \emptyset$ and thus we characterize only $\mathcal{H}_{<1/2}$ and $\mathcal{H}_{>1/2}$. In particular, Lemmata 11 and 12 are equivalent to Lemma 3 when $0 < p < 1/3$ and $p = 1/3$ respectively. Lemma 13 is equivalent to Lemma 6 when $0 < p \leq 1/3$. These lemmata provide quantitative bounds that are $poly(1/\varepsilon)$ for separating hypotheses that belong to different groups. Further, they imply different phases that need to be considered in order to argue for the convergence of evolution. Our arguments for proving convergence are the same for every $p \in (0, 1/3]$. Under $\mathcal{U}_n$ the arguments for the convergence are the same for the phases that are shared with $p \in (0, 1/3]$, but we need to argue about an additional phase since we may encounter hypotheses that correspond to Figure 1(b). As the quantitative bounds are different when $p \in (0, 1/3)$, $p = 1/3$ and $p = 1/2$, we prove the complexity bound only for $\mathcal{U}_n$ and give the exact results for $p \in (0, 1/3]$ in Appendix C.[5]

Lemma 3 (Longer is Better under $\mathcal{H}_{<1/2}$). *Let $h, h' \in \mathcal{H}_{<1/2}$ such that $|h'| < |h| \leq q$. Then, under the uniform distribution $\mathcal{U}_n$, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{1-2q}$.*

Lemma 4 ($\mathcal{H}_{1/2} \not\rightsquigarrow \mathcal{H}_{<1/2}$). *Under $\mathcal{U}_n$, let $h \in \mathcal{H}_{1/2}$ and $h' \in \mathcal{H}_{<1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{-q}$.*

Lemma 5 ($\mathcal{H}_{>1/2} \not\rightsquigarrow \mathcal{H}_{1/2}$). *Under $\mathcal{U}_n$, let $h \in \mathcal{H}_{>1/2}$ and $h' \in \mathcal{H}_{1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{1-q}$.*

Lemma 6 ($\mathcal{H}_{>1/2} \not\rightsquigarrow \mathcal{H}_{<1/2}$ under $\mathcal{U}_n$). *Under the uniform distribution $\mathcal{U}_n$, let $h \in \mathcal{H}_{>1/2}$ and $h' \in \mathcal{H}_{<1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 3 \cdot 2^{-q}$.*

Convergence

We now prove the lemmata that signify the different phases of the algorithm in every case of $p \in (0, 1/3] \cup \{1/2\}$. We use the terms *generalization* and *specialization* as in Tom Mitchell's framework of *version spaces* (Mitchell 1977; 1997). That is, a Boolean function f is a *generalization* (resp., *specialization*) of a Boolean function f' iff the set of satisfying truth assignments for f is a *superset* (resp., *subset*) of the set of satisfying truth assignments for f'.

Lemma 7 (Long Targets). *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p \in \mathbb{R}_{alg}$ such that $p \in (0, 1/3] \cup \{1/2\}$. Starting with a short hypothesis h_0, the (1+1) EA, within $\lceil 16enq/\delta \rceil$ generations, assuming that the performance of each hypothesis generated is estimated within $\epsilon_s = t/2$ of its true value, with probability at least $1 - \delta/16$, will evolve a hypothesis h such that, either $h \in \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$, or it is the case that $h \in \mathcal{H}_{<1/2}$ and $|h| = q$.*

[5] We note that one can also work under $\mathcal{H} = \mathcal{C}_n$ and in fact prove some slightly more general statements. In particular, Lemma 4 would require only h' to have size at most q and Lemma 5 would require only h to have size at most q. On the other hand, Lemma 6 would still require both h and h' to have size at most q.

Proof. Suffices to prove the lemma for an initial hypothesis $h_0 \in \mathcal{H}_{<1/2}$, otherwise the statement is trivial as $h_0 \in \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$ from the very beginning.

As long as $h \in \mathcal{H}_{<1/2}$ and $|h| < q$, the probability of adding at least one good or bad variable to the hypothesis is lower bounded by the probability of the event of not touching the variables that appear in h and introducing precisely one variable. In other words, for this probability it holds $(1-1/n)^{n-1} \cdot \frac{1}{n} \geq \frac{1}{e} \cdot \frac{1}{n}$. Thus, the expected time to introduce at least one variable is at most en. Conditioning on $h \in \mathcal{H}_{<1/2}$ throughout, the expected time to form a hypothesis of size precisely q is at most enq. We now apply Markov's inequality with failure probability $\delta/16$. □

Lemma 8 (Specialization under $p \in (0, 1/3]$ or Best Approximation under $\mathcal{U}_n$)**.** *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p \in \mathbb{R}_{alg}$; $p \in (0, 1/3] \cup \{1/2\}$. Unless the target is long, starting with a hypothesis h_0 such that $h_0 \in \mathcal{H}_{<1/2}$ and $|h_0| = q$, the (1+1) EA, within $\lceil 16en^2q/\delta \rceil$ generations, assuming that the performance of each hypothesis generated is estimated within $\epsilon_s = t/2$ of its true value, with probability at least $1 - \delta/16$, will evolve a hypothesis $h \in \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$.*

Proof. Due to Lemmata 11, 12 and 3, as long as $h \in \mathcal{H}_{<1/2}$, any mutations that form a hypothesis $h' \in \mathcal{H}_{<1/2}$ such that $|h'| < |h|$ will cause a noticeable decrease in performance by the selection of tolerance. On the other hand, swapping a bad with a good variable provides a noticeable increase in performance, again due to the selection of tolerance, and occurs with probability at least $(1-1/n)^{n-2} \cdot \frac{1}{n} \cdot \frac{1}{n} > \frac{1}{e} \cdot \frac{1}{n^2}$. Conditioning on $h \in \mathcal{H}_{<1/2}$, up to q such swaps will occur within expected time not more than en^2q. However, for short and medium targets, this implies enough swaps that lead to $u = 0$ when $p \in (0, 1/3]$ or $u = 1$ when $p = 1/2$ and thus in either case a hypothesis h is formed such that $h \in \mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$. We now apply Markov's inequality with failure probability $\delta/16$.

Note that the final hypothesis h belongs to $\mathcal{H}_{1/2}$ only under $\mathcal{U}_n$; if it is the case that evolution takes place under a binomial distribution with parameter $p \in (0, 1/3]$, then h belongs to $\mathcal{H}_{>1/2}$, which in fact implies that $u = 0$. □

Lemma 9 (Maintain Best q-Approximation of Medium Targets or Create a Specialization of Short Targets under $\mathcal{U}_n$)**.** *Under the uniform distribution $\mathcal{U}_n$, let $h_0 \in \mathcal{H}_{1/2}$ such that $|h_0| \leq q$. The (1+1) EA, within $\lceil 16en^2/\delta \rceil$ generations, assuming that the performance of each hypothesis generated is estimated within $\epsilon_s = t/2$ of its true value, with probability at least $1 - \delta/16$, will, either maintain a best q-approximation for a target of size $q+1$, or evolve a hypothesis $h \in \mathcal{H}_{>1/2}$.*

Proof. We have $u = 1$, corresponding to Figure 1(b).

First of all, if the target has size $|c| = q + 1$, then, since $u = 1$, we have $|h| = m = q$. That is, h is a best q-approximation of c. Due to Lemma 4, h is noticeably better (by an amount of at least 2^{-q}) compared to any other hypothesis of size at most q that is not a best q-approximation (as for any such other hypothesis h' it holds $h' \in \mathcal{H}_{<1/2}$). Thus, the formation is stable and h can only mutate to other hypotheses that are also best q-approximations.

