INDEX TERMS: Statistical query, PAC learning, classification noise, noise-tolerant learning, SQ dimension.

1 PROBLEM DEFINITION

The problem deals with learning to classify from random labeled examples in Valiant’s PAC model [30]. In the random classification noise model of Angluin and Laird [1] the label of each example given to the learning algorithm is flipped randomly and independently with some fixed probability \( \eta \) called the noise rate. Robustness to such benign form of noise is an important goal in the design of learning algorithms. Kearns defined a powerful and convenient framework for constructing noise-tolerant algorithms based on statistical queries. Statistical query (SQ) learning is a natural restriction of PAC learning that models algorithms that use statistical properties of a data set rather than individual examples. Kearns demonstrated that any learning algorithm that is based on statistical queries can be automatically converted to a learning algorithm in the presence of random classification noise of arbitrary rate smaller than the information-theoretic barrier of \( 1/2 \). This result was used to give the first noise-tolerant algorithm for a number of important learning problems. In fact, virtually all known noise-tolerant PAC algorithms were either obtained from SQ algorithms or can be easily cast into the SQ model.

In subsequent work the model of Kearns has been extended to other settings and found a number of additional applications in machine learning and theoretical computer science.

1.1 Definitions and Notation

Let \( \mathcal{C} \) be a class of \( \{-1, +1\} \)-valued functions (also called concepts) over an input space \( X \). In the basic PAC model a learning algorithm is given examples of an unknown function \( f \) from \( \mathcal{C} \) on points randomly chosen from some unknown distribution \( D \) over \( X \) and should produce a hypothesis \( h \) that approximates \( f \). More formally, an example oracle \( \text{EX}(f, D) \) is an oracle that upon being invoked returns an example \( \langle x, f(x) \rangle \), where \( x \) is chosen randomly with respect to \( D \), independently of any previous examples. A learning algorithm for \( \mathcal{C} \) is an algorithm that for every \( \epsilon > 0 \), \( \delta > 0 \), \( f \in \mathcal{C} \), and distribution \( D \) over \( X \), given \( \epsilon, \delta \), and access to \( \text{EX}(f, D) \) outputs, with probability at least \( 1 - \delta \), a hypothesis \( h \) that \( c \)-approximates \( f \) with respect to \( D \) (i.e. \( \Pr_{D}[f(x) \neq h(x)] \leq \epsilon \)).

Efficient learning algorithms are algorithms that run in time polynomial in \( 1/\epsilon \), \( 1/\delta \), and the size of the learning problem \( s \). The size of a learning problem is determined by the description length of \( f \) under some fixed representation scheme for functions in \( \mathcal{C} \) and the description length of an element in \( X \) (often proportional to the dimension \( n \) of the input space).

A number of variants of this basic framework are commonly considered. The basic PAC model is also referred to as distribution-independent learning to distinguish it from distribution-specific PAC learning in which the learning algorithm is required to learn with respect to a single distribution \( D \).
known in advance. A weak learning algorithm is a learning algorithm that can produce a hypothesis whose error on the target concept is noticeably less than 1/2 (not necessarily any ε > 0). More precisely, a weak learning algorithm produces a hypothesis h such that Pr[f(x) ≠ h(x)] ≤ 1/2 – 1/p(s) for some fixed polynomial p. The basic PAC model is often referred to as strong learning in this context.

In the random classification noise model EX(f, D) is replaced by a faulty oracle EXη(f, D), where η is the noise rate. When queried, this oracle returns a noisy example (x, b) where b = f(x) with probability 1 – η and –f(x) with probability η independently of previous examples. When η approaches 1/2 the label of the corrupted example approaches the result of a random coin flip, and therefore the running time of learning algorithms in this model is allowed to depend on 1/(1−2η) (the dependence must be polynomial for the algorithm to be considered efficient). For simplicity one usually assumes that η is known to the learning algorithm. This assumption can be removed using a simple technique due to Laird [26].

To formalize the idea of learning from statistical properties of a large number of examples, Kearns introduced a new oracle STAT(f, D) that replaces EX(f, D). The oracle STAT(f, D) takes as input a statistical query (SQ) of the form (χ, τ) where χ is a {−1, +1}-valued function on labeled examples and τ ∈ [0, 1] is the tolerance parameter. Given such a query the oracle responds with an estimate v of PrD[χ(x, f(x)) = 1] that is accurate to within an additive ±τ.

Note that the oracle does not guarantee anything else on the value v beyond |v − PrD[χ(x, f(x)) = 1]| ≤ τ and an SQ learning algorithm needs to work with any possible implementation of the oracle. Yang proposed a stronger, honest version of the oracle which to a call with function χ returns the value of χ(x, f(x)), where x is chosen randomly and independently according to D [32]. This version was shown to be equivalent to the original model up to polynomial factors [18].