In the other case, $|c| \leq q$. (Targets with size $|c| > q + 1$ imply $u \geq 2 \Rightarrow h \in \mathcal{H}_{<1/2}$ contradicting the premise of the lemma that $h \in \mathcal{H}_{1/2}$.) Neutral mutations do not affect the number of good variables that already appear in h. Also, any beneficial mutation results in the introduction of the last good variable that is missing from the hypothesis. Due to Lemma 5 introducing the last missing good variable results in noticeable increase in performance. Further, by Lemma 4, any hypothesis that has fewer good variables than h results in a noticeable decrease in performance. Thus, neutral mutations in this phase will generate hypotheses with sizes in $\{m, m+1, \ldots, q\} = \{|c| - 1, |c|, \ldots, q\}$ depending on the number of present bad variables in these hypotheses. In every generation the last good variable is introduced into the hypothesis due to either a beneficial swap or a beneficial addition of the last good variable. When $|h| < q$, the last good variable is added with probability at least $(1-1/n)^{n-1} \cdot \frac{1}{n} \geq \frac{1}{en}$. When $|h| = q$ the probability of a beneficial swap is at least $(1-1/n)^{n-2} \cdot \frac{1}{n} \cdot \frac{1}{n} \geq \frac{1}{en^2}$. Hence, regardless of the size of h, the probability that the last good variable is introduced into the hypothesis within one generation, is at least $\frac{1}{en^2}$. We now apply Markov's inequality with failure probability $\delta/16$. □

Lemma 10 (Identification of Short Targets)**.** *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p \in \mathbb{R}_{alg}$ such that $p \in (0, 1/3] \cup \{1/2\}$. For an initial hypothesis $h_0 \in \mathcal{H}_{>1/2}$ such that $|h_0| \leq q$, the (1+1) EA, within $\lceil 16enq/\delta \rceil$ generations, assuming that the performance of each hypothesis generated is estimated within $\epsilon_s = t/2$ of its true value, with probability at least $1 - \delta/16$, will evolve to the target c.*

Proof. $h_0 \in \mathcal{H}_{>1/2} \Rightarrow u = 0$, corresponding to Figure 1(c). Further, $m = |c| \leq q$.

Any hypothesis that is missing $u \geq 1$ variables has noticeably smaller performance compared to any hypothesis that is a specialization of the target ($u = 0$). Under $\mathcal{U}_n$ Lemmata 5 and 6 provide the performance gap between such hypotheses, while for binomial distributions $\mathcal{B}_n$ with $p \in (0, 1/3]$ the gap in performance is provided by Lemma 13.

If the starting hypothesis h_0 contains redundant bad variables, then beneficial mutations are those that remove one or more of those in one step. Such a beneficial removal of one bad variable will occur within one generation with probability at least $(1-1/n)^{n-1} \cdot \frac{1}{n} \geq \frac{1}{en}$ and will be identified as such. Since h_0 contains not more than q bad variables, by linearity of expectation, it follows that within enq generations all the redundant bad variables are expected to be removed from h_0 thus leading to the target c. In this formation the only neutral mutation is the target itself. We now apply Markov's inequality with failure probability $\delta/16$. □

Complexity

Lemmata 7, 8, 9 and 10 are used in order to glue together the different phases until the convergence.

Theorem 4 (Evolution in $\mathcal{C}_n^{\leq q}$ under $\mathcal{U}_n$)**.** *Let $q = \lceil \log_2(3/\varepsilon) \rceil$. Under the uniform distribution $\mathcal{U}_n$, the (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(n^2 q/(\delta\varepsilon^4)\right)$, will evolve a hypothesis h such that* $\mathbf{Pr}\left(\mathit{Perf}_{\mathcal{U}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.

Proof Sketch. Evolution is split naturally into four different phases as it is dictated by Lemmata 7, 8, 9 and 10. The key observation is that tolerance is set low enough and the sample size needed for estimating the performance of each individual is sufficiently large so that the gaps implied by all these lemmata can be realized by the evolutionary mechanism throughout the evolution. Hence, evolution proceeds monotonically along the phases implied by these lemmata. It turns out that it is sufficient to set the tolerance t to be,

$$t = 2^{-2q}\,.$$

Due to Lemmata 7, 8, 9 and 10, evolution lasts not more than $53en^2q/\delta$ generations except with probability at most $\delta/4$. By a Hoeffding bound (Proposition 1) argument, the number of samples needed for estimating the performance of each hypothesis generated along the evolution within $\epsilon_s = t/2$ of its true value, is $\mathcal{O}\left(\frac{1}{t^2} \cdot \ln\left(n^2q/\delta\right)\right)$ except with probability at most $3\delta/4$. Thus from a union bound, except with probability at most δ, evolution will last at most $53en^2q/\delta$ generations and moreover the performance of each hypothesis generated will be estimated within $\epsilon_s = t/2$ of its true value by using sample size $\mathcal{O}\left(\frac{1}{t^2} \cdot \ln\left(n^2q/\delta\right)\right)$.

Finally, $q = \lceil \log_2(3/\varepsilon) \rceil \Rightarrow 2^{-q} \geq \varepsilon/6$ and hence $t = 2^{-2q} \geq (\varepsilon/6)^2$. The total sample size follows. □

7 Uniform Setup for a Class of Distributions

Algorithm 2 provides a uniform setup so that the (1+1) EA can converge to an ε-optimal hypothesis *regardless of the value of p that characterizes the underlying distribution, as long as p belongs to a specific family*. The idea is that, on one hand Lemmata 1 and 2 hold for any sufficiently large frontier q and on the other hand we will set the tolerance t so that it lower bounds any tolerance that was used by Algorithm 1 when the exact value of p was used and p belongs to the particular family. The algorithm can now also operate on distributions where p is transcendental since p is not even part of the input.

Theorem 2 (Evolution with a Uniform Setup)**.** *Let $\alpha \in \mathbb{R}_{alg}$ with $0 < \alpha < 3/13$. Let $k = \lceil \log_2(1/\alpha) \rceil$. Let $q = \lceil \log_2(3/\varepsilon) \rceil$. Let $\mathcal{I} = [\alpha, 1/3 - 4\alpha/9] \cup \{1/3\} \cup \{1/2\}$. Let $p \in \mathcal{I}$. Consider a binomial distribution $\mathcal{B}_n$ over $\{0,1\}^n$ that is characterized by p. Then, the (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(6^{4k} n^2 q/(\delta\varepsilon^{4k})\right)$, will evolve a hypothesis h such that* $\mathbf{Pr}\left(\mathit{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.

Proof. Note that for any $0 < \varepsilon < 2$, we have $q \geq 1$.

Case $p \in \mathcal{I}_1 = [\alpha, 1/3 - 4\alpha/9]$**.** We first use the fact that $4\alpha/3$ is a lower bound for the quantity $1 - 3p$ for any $p \in \mathcal{J} = [3/13, 1/3 - 4\alpha/9]$. Thus, for any $p \in \mathcal{I}_1$ we have $\min\{4p^q/3, 1 - 3p\} \geq \min\{4p^q/3, 4\alpha/3\} \geq \min\{4\alpha^q/3, 4\alpha/3\} = 4\alpha^q/3$. It follows that for any $p \in \mathcal{I}_1$ we have $p^{q-1} \cdot \min\{4p^q/3, 1 - 3p\} \geq \alpha^{q-1} \cdot (4\alpha^q/3) > 52\alpha^{2q}/9$, where in the last inequality we used the fact that $\alpha < 3/13$. Further, $2^{-k} \leq \alpha \Rightarrow \alpha^q \geq 2^{-kq} = 2^{-k\lceil \log_2(3/\varepsilon) \rceil} \geq \left(2^{-1-\log_2(3/\varepsilon)}\right)^k = (\varepsilon/6)^k$. Therefore, it suffices to set the tolerance t so that,

$$t \leq \frac{52}{9} \cdot (\varepsilon/6)^{2k} \leq \frac{52}{9} \cdot \alpha^{2q}\,. \tag{7}$$

Case $p = 1/3$**.** We observe that $\frac{2}{3} \cdot 3^{-2q} \geq \frac{2}{3} \cdot \left(3^{-1-\log_2(3/\varepsilon)}\right) = \frac{2}{27} \cdot \left(3^{-\log_3(3/\varepsilon)}\right)^{2/\log_3(2)} = \frac{2}{27} \cdot (\varepsilon/3)^{2/\log_3(2)} \geq \frac{2}{27} \cdot (\varepsilon/3)^{3.17}$. Hence, it suffices to set,

$$t \leq \frac{2}{27} \cdot (\varepsilon/3)^{3.17} \leq \frac{2}{3} \cdot 3^{-2q}\,. \tag{8}$$

Case $p = 1/2$**.** Since $2^{-2q} \geq (\varepsilon/6)^2$ it suffices to set,

$$t \leq (\varepsilon/6)^2 \leq 2^{-2q}\,. \tag{9}$$

Now, using the fact that $\varepsilon < 2$ and $k \geq 3$, the first observation is, $\frac{52}{9} \cdot (\varepsilon/6)^{2k} \leq \frac{52}{9} \cdot (\varepsilon/6)^6 = \frac{52}{9} \cdot (\varepsilon/6)^4 \cdot (\varepsilon/6)^2 \leq \frac{52}{9} \cdot (2/6)^4 \cdot (\varepsilon/6)^2 \leq \frac{52}{9^3} \cdot (\varepsilon/6)^2 < \varepsilon^2/36$. The second observation is, $\varepsilon^{2.83} < 2^{2.83} < \frac{2 \cdot 9 \cdot 6^6}{52 \cdot 27 \cdot 3^{3.17}}$ and thus $\frac{52}{9} \cdot (\varepsilon/6)^{2k} \leq \frac{52}{9} \cdot (\varepsilon/6)^6 = \frac{52}{9 \cdot 6^6} \cdot \varepsilon^{2.83} \cdot \varepsilon^{3.17} < \frac{2}{27} \cdot (\varepsilon/3)^{3.17}$. With these observations, by (7), (8) and (9), it suffices to set,

$$t = \frac{52}{9} \cdot (\varepsilon/6)^{2k}$$

unconditionally, so that evolution can achieve its goal with a uniform setup for every $p \in \mathcal{I}$. □

Algorithm 2: Mutator so that the (1+1) EA converges with a uniform setup for a class of binomial distributions.

Input: n, $\alpha \in (0, 3/13)$, $\delta \in (0,1)$, $\varepsilon \in (0,2)$, $h \in \mathcal{H}$
Output: a new hypothesis

$q \leftarrow \lceil \log_2(3/\varepsilon) \rceil$; $k \leftarrow \lceil \log_2(1/\alpha) \rceil$;
$h' \leftarrow$ `Mutate(`h`)`;
if $|h'| \leq q$ **then** $N \leftarrow \{h'\}$; **else return** h;
$t \leftarrow \frac{52}{9} \cdot (\varepsilon/6)^{2k}$;
$\delta_s \leftarrow \delta^2/(142en^2q)$;
$\epsilon_s \leftarrow t/2$;
$\nu_h \leftarrow$ `Perf(`p, h, ϵ_s, δ_s`)`; $\nu_{h'} \leftarrow$ `Perf(`p, h', ϵ_s, δ_s`)`
if $\nu_{h'} > \nu_h + t$ **then return** h';
else if $\nu_{h'} \geq \nu_h - t$ **then return** `USelect(`$\{h\} \cup \{h'\}$`)`;
else return h;

8 Conclusions

We examined a (1+1) EA under a set of binomial distributions. We provided distribution-specific results as well as a distribution-independent result for a class of distributions. A natural open question is whether the (1+1) EA can converge for a broader set of distributions as, for example, the swapping algorithm in (Diochnos 2016) does.

Acknowledgements

The author would like to thank Alyson Irizarry for some fruitful discussions in the framework of EAs.

A Fact

Proposition 1 (Hoeffding Bound; (Hoeffding 1963)). *Let $X_1, \ldots, X_R$ be R independent random variables, each taking values in the range $\mathfrak{I} = [\alpha, \beta]$. Let μ denote the mean of their expectations. Then* $\mathbf{Pr}\left(\left|\frac{1}{R}\sum_{i=1}^{R} X_i - \mu\right| \geq \epsilon\right) \leq 2e^{-2R\epsilon^2/(\beta-\alpha)^2}$.

B Omitted Discussion Related to the Uniform Distribution $\mathcal{U}_n$

Lemma 3 (Longer is Better under $\mathcal{H}_{<1/2}$). *Let $h, h' \in \mathcal{H}_{<1/2}$ such that $|h'| < |h| \leq q$. Then, under the uniform distribution $\mathcal{U}_n$, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{1-2q}$.*

Proof. We will prove the lemma by distinguishing cases on the size of the target. The technique is identical to the case where $p = 1/3$. Let $|h| = \lambda + |h'|$, for $\lambda \geq 1$.

Case $|c| \leq q + 1$. By (6), we have $\text{Perf}_{\mathcal{U}_n}(h, c) = 1 - 2^{1-|c|} - 2^{1-|h|} + 2^{2-|c|-r}$. Since $|h| = \lambda + |h'|$ and moreover the number r of redundant variables in h can be $r \in \{0, \ldots, q\}$, it follows that

$$\text{Perf}_{\mathcal{U}_n}(h, c) \geq 1 - 2^{1-|c|} - 2^{1-\lambda-|h'|} + 2^{2-|c|-q}. \quad (10)$$

On the other hand, by (6), letting u' be the number of good undiscovered variables in h', we have $\text{Perf}_{\mathcal{U}_n}(h', c) = 1 - 2^{1-|c|} - 2^{1-|h'|} + 2^{2-|h'|-u'}$. Since $h' \in \mathcal{H}_{<1/2} \Rightarrow u' \geq 2$ it follows that

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h', c) &\leq 1 - 2^{1-|c|} - 2^{1-|h'|} + 2^{-|h'|} \\ &= 1 - 2^{1-|c|} - 2^{-|h'|}. \end{aligned} \quad (11)$$

Thus, by (10) and (11) it follows that

$$\begin{aligned} \Delta &= \text{Perf}_{\mathcal{U}_n}(h, c) - \text{Perf}_{\mathcal{U}_n}(h', c) \\ &\geq 2^{-|h'|} \cdot (1 - 2^{1-\lambda}) + 2^{2-|c|-q} \\ &\geq 2^{1-2q}, \end{aligned} \quad (12)$$

where the last inequality is obtained since $\lambda \geq 1$ and $|c| \leq q + 1$.

Case $|c| \geq q + 2$. For h, (10) continues to hold. On the other hand, letting r' be the number of (bad) redundant variables in h', we have $\text{Perf}_{\mathcal{U}_n}(h', c) = 1 - 2^{1-|c|} - 2^{1-|h'|} + 2^{2-|c|-r'}$, and thus

$$\text{Perf}_{\mathcal{U}_n}(h', c) \leq 1 - 2^{1-|c|} - 2^{1-|h'|} + 2^{2-|c|}. \quad (13)$$

Hence, by (10) and (13) we have

$$\begin{aligned} \Delta &= \text{Perf}_{\mathcal{U}_n}(h, c) - \text{Perf}_{\mathcal{U}_n}(h', c) \\ &\geq 2^{1-|h'|} \cdot (1 - 2^{-\lambda}) - 2^{2-|c|} \cdot (1 - 2^{-q}) \\ &\geq 2^{1-(q-1)} \cdot \frac{1}{2} - 2^{2-|c|} \\ &\geq 2^{1-q} - 2^{-q} \\ &= 2^{-q} \end{aligned}$$

The lemma follows by observing that since $q \geq 1$, we have that $1 \geq 2^{1-q} \Rightarrow 2^{-q} \geq 2^{1-2q}$. □

Lemma 4 ($\mathcal{H}_{1/2} \not\rightsquigarrow \mathcal{H}_{<1/2}$). *Under $\mathcal{U}_n$, let $h \in \mathcal{H}_{1/2}$ and $h' \in \mathcal{H}_{<1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{-q}$.*

Proof. First, $h \in \mathcal{H}_{1/2} \Rightarrow u = 1$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h, c) &= 1 - 2^{1-|h|} - 2^{1-|c|} + 2^{2-|h|-1} \\ &= 1 - 2^{1-|c|}. \end{aligned}$$

On the other hand, $h' \in \mathcal{H}_{<1/2} \Rightarrow u' \geq 2$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h', c) &= 1 - 2^{1-|h'|} - 2^{1-|c|} + 2^{2-|h'|-u'} \\ &\leq 1 - 2^{1-|h'|} - 2^{1-|c|} + 2^{2-|h'|-2} \\ &= 1 - 2^{-|h'|} - 2^{1-|c|} \\ &\leq 1 - 2^{-q} - 2^{1-|c|} \end{aligned}$$

It follows that $\text{Perf}_{\mathcal{U}_n}(h, c) - \text{Perf}_{\mathcal{U}_n}(h', c) \geq 2^{-q}$. □

Lemma 5 ($\mathcal{H}_{>1/2} \not\rightsquigarrow \mathcal{H}_{1/2}$). *Under $\mathcal{U}_n$, let $h \in \mathcal{H}_{>1/2}$ and $h' \in \mathcal{H}_{1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 2^{1-q}$.*