Chernoff bounds easily imply that STAT(f, D) can, with high probability, be simulated using EX(f, D) by estimating PrD[χ(x, f(x)) = 1] on O(τ−2) examples. Therefore the SQ model is a restriction of the PAC model. Efficient SQ algorithms allow only efficiently evaluable χ’s and impose an inverse polynomial lower bound on the tolerance parameter over all oracle calls. Kearns also observes that in order to simulate all the statistical queries used by an algorithm one does not necessarily need new examples for each estimation. Instead, assuming that the set of possible queries of the algorithm has Vapnik-Chervonenkis dimension d, all its statistical queries can be simulated using O(dτ−2(1 − 2η)−2 log (1/δ)) examples [24].

2 KEY RESULTS

2.1 Statistical Queries and Noise-tolerance

The main result given by Kearns is a way to simulate statistical queries using noisy examples.

Lemma 1 ([24]). Let (χ, τ) be a statistical query such that χ can be evaluated on any input in time T and let EXη(f, D) be a noisy oracle. The value PrD[χ(x, f(x)) = 1] can, with probability at least 1 − δ, be estimated within τ using O(τ−2(1 − 2η)−2 log (1/δ)) examples from EXη(f, D) and time O(τ−2(1 − 2η)−2 log (1/δ) · T).

This simulation is based on estimating several probabilities using examples from the noisy oracle and then offsetting the effect of noise. The lemma implies that any efficient SQ algorithm for a concept class C can be converted to an efficient learning algorithm for C tolerating random classification noise of any rate η < 1/2.

Theorem 2 ([24]). Let C be a concept class efficiently PAC learnable from statistical queries. Then C is efficiently PAC learnable in the presence of random classification noise of rate η for any η < 1/2.
Balcan and Feldman describe more general conditions on noise under which a specific SQ algorithm can be simulated in the presence of noise [3].

2.2 Statistical Query Algorithms

Kearns showed that, despite the major restriction on the way an SQ algorithm accesses the examples, many PAC learning algorithms known at the time can be modified to use statistical queries instead of random examples [24]. Examples of learning algorithms for which he described an SQ analogue and thereby obtained a noise-tolerant learning algorithm include:

- Learning decision trees of constant rank.
- Attribute-efficient algorithms for learning conjunctions.
- Learning axis-aligned rectangles over $\mathbb{R}^n$.
- Learning $\text{AC}^0$ (constant-depth unbounded fan-in) Boolean circuits over $\{0, 1\}^n$ with respect to the uniform distribution in quasipolynomial time.

Subsequent works have provided numerous additional examples of algorithms used in theory and practice of machine learning that can either be implemented using statistical queries or can be replaced by an alternative SQ-based algorithm of similar complexity. For example the Perceptron algorithm and learning of linear threshold functions [6, 12], boosting [2], attribute-efficient learning via the Winnow algorithm (cf. [17]), $k$-means clustering [5] and convex optimization-based methods [20]. We note that many learning algorithms rely only on evaluations of functions on random examples and therefore can be seen as using access to the honest statistical query oracle. In such cases the SQ implementation follows immediately from the equivalence of the Kearns' SQ oracle and the honest one [18].

The only known example of a technique for which there is no SQ analogue is Gaussian elimination for solving linear equations over a finite field. This technique can be used to learn parity functions that are not learnable using SQs (as we discuss below). As a result, with the exception of the parity learning problem, known bounds on the complexity of learning from random examples are, up to polynomial factors, the same as known bound for learning with statistical queries.

2.3 Statistical Query Dimension

The restricted way in which SQ algorithms use examples makes it simpler to understand the limitations of efficient learning in this model. A long-standing open problem in learning theory is learning of the concept class of all parity functions over $\{0, 1\}^n$ with noise (a parity function is a XOR of some subset of $n$ Boolean inputs). Kearns has demonstrated that parities cannot be efficiently learned using statistical queries even under the uniform distribution over $\{0, 1\}^n$ [24]. This hardness result is unconditional in the sense that it does not rely on any unproven complexity assumptions.

The technique of Kearns was generalized by Blum et al. who proved that efficient SQ learnability of a concept class $\mathcal{C}$ is characterized by a relatively simple combinatorial parameter of $\mathcal{C}$ called the statistical query dimension [7]. The quantity they defined measures the maximum number of “nearly uncorrelated” functions in a concept class. (The definition and the results were simplified and strengthened in subsequent works [18, 29] and we use the improved statements here.) More formally,

**Definition 3.** For a concept class $\mathcal{C}$ and distribution $\mathcal{D}$, the statistical query dimension of $\mathcal{C}$ with respect to $\mathcal{D}$, denoted $\text{SQ-DIM}(\mathcal{C}, \mathcal{D})$, is the largest number $d$ such that $\mathcal{C}$ contains $d$ functions $f_1, f_2, \ldots, f_d$ such that for all $i \neq j$, $|\mathbb{E}_{\mathcal{D}}[f_if_j]| \leq \frac{1}{d}$.
Blum et al. relate the SQ dimension to learning in the SQ model as follows.