Proof. First, $h \in \mathcal{H}_{>1/2} \Rightarrow u = 0$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h, c) &= 1 - 2^{1-|h|} - 2^{1-|c|} + 2^{2-|h|-u} \\ &= 1 - 2^{1-|h|} - 2^{1-|c|} + 2^{2-|h|} \\ &= 1 + 2^{1-|h|} - 2^{1-|c|} \\ &\geq 1 + 2^{1-q} - 2^{1-|c|} \end{aligned}$$

On the other hand, $h' \in \mathcal{H}_{1/2} \Rightarrow u' = 1$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h', c) &= 1 - 2^{1-|h'|} - 2^{1-|c|} + 2^{2-|h'|-1} \\ &= 1 - 2^{1-|c|}. \end{aligned}$$

It follows that $\text{Perf}_{\mathcal{U}_n}(h, c) - \text{Perf}_{\mathcal{U}_n}(h', c) \geq 2^{1-q}$. □

Lemma 6 ($\mathcal{H}_{>1/2} \not\rightsquigarrow \mathcal{H}_{<1/2}$ under $\mathcal{U}_n$). *Under the uniform distribution $\mathcal{U}_n$, let $h \in \mathcal{H}_{>1/2}$ and $h' \in \mathcal{H}_{<1/2}$. Then, Perf$_{\mathcal{U}_n}(h, c) \geq$ Perf$_{\mathcal{U}_n}(h', c) + 3 \cdot 2^{-q}$.*

Proof. First, $h \in \mathcal{H}_{>1/2} \Rightarrow u = 0$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h, c) &= 1 - 2^{1-|h|} - 2^{1-|c|} + 2^{2-|h|-u} \\ &= 1 - 2^{1-|h|} - 2^{1-|c|} + 2^{2-|h|} \\ &= 1 + 2^{1-|h|} - 2^{1-|c|} \\ &\geq 1 + 2^{1-q} - 2^{1-|c|} \end{aligned}$$

On the other hand, $h' \in \mathcal{H}_{<1/2} \Rightarrow u' \geq 2$. Then, by (6),

$$\begin{aligned} \text{Perf}_{\mathcal{U}_n}(h', c) &= 1 - 2^{1-|h'|} - 2^{1-|c|} + 2^{2-|h'|-u'} \\ &\leq 1 - 2^{1-|h'|} - 2^{1-|c|} + 2^{2-|h'|-2} \\ &= 1 - 2^{-|h'|} - 2^{1-|c|} \\ &\leq 1 - 2^{-q} - 2^{1-|c|} \end{aligned}$$

Thus, $\text{Perf}_{\mathcal{U}_n}(h, c) - \text{Perf}_{\mathcal{U}_n}(h', c) \geq 3 \cdot 2^{-q}$. □

Theorem 4 (Evolution in $\mathcal{C}_n^{\leq q}$ under $\mathcal{U}_n$). *Let $q = \lceil \log_2(3/\varepsilon) \rceil$. Under the uniform distribution $\mathcal{U}_n$, the (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(n^2 q/(\delta\varepsilon^4)\right)$, will evolve a hypothesis h such that* $\mathbf{Pr}\left(\text{Perf}_{\mathcal{U}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.

Proof. Note that long targets are taken care of by Lemma 7 as soon as a hypothesis of size q has been formed. The reason is that any approximation of a long target belongs to $\mathcal{H}_{<1/2}$ and due to Lemma 3 all shorter hypotheses have performance that will be noticeably smaller by the selection of tolerance. Thus, below we will only discuss about short and medium targets.

Phase 1. In the first phase of the evolution (Lemma 7), as long as $h \in \mathcal{H}_{<1/2}$, increasing the size of h results in increase in performance by an amount of at least 2^{1-2q} due to Lemma 3. Lemma 7 serves its purpose when either $h \in \mathcal{H}_{1/2}$ (leading to phase 3), or $h \in \mathcal{H}_{>1/2}$ (leading to phase 4), or it is still the case that $h \in \mathcal{H}_{<1/2}$ and moreover $|h| = q$ (leading to phase 2).

Phase 2. In the second phase of the evolution (Lemma 8), as long as $h \in \mathcal{H}_{<1/2}$ and $|h| = q$, due to Lemma 3, any shorter hypothesis will have noticeably smaller performance by the selection of tolerance. Thus, the only beneficial mutations are those that increase the number of good variables while retaining the size to be q for the hypothesis. The smallest increase in performance on such beneficial swaps is obtained when only one bad variable is replaced by a good one in one step and the increase is $4p^{|h|+u-1}(1-p) = 2^{2-|h|-u} \geq 2^{1-2q}$. (Note also that if many good variables are brought into the hypothesis in one generation, such that the resulting hypothesis belongs to $\mathcal{H}_{1/2} \cup \mathcal{H}_{>1/2}$, then by Lemma 4 the increase in performance is at least 2^{-q} and by Lemma 6 the increase in performance is at least $3 \cdot 2^{-q}$. Both such increases are at least as large as 2^{1-2q} for any $q \geq 1$.) Hence we reach phase 3 ($h \in \mathcal{H}_{1/2}$) or phase 4 ($h \in \mathcal{H}_{>1/2}$) below.

Phase 3. In the third phase of the evolution (Lemma 9), a hypothesis h_0 has been formed that is missing precisely one variable from the target c. Due to Lemma 4, any other short hypothesis $h \in \mathcal{H}_{<1/2}$ has noticeably smaller performance by an amount of at least 2^{-q}.

In the case where the target is medium (that is, $|c| = q+1$), then h_0 is already a best q-approximation of c since $h_0 \in \mathcal{H}_{1/2} \Rightarrow u = 1$. Due to our remark above, this formation is stable, as any hypothesis h with $|h| < q$ will have $u \geq 2$, and moreover, among the hypotheses of size q again there can not be more than 2 good variables missing from h as this would imply $h \in \mathcal{H}_{<1/2}$ again.

In the case where the target is short, we have that $m = |c| - 1 \leq q - 1$. Therefore, $|h_0| \in \{|c|-1, |c|, \ldots, q\}$. As explained in the proof of Lemma 9 we are interested in introducing the last missing good variable to h, which can either happen by appending it to h when $|h| < q$, or by performing a beneficial swap when there are redundant bad variables present in h. Either of these two mutations results in a new hypothesis $h' \in \mathcal{H}_{>1/2}$ and due to Lemma 5 the increase in performance is at least 2^{1-q}. Similarly, until such a good event occurs, by the selection of tolerance, any hypothesis $h'' \in \mathcal{H}_{<1/2}$ has noticeably smaller performance compared to h by an amount of at least 2^{-q} due to Lemma 4. Thus, when the target is short, with the application of Lemma 9, we reach phase 4.

Phase 4. In the fourth phase of the evolution (Lemma 10), a specialization of a short target has been formed. Removing or replacing one or more good variables from the hypothesis within one mutation, results in transitioning from a hypothesis $h \in \mathcal{H}_{>1/2}$ to a hypothesis $h' \in \mathcal{H}_{1/2} \cup \mathcal{H}_{<1/2}$ and due to Lemmata 5 and 6, the decrease in performance is at least 2^{1-q}. By the selection tolerance such mutations will be characterized as deleterious. Thus, the only beneficial mutations that can occur are those that delete one or more bad variables from h in one step (without affecting the good variables). Deleting one such bad variable results in increase in performance by an amount of $2p^{|h|-1}(1-p) = 2^{1-|h|} \geq 2^{1-q}$.

Once we reach the target, any mutation that removes one or more good variables, due to Lemmata 5 and 6, will have as a result a noticeable decrease in performance by an amount of at least 2^{1-q}. Similarly introducing one or more bad variables (which thus maintains that $h \in \mathcal{H}_{>1/2}$) also results in a decrease in performance by an amount $|\Delta| = 2p^{|h|}(1-p) = 2^{-|h|} \geq 2^{-q}$.

Thus, from all four phases, when a beneficial or deleterious mutation occurs, then the performance of the hypothesis is affected by an additive amount at least 2^{1-2q}. Therefore, we set the tolerance t to be

$$t = 2^{-2q}\,.$$

Due to Lemmata 7, 8, 9 and 10, evolution lasts not more than $\lceil 16enq/\delta \rceil + \lceil 16en^2q/\delta \rceil + \lceil 16en^2/\delta \rceil + \lceil 16enq/\delta \rceil \leq 4 + 32enq/\delta + 17en^2q/\delta \leq 53en^2q/\delta$ generations (regardless of the target) with failure probability, by a union bound, not more than $\delta/4$. The neighborhood in each generation has size not larger than 2. Hence, the total number of hypotheses that need to be estimated is not more than $106en^2q/\delta$. By the analysis above we want to estimate the performance of each hypothesis within $\epsilon_s = t/2$ of its true value.

Requiring $R \geq \left\lceil \frac{8}{t^2} \cdot \ln\left(\frac{284en^2q}{\delta^2}\right) \right\rceil$ samples for estimating the empirical performance of each hypothesis, it follows by Hoeffding's bound (Proposition 1), using $\alpha = -1$ and $\beta = 1$, that the empirical performance of each hypothesis is estimated within $\epsilon_s = t/2$ of its exact value with probability at least $1 - \delta^2/(142en^2q)$. As the number of different hypotheses is not more than $106en^2q/\delta$, by the union bound, the performance of *every* hypothesis in this phase is computed within $\epsilon_s = t/2$ of its exact value except with probability at most $\sum_{i=1}^{106en^2q/\delta} \delta^2/(142en^2q) \leq \sum_{i=1}^{106en^2q/\delta} 3\delta^2/(4 \cdot 106en^2q) = 3\delta/4$.

Thus, with a union bound argument again, the performance of each hypothesis is computed within $\epsilon_s = t/2$ of its true value and evolution achieves its goal within $\mathcal{O}\left(n^2q/\delta\right)$ generations with total probability at least $1 - \delta$.

Since $q = \lceil \log_2(3/\varepsilon) \rceil$ we have $2^{-q} \geq \frac{\varepsilon}{6}$ and hence $t = 2^{-2q} \geq \varepsilon^2/36$. The total sample size follows. □

C Omitted Discussion Related to Binomial Distributions with $p \in (0, 1/3]$

Lemma 11. *Let $h, h' \in \mathcal{H}_{<1/2}$ such that $|h'| < |h| \leq q$. Then, under a binomial distribution $\mathcal{B}_n$ with $p \in (0, 1/3)$, $\mathit{Perf}_{\mathcal{B}_n}(h, c) \geq \mathit{Perf}_{\mathcal{B}_n}(h', c) + 2p^{q-1} \cdot (1 - 3p)$.*

Proof. Let the target be

$$c = \bigwedge_{i=1}^{m_1} x_i \wedge \bigwedge_{i=m_1+1}^{m} x_i \wedge \bigwedge_{k=1}^{u_1} y_k \wedge \bigwedge_{k=u_1+1}^{u} y_k \,.$$

Further let,

$$\begin{cases} h = \bigwedge_{i=1}^{m_1} x_i \wedge \bigwedge_{i=m_1+1}^{m} x_i \wedge \bigwedge_{\ell=1}^{r_1} w_\ell \wedge \bigwedge_{\ell=r_1+1}^{r} w_\ell \\ h' = \bigwedge_{i=1}^{m_1} x_i \wedge \bigwedge_{k=1}^{u_1} y_k \wedge \bigwedge_{\ell=1}^{r_1} w_\ell \wedge \bigwedge_{j=1}^{r_3} z_j \end{cases}$$

be the two short hypotheses, such that $|h| = |h'| + \lambda$ for $\lambda \geq 1$ and moreover, $U = \prod_{k=1}^{u} p_{y_k} = p^u < 1/2$ and $U' = \left(\prod_{i=m_1+1}^{m} p_{x_i}\right) \cdot \left(\prod_{k=u_1+1}^{u} p_{y_k}\right) = p^{m_2} \cdot p^{u_2} < 1/2$, where $m_2 = |\{x_{m_1+1}, \ldots, x_m\}|$, $u_2 = |\{y_{u_1+1}, \ldots, y_u\}|$ and $r_2 = |\{w_{r_1+1}, \ldots, w_r\}|$.

By construction we have $|h| = m_1 + m_2 + r_1 + r_2 = \lambda + m_1 + u_1 + r_1 + r_3 = \lambda + |h'|$. In other words, it holds

$$m_2 + r_2 = u_1 + r_3 + \lambda \,. \tag{14}$$

For the difference Δ in performance we have

$$\begin{aligned}
\Delta &= \mathrm{Perf}_{\mathcal{B}_n}(h, c) - \mathrm{Perf}_{\mathcal{B}_n}(h', c) \\
&= 2p^{m_1+u_1+r_1+r_3} + 2p^{m_1+m_2+u_1+u_2} \\
&\quad -4p^{m_1+m_2+u_1+u_2+r_1+r_3} \\
&\quad -2p^{m_1+m_2+r_1+r_2} - 2p^{m_1+m_2+u_1+u_2} \\
&\quad +4p^{m_1+m_2+u_1+u_2+r_1+r_2} \\
&= 2p^{m_1+r_1} \cdot \left(p^{u_1+r_3} - p^{m_2+r_2}\right) \\
&\quad +2p^{m_1+r_1} \cdot \left(2p^{m_2+u_1+u_2+r_2} - 2p^{m_2+u_1+u_2+r_3}\right) \\
&= 2p^{m_1+r_1} \cdot \left(p^{u_1+r_3} - p^{u_1+r_3+1}\right) \\
&\quad +2p^{m_1+r_1} \cdot \left(2p^{m_2+u_1+u_2+r_2} - 2p^{m_2+u_1+u_2+r_3}\right) \\
&= 2p^{m_1+r_1+u_1+r_3} - 2p^{m_1+r_1+u_1+r_3}p \\
&\quad +4p^{m_1+m_2+r_1+r_2+u_1+u_2} \\
&\quad -4p^{m_1+m_2+r_1+r_3+u_1+u_2} \\
&= 2p^{m_1+r_1+u_1+r_3} \cdot (1-p) \\
&\quad +2p^{m_1+r_1+u_1+r_3} \cdot \left(2p^{m_2+r_2+u_2-r_3} - 2p^{m_2+u_2}\right) \\
&\geq 2p^{|h'|} \cdot \left(1 - p + 2p^{\lambda+u_1+u_2} - 2p\right) \qquad (15) \\
&> 2p^{|h'|} \cdot (1 - 3p) \\
&\geq 2p^{q-1} \cdot (1 - 3p) \,,
\end{aligned}$$

where (15) follows by (14) and the fact that $p^{m_2+u_2} = U' < 1/2 \Rightarrow m_2 + u_2 \geq 1$. The claim follows. □

Lemma 12. *Let $h, h' \in \mathcal{H}_{<1/2}$ such that $|h'| < |h| \leq q$. Then, under a binomial distribution $\mathcal{B}_n$ with $p = 1/3$, $\mathit{Perf}_{\mathcal{B}_n}(h, c) \geq \mathit{Perf}_{\mathcal{B}_n}(h', c) + \frac{4}{3} \cdot 3^{-2q}$.*

Proof. We will prove the lemma by distinguishing cases on the size of the target. Let $|h| = \lambda + |h'|$, for $\lambda \geq 1$.

Case $|c| \leq q + 1$. By (6), we have $\mathrm{Perf}_{\mathcal{B}_n}(h, c) = 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-\lambda-|h'|} + 4 \cdot 3^{-|c|-r}$. Since the number r of redundant variables in h can be $r \in \{0, \ldots, q\}$, we have,

$$\mathrm{Perf}_{\mathcal{B}_n}(h, c) \geq 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-\lambda-|h'|} + 4 \cdot 3^{-|c|-q} \,. \tag{16}$$

On the other hand, by (6), letting u' be the number of good undiscovered variables in h', we have $\mathrm{Perf}_{\mathcal{B}_n}(h', c) = 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-|h'|} + 4 \cdot 3^{-|h'|-u'}$. Since $h' \in \mathcal{H}_{<1/2} \Rightarrow u' \geq 1$ it follows that

$$\begin{aligned}
\mathrm{Perf}_{\mathcal{B}_n}(h', c) &\leq 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-|h'|} + 4 \cdot 3^{-|h'|-1} \\
&= 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-|h'|-1} \,. \qquad (17)
\end{aligned}$$

Thus, by (16) and (17) it follows that

$$\begin{aligned}
\Delta &= \mathrm{Perf}_{\mathcal{B}_n}(h, c) - \mathrm{Perf}_{\mathcal{B}_n}(h', c) \\
&\geq 2 \cdot 3^{-|h'|-1} - 2 \cdot 3^{-|h'|-\lambda} + 4 \cdot 3^{-|c|-q} \\
&\geq 4 \cdot 3^{-2q-1} \,,
\end{aligned}$$

where the last inequality is obtained since $|c| \leq q + 1$ and we also used the fact that $\lambda \geq 1$.