**Theorem 4** ([7, 18]). Let $C$ be a concept class and $D$ be a distribution such that $\text{SQ-DIM}(C, D) = d$.

- If all queries are made with tolerance of at least $1/d^{1/3}$, then at least $d^{1/3} - 2$ queries are required to learn $C$ with error $1/2 - 1/(2d^3)$ in the SQ model.

- There exists an algorithm for learning $C$ with respect to $D$ that makes $d$ fixed queries, each of tolerance $1/(4d)$, and finds a hypothesis with error at most $1/2 - 1/(2d)$.

Thus SQ-DIM characterizes weak SQ learnability relative to a fixed distribution $D$ up to a polynomial factor. Parity functions are uncorrelated with respect to the uniform distribution and therefore any concept class that contains a superpolynomial number of parity functions cannot be learned by statistical queries with respect to the uniform distribution. This, for example, includes such important concept classes as $k$-juntas over $\{0, 1\}^n$ (or functions that depend on at most $k$ input variables) for $k = \omega(1)$ and decision trees of superconstant size.

Simon showed that (strong) PAC learning relative to a fixed distribution $D$ using SQs can also be characterized by a more general and involved dimension [28]. Simpler and tighter characterizations of distribution-specific PAC learning using SQs have been demonstrated by Feldman [15] and Szörényi [29]. Feldman also extended the characterization to the agnostic learning model.

Despite characterizing the number of queries of certain tolerance, the SQ-DIM and its generalizations capture surprisingly well the computational complexity of SQ learning of most concept classes. One reason for this is that if a concept class has polynomial SQ-DIM then it can be learned by a polynomial-time algorithm with advice also referred to as a “non-uniform” algorithm (cf. [19]). However it was shown by Feldman and Kanade that for strong PAC learning there exist artificial problems whose computational complexity is larger than their statistical query complexity [19].

Applications of these characterizations to proving lower bounds on SQ algorithms can be found in [11, 15, 16, 25]. Relationships of SQ-DIM to other notions of complexity of concept classes were investigated in [22, 27].

### 3 APPLICATIONS

The ideas behind the use of statistical queries to produce noise tolerant algorithms were adapted to learning using membership queries (or ability to ask for the value of the unknown function at any point) and used to give a noise-tolerant algorithm for learning DNF with respect to the uniform distribution [9, 21]. The SQ model of learning was generalized to active learning (or learning where labels are requested only for some of the points) and used to obtain new efficient noise tolerant active learning algorithms [3].

The restricted way in which an SQ algorithm uses data implies can it be used to obtain learning algorithms with additional useful properties. Blum et al. [5] show that an SQ algorithm can be used to obtain a differentially-private [13] algorithm for the problem. In fact, SQ algorithms are equivalent to local (or randomized-response) differentially-private algorithms [23]. Chu et al. [10] show that SQ algorithms can be automatically parallelized on multicore architectures and give many examples of popular machine learning algorithms that can be sped up using this approach.

The SQ learning model has also been instrumental in understanding Valiant’s model of evolution as learning [31]. Feldman showed that the model is equivalent to learning with a restricted form of SQs referred to as correlational SQs [14]. A correlational SQ is a query of the form $\chi(x, \ell) = g(x) \cdot \ell$ for some $g : X \to [-1, 1]$. Such queries were first studied by Ben-David et al. [4] (remarkably, before the introduction of the SQ model itself) and distribution-specific learning with such queries is equivalent to learning with (unrestricted) SQs.

Statistical query-based access can naturally be defined for any problem where the input is a set of i.i.d. samples from a distribution. Feldman et al. show that lower bounds based on SQ-DIM can be extended to this more general setting and give examples of applications [18, 20].
4 OPEN PROBLEMS

The main questions related to learning with random classification noise are still open. Is every concept class efficiently learnable in the PAC model also learnable in the presence of random classification noise? Is every concept class efficiently learnable in the presence of random classification noise of arbitrarily high rate (less than 1/2) also efficiently learnable using statistical queries? A partial answer to this question was provided by Blum et al. who show that Gaussian elimination can be used in low dimension to obtain a class learnable with random classification noise of constant rate $\eta < 1/2$ but not learnable using SQs [8]. For both questions a central issue seems to be obtaining a better understanding of the complexity of learning parities with noise.

The complexity of learning from statistical queries remains an active area of research with many open problems. For example there is currently an exponential gap between known lower and upper bounds on the complexity of distribution-independent SQ learning of polynomial-size DNF formulae and AC$^0$ circuits (cf. [27]). Several additional open problems on complexity of SQ learning can be found in [16, 17, 22].

5 EXPERIMENTAL RESULTS

None is reported.

6 DATA SETS

None is reported.

7 URL to CODE

None is reported.

8 CROSS REFERENCES

None is reported. (Fill in once it’s clear what the entries will be.)

9 RECOMMENDED READING