Case $|c| \geq q + 2$. For h, (16) continues to hold. On the other hand, letting r' be the number of (bad) redundant variables in h', we have $\mathrm{Perf}_{\mathcal{B}_n}(h', c) = 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-|h'|} + 4 \cdot 3^{-|c|-r'}$, and thus

$$\mathrm{Perf}_{\mathcal{B}_n}(h', c) \leq 1 - 2 \cdot 3^{-|c|} - 2 \cdot 3^{-|h'|} + 4 \cdot 3^{-|c|} \,. \tag{18}$$

Hence, by (16) and (18) we have

$$\begin{aligned}
\Delta &= \mathrm{Perf}_{\mathcal{B}_n}(h, c) - \mathrm{Perf}_{\mathcal{B}_n}(h', c) \\
&\geq 2 \cdot 3^{-|h'|} - 2 \cdot 3^{-|h'|-\lambda} + 4 \cdot 3^{-|c|-q} - 4 \cdot 3^{-|c|} \\
&> 2 \cdot 3^{-|h'|} \cdot \left(1 - 3^{-\lambda}\right) - 4 \cdot 3^{-|c|} \\
&\geq 2 \cdot 3^{1-q} \cdot \frac{2}{3} - 4 \cdot 3^{-q-2} \\
&= \frac{32}{9} \cdot 3^{-q} \,.
\end{aligned}$$

Since $q \geq 1$, the lemma follows by observing that $\frac{32}{9} \cdot 3^{-q} > \frac{4}{3} \cdot 3^{-q} \geq \frac{4}{3} \cdot 3^{-2q}$. □

Lemma 13 ($\mathcal{H}_{>1/2} \not\rightsquigarrow \mathcal{H}_{<1/2}$ under $\mathcal{B}_n$ with $p \in (0, 1/3]$)**.** *Under a binomial distribution $\mathcal{B}_n$ with parameter $p \in (0, 1/3]$, let $h \in \mathcal{H}_{>1/2}$ and $h' \in \mathcal{H}_{<1/2}$ such that $|h| \leq q$ and $|h'| \leq q$. Then, $\mathit{Perf}_{\mathcal{B}_n}(h, c) \geq \mathit{Perf}_{\mathcal{B}_n}(h', c) + \frac{8}{3} \cdot p^q$.*

Proof. First, $h \in \mathcal{H}_{>1/2} \Rightarrow u = 0$. Then, by (6),

$$\begin{aligned}
\mathrm{Perf}_{\mathcal{B}_n}(h, c) &= 1 - 2p^{|h|} - 2p^{|c|} + 4p^{|h|+u} \\
&= 1 - 2p^{|h|} - 2p^{|c|} + 4p^{|h|+0} \\
&= 1 + 2p^{|h|} - 2p^{|c|} \,.
\end{aligned}$$

Before we proceed, note that for $p \in (0, 1/3]$, we have $1 - 2p^u \geq 1 - 2p \geq 1 - 2/3 = 1/3$ for any integer $u \geq 1$.

Then, since $h' \in \mathcal{H}_{<1/2}$ we have $u' \geq 1$. As a result, by (6),

$$\begin{aligned}\text{Perf}_{\mathcal{B}_n}(h', c) &= 1 - 2p^{|h'|} - 2p^{|c|} + 4p^{|h'|+u'} \\ &= 1 - 2p^{|c|} - 2p^{|h'|}\left(1 - 2p^{u'}\right) \\ &\leq 1 - 2p^{|c|} - \frac{2}{3} \cdot p^{|h'|}.\end{aligned}$$

It follows that $\text{Perf}_{\mathcal{B}_n}(h, c) - \text{Perf}_{\mathcal{B}_n}(h', c) \geq 2p^{|h|} + \frac{2}{3} \cdot p^{|h'|} \geq 2p^q + \frac{2}{3} \cdot p^q = \frac{8}{3} \cdot p^q$. □

Convergence Theorems for the (1+1) EA

Theorem 5 (Evolution in $\mathcal{C}_n^{\leq q}$ when $p \in (0, 1/3)$)**.** *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p \in \mathbb{R}_{alg}$ such that $p \in (0, 1/3)$. The (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2 q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(\frac{n^2 q}{\delta \cdot \varepsilon^2 \cdot (\min\{4p\varepsilon/9, 1-3p\})^2}\right)$, will evolve a hypothesis h such that $\mathbf{Pr}\left(\mathit{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.*

Proof. Note that long targets are taken care of by Lemma 7 as soon as a hypothesis of size q has been formed. Thus, below we will only discuss about short targets.

Phase 1. In the first phase of the evolution (Lemma 7), as long as $h \in \mathcal{H}_{<1/2}$, increasing the size of h results in increase in performance by an amount of at least $2p^{q-1}(1-3p)$ due to Lemma 11. Lemma 7 serves its purpose when either $h \in \mathcal{H}_{>1/2}$ (leading to phase 3 below), or it is still the case that $h \in \mathcal{H}_{<1/2}$ and moreover $|h| = q$ (leading to phase 2). Of course it can happen that all the good variables are brought into the hypothesis in one generation; then by Lemma 13 the increase in performance is at least $\frac{8}{3}p^q$.

Phase 2. In the second phase of the evolution (Lemma 8), as long as $h \in \mathcal{H}_{<1/2}$ and $|h| = q$, due to Lemma 11, any shorter hypothesis will have noticeably smaller performance by the selection of tolerance. Thus, the only beneficial mutations are those that increase the number of good variables while retaining the size of the hypothesis to be q. The smallest increase in performance on such beneficial swaps is obtained when only one bad variable is replaced by a good one in one step and the increase is $4p^{|h|+u-1}(1-p) \geq 4p^{2q-1}(1-p) \geq 8p^{2q-1}/3$. Hence, we reach phase 3 once sufficiently many beneficial swaps have occurred.

Phase 3. In the third phase of the evolution (Lemma 10), a specialization of the target has been formed. Removing or replacing one or more good variables from the hypothesis results in transitioning from a hypothesis $h \in \mathcal{H}_{>1/2}$ to a hypothesis $h' \in \mathcal{H}_{<1/2}$; due to Lemma 13 the decrease in performance will be at least $\frac{8}{3}p^q$ and by the selection of the tolerance it will be characterized as deleterious. Thus, the only beneficial mutations that can occur are those that delete one or more bad variables from h in one step (without affecting the good ones). Deleting such a bad variable increases the performance by an amount of $2p^{|h|-1}(1-p) \geq 2p^{q-1}(1-p)$. In turn, for any $p \in (0, 1/3]$ we have that $2p^{q-1}(1-p) \geq 4p^{2q-1}(1-p)$. Once we reach the target, any mutation that removes one or more good variables, due to Lemma 13 will have as a result a noticeable decrease in performance. Similarly introducing one or more bad variables (which thus maintains that $h \in \mathcal{H}_{>1/2}$) also results in a decrease in performance by an amount $|\Delta| = 2p^{|h|}(1-p) \geq 2p^{q-1}(1-p)$.

Thus, from all three phases, and using the fact that $8p^{2q-1}/3$ is a lower bound for the three quantities $2p^{q-1}(1-p)$, $4p^{2q-1}(1-p)$, and $\frac{8}{3}p^q$, when a mutation occurs that affects the true performance for any $0 < p \leq 1/3$, then the performance of a hypothesis changes by at least an additive factor of $|\Delta| = 2p^{q-1} \cdot \min\{4p^q/3,\ 1-3p\}$. We thus set the tolerance to be

$$t = p^{q-1} \cdot \min\{4p^q/3,\ 1-3p\}.$$

Due to Lemmata 7, 8 and 10, evolution lasts not more than $\lceil 16enq/\delta \rceil + \lceil 16en^2q/\delta \rceil + \lceil 16enq/\delta \rceil \leq 3 + 32enq/\delta + 16en^2q/\delta \leq 35enq/\delta + 16en^2q/\delta \leq 51en^2q/\delta$ generations (regardless of the target) with failure probability, by a union bound, not more than $3\delta/16$. The neighborhood in each generation has size not larger than 2. Hence, the total number of hypotheses that need to be estimated is not more than $102en^2q/\delta$. Further, we want to estimate the performance of each hypothesis within $\epsilon_s = t/2$ of its true value.

Requiring $R \geq \left\lceil \frac{8}{t^2} \cdot \ln\left(\frac{252en^2q}{\delta^2}\right) \right\rceil$ samples for estimating the empirical performance of each hypothesis, it follows by Hoeffding's bound (Proposition 1), using $\alpha = -1$ and $\beta = 1$, that the empirical performance of each hypothesis is estimated within $\epsilon_s = t/2$ of its exact value except with probability at most $\delta^2/(126en^2q)$. As the number of different hypotheses is not more than $102en^2q/\delta$, by the union bound, the performance of *every* hypothesis in this phase is computed within $\epsilon_s = t/2$ of its exact value except with probability at most $\sum_{i=1}^{102en^2q/\delta} \delta^2/(126en^2q) \leq \sum_{i=1}^{102en^2q/\delta} 13\delta^2/(16 \cdot 102en^2q) = 13\delta/16$.

Thus, with a union bound argument again, the performance of each hypothesis is computed within $\epsilon_s = t/2$ of its true value and evolution achieves its goal within $\mathcal{O}\left(n^2q/\delta\right)$ generations except with probability at most δ.

Since $q = \lceil \log_{1/p}(3/\varepsilon) \rceil$ we have $p^q \geq p^{1+\log_{1/p}(3/\varepsilon)} = p\varepsilon/3$ and hence $t = p^{q-1} \cdot \min\{4p^q/3, 1-3p\} \geq \frac{\varepsilon}{3} \cdot \min\{4p\varepsilon/9, 1-3p\}$. The sample size follows. □

Theorem 6 (Evolution in $\mathcal{C}_n^{\leq q}$ when $p = 1/3$)**.** *Let $\mathcal{B}_n$ be a binomial distribution with parameter $p = 1/3$. The (1+1) EA, using the hypothesis class $\mathcal{H} = \mathcal{C}_n^{\leq q}$, in $\mathcal{O}\left(n^2q/\delta\right)$ generations, with total sample size $\widetilde{\mathcal{O}}\left(\frac{n^2q}{\delta \cdot \varepsilon^4}\right)$, will evolve a hypothesis h such that $\mathbf{Pr}\left(\mathit{Perf}_{\mathcal{B}_n}(h, c) > 1 - \varepsilon\right) \geq 1 - \delta$.*

Proof. The only difference in the proof of this theorem compared to Theorem 5 is that for phases 1 and 2, instead of using Lemma 11 we use Lemma 12. Thus, the performance of a hypothesis due to beneficial or deleterious mutations, is affected by at least an amount of $\min\{8 \cdot 3^{1-2q}/3, 4 \cdot$

$3^{-1-2q}\} = \min\{8 \cdot 3^{-2q}, \frac{4}{3} \cdot 3^{-2q}\} = \frac{4}{3} \cdot 3^{-2q}$. Therefore, we set the tolerance t to be

$$t = 2 \cdot 3^{-1-2q} .$$

Since $q = \lceil \log_3(3/\varepsilon) \rceil$, we have that $3^{-q} > 3^{-2} \cdot \varepsilon$. As a consequence from the above, for the tolerance we have $t = 2 \cdot 3^{-1-2q} \geq 2 \cdot 3^{-5} \cdot \varepsilon^2$. The sample size follows. □

References

Ajtai, M.; Feldman, V.; Hassidim, A.; and Nelson, J. 2016. Sorting and selection with imprecise comparisons. *ACM Transactions on Algorithms* 12(2):19.

Angelino, E., and Kanade, V. 2014. Attribute-efficient evolvability of linear functions. In *ITCS*, 287–300.

Angluin, D., and Laird, P. D. 1987. Learning From Noisy Examples. *Machine Learning* 2(4):343–370.

Aslam, J. A., and Decatur, S. E. 1998. Specification and Simulation of Statistical Query Algorithms for Efficiency and Noise Tolerance. *Journal of Computer and System Sciences* 56(2):191–208.

Astete-Morales, S.; Cauwet, M.-L.; and Teytaud, O. 2015. Evolution Strategies with Additive Noise: A Convergence Rate Lower Bound. In *FOGA*, 76–84.

Balcan, M.-F.; Berlind, C.; Ehrlich, S.; and Liang, Y. 2013. Efficient Semi-Supervised and Active Learning of Disjunctions. In *ICML*, 633–641.

Ben-David, S.; Itai, A.; and Kushilevitz, E. 1995. Learning by Distances. *Information and Computation* 117(2):240–250.

Benedek, G. M., and Itai, A. 1991. Learnability with Respect to Fixed Distributions. *Theoretical Computer Science* 86(2):377–390.

Blum, A.; Furst, M. L.; Jackson, J. C.; Kearns, M. J.; Mansour, Y.; and Rudich, S. 1994. Weakly learning DNF and characterizing statistical query learning using Fourier analysis. In *STOC*, 253–262.

Bshouty, N. H., and Feldman, V. 2002. On Using Extended Statistical Queries to Avoid Membership Queries. *Journal of Machine Learning Research* 2:359–395.

Bshouty, N. H., and Tamon, C. 1996. On the Fourier Spectrum of Monotone Functions. *Journal of the ACM* 43(4):747–770.

Bshouty, N. H.; Eiron, N.; and Kushilevitz, E. 2002. PAC learning with nasty noise. *Theoretical Computer Science* 288(2):255–275.

Corus, D.; Dang, D.; Eremeev, A. V.; and Lehre, P. K. 2014. Level-Based Analysis of Genetic Algorithms and Other Search Processes. In *PPSN*, 912–921.

Dang, D.-C., and Lehre, P. K. 2015. Efficient Optimisation of Noisy Fitness Functions with Population-based Evolutionary Algorithms. In *FOGA*, 62–68.

Decatur, S. E. 1993. Statistical Queries and Faulty PAC Oracles. In *COLT*, 262–268.

Diochnos, D. I., and Turán, Gy. 2009. On Evolvability: The Swapping Algorithm, Product Distributions, and Covariance. In *SAGA*, 74–88.

Diochnos, D. I.; Emiris, I. Z.; and Tsigaridas, E. P. 2009. On the asymptotic and practical complexity of solving bivariate systems over the reals. *Journal of Symbolic Computation* 44(7):818–835.

Diochnos, D. I. 2016. On the Evolution of Monotone Conjunctions: Drilling for Best Approximations. In *ALT*, 98–112.

Droste, S.; Jansen, T.; and Wegener, I. 2002. On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science* 276(1-2):51–81.

Droste, S. 2004. Analysis of the (1+1) EA for a Noisy OneMax. In *GECCO*, 1088–1099.

Duda, R. O., and Shortliffe, E. H. 1983. Expert Systems Research. *Science* 220:261–268.

Feldman, V.; Grigorescu, E.; Reyzin, L.; Vempala, S. S.; and Xiao, Y. 2017. Statistical Algorithms and a Lower Bound for Detecting Planted Cliques. *Journal of the ACM* 64(2):8:1–8:37.

Feldman, V. 2008. Evolvability from learning algorithms. In *STOC*, 619–628.

Feldman, V. 2009. Robustness of Evolvability. In *COLT*, 277–292.

Feldman, V. 2011. Distribution-Independent Evolvability of Linear Threshold Functions. In *COLT*, 253–272.

Feldman, V. 2012. A Complete Characterization of Statistical Query Learning with Applications to Evolvability. *Journal of Computer and System Sciences* 78(5):1444–1459.

Friedrich, T., and Neumann, F. 2017. What's Hot in Evolutionary Computation. In *AAAI*, 5064–5066.

Gießen, C., and Kötzing, T. 2016. Robustness of Populations in Stochastic Environments. *Algorithmica* 75(3):462–489.

Goldman, S. A., and Sloan, R. H. 1995. Can PAC Learning Algorithms Tolerate Random Attribute Noise? *Algorithmica* 14(1):70–84.

Gutjahr, W. J., and Pflug, G. C. 1996. Simulated Annealing for noisy cost functions. *Journal of Global Optimization* 8(1):1–13.

Hancock, T. R., and Mansour, Y. 1991. Learning Monotone $k\mu$ DNF Formulas on Product Distributions. In *COLT*, 179–183.

Hanneke, S.; Kanade, V.; and Yang, L. 2015. Learning with a Drifting Target Concept. In *ALT 2015*, 149–164.

Hoeffding, W. 1963. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association* 58(301):13–30.

Holland, J. H. 1986. Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems. In Michalski, R. S.; Carbonell, J. G.; and Mitchell, T. M., eds., *Machine Learning, An Artificial Intelligence Approach (Volume II)*. Los Alamos, CA: Morgan Kaufmann. chapter 20, 593–623.

Hoos, H. H., and Stützle, T. 2004. *Stochastic Local Search: Foundations & Applications*. Elsevier / Morgan Kaufmann.

Jackson, J. C., and Servedio, R. A. 2006. On Learning Random DNF Formulas Under the Uniform Distribution. *Theory of Computing* 2(8):147–172.

Jackson, J. C.; Klivans, A. R.; and Servedio, R. A. 2002. Learnability beyond AC0. In *STOC*, 776–784.

Kalai, A. T., and Vempala, S. 2006. Simulated annealing for convex optimization. *Mathematics of Operations Research* 31(2):253–266.

Kanade, V.; Valiant, L. G.; and Vaughan, J. W. 2010. Evolution with Drifting Targets. In *COLT*, 155–167.

Kanade, V. 2011. Evolution with Recombination. In *FOCS*, 837–846.

Kearns, M. J., and Li, M. 1993. Learning in the Presence of Malicious Errors. *SIAM Journal on Computing* 22(4):807–837.

Kearns, M. J. 1998. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM* 45(6):983–1006.

Khardon, R. 1994. On Using the Fourier Transform to Learn Disjoint DNF. *Information Processing Letters* 49(5):219–222.

Klivans, A. R.; O'Donnell, R.; and Servedio, R. A. 2004. Learning intersections and thresholds of halfspaces. *Journal of Computer and System Sciences* 68(4):808–840.

Kötzing, T.; Neumann, F.; and Spöhel, R. 2011. PAC Learning and Genetic Programming. In *GECCO*, 2091–2096.

Koza, J. R. 1993. *Genetic programming - on the programming of computers by means of natural selection*. Complex adaptive systems. MIT Press.

Laird, P. D. 1988. *Learning from Good and Bad Data*. Boston: Kluwer Academic Publishers.

Lehre, P. K. 2011. Fitness-Levels for Non-Elitist Populations. In *GECCO*, 2075–2082.

Linial, N.; Mansour, Y.; and Nisan, N. 1993. Constant Depth Circuits, Fourier Transform, and Learnability. *Journal of the ACM* 40(3):607–620.

Livnat, A., and Papadimitriou, C. H. 2016. Sex as an algorithm: the theory of evolution under the lens of computation. *Communications of the ACM* 59(11):84–93.

Livnat, A.; Papadimitriou, C.; Dushoff, J.; and Feldman, M. W. 2008. A mixability theory for the role of sex in evolution. *Proceedings of the National Academy of Sciences* 105(50):19803–19808.

Livnat, A.; Papadimitriou, C.; Pippenger, N.; and Feldman, M. W. 2010. Sex, mixability, and modularity. *Proceedings of the National Academy of Sciences* 107(4):1452–1457.

Livnat, A.; Papadimitriou, C. H.; Rubinstein, A.; Valiant, G.; and Wan, A. 2014. Satisfiability and Evolution. In *FOCS*, 524–530.

Mahloujifar, S.; Diochnos, D. I.; and Mahmoody, M. 2017. Learning under p-Tampering Attacks. *CoRR* abs/1711.03707. To appear in ISAIM 2018.

Mansour, Y., and Parnas, M. 1998. Learning Conjunctions with Noise under Product Distributions. *Information Processing Letters* 68(4):189–196.

Michael, L. 2012. Evolvability via the Fourier transform. *Theoretical Computer Science* 462:88–98.

Mitchell, T. M. 1977. Version spaces: A candidate elimination approach to rule learning. In *IJCAI*, 305–310.

Mitchell, T. M. 1997. *Machine learning*. McGraw Hill series in computer science. McGraw-Hill.

Mitchell, M. 1998. *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press.

Prugel-Bennett, A.; Rowe, J.; and Shapiro, J. 2015. Run-Time Analysis of Population-Based Evolutionary Algorithm in Noisy Environments. In *FOGA*, 69–75.

Qian, C.; Bian, C.; Jiang, W.; and Tang, K. 2017. Running Time Analysis of the (1+1)-EA for OneMax and LeadingOnes under Bitwise Noise. In *GECCO*, 1399–1406.

Quinlan, J. R. 1986. The Effect of Noise on Concept Learning. In Michalski, R. S.; Carbonell, J. G.; and Mitchell, T. M., eds., *Machine Learning, An Artificial Intelligence Approach (Volume II)*. Los Alamos, CA: Morgan Kaufmann. chapter 6, 149–166.

Reischuk, R., and Zeugmann, T. 1999. A Complete and Tight Average-Case Analysis of Learning Monomials. In *STACS*, 414–423. Springer.

Sakai, Y., and Maruoka, A. 2000. Learning Monotone Log-Term DNF Formulas under the Uniform Distribution. *Theory of Computing Systems* 33(1):17–33.

Sellie, L. 2008. Learning Random Monotone DNF Under the Uniform Distribution. In *COLT*, 181–192.

Sellie, L. 2009. Exact learning of random DNF over the uniform distribution. In *STOC*, 45–54.

Servedio, R. A. 1999. On PAC Learning Using Winnow, Perceptron, and a Perceptron-like Algorithm. In *COLT*, 296–307.

Servedio, R. A. 2004. On learning monotone DNF under product distributions. *Information and Computation* 193(1):57–74.

Shackelford, G., and Volper, D. 1988. Learning k-DNF with Noise in the Attributes. In *COLT*, 97–103.

Simon, H. U. 2009. Smart PAC-Learners. In *ALT*, 353–367.

Simon, H. U. 2014. PAC-learning in the presence of one-sided classification noise. *Annals of Mathematics and Artificial Intelligence* 71(4):283–300.

Sloan, R. H. 1995. Four Types of Noise in Data for PAC Learning. *Information Processing Letters* 54(3):157–162.

Sudholt, D. 2010. General Lower Bounds for the Running Time of Evolutionary Algorithms. In *PPSN*, 124–133.

Szörényi, B. 2009. Characterizing Statistical Query Learning: Simplified Notions and Proofs. In *ALT*, 186–200.

Valiant, L. G. 1984. A Theory of the Learnable. *Communications of the ACM* 27(11):1134–1142.

Valiant, L. G. 1985. Learning disjunctions of conjunctions. In *IJCAI*, 560–566.

Valiant, L. G. 2009. Evolvability. *Journal of the ACM* 56(1):3:1–3:21.

Valiant, L. 2013. *Probably Approximately Correct: Nature's Algorithms for Learning and Prospering in a Complex World*. New York, NY, USA: Basic Books, Inc.

Valiant, P. 2014. Evolvability of Real Functions. *ACM Transactions on Computation Theory* 6(3):12:1–12:19.

Verbeurgt, K. A. 1998. Learning Sub-classes of Monotone DNF on the Uniform Distribution. In *ALT*, 385–399.

Watson, R. A., and Szathmáry, E. 2016. How can evolution learn? *Trends in Ecology & Evolution* 31(2):147–157.

Wegener, I., and Witt, C. 2005. On the analysis of a simple evolutionary algorithm on quadratic pseudo-boolean functions. *Journal of Discrete Algorithms* 3(1):61–78.

Wegener, I. 2001. Theoretical Aspects of Evolutionary Algorithms. In *ICALP*, 64–78.

Wegener, I. 2003. Methods for the Analysis of Evolutionary Algorithms on Pseudo-Boolean Functions. In *Evolutionary Optimization*. Springer. 349–369.