A Tutorial on Statistical Queries

Lev Reyzin

University of Illinois at Chicago
Department of Mathematics (MSCS)

April 8, 2018
Algorithmic Learning Theory

Table of contents

An Introduction to Statistical Query (SQ) Learning

Why Statistical Queries?

SQs have many connections to a variety of modern topics, including to evolvability, differential privacy, adaptive data analysis, and deep learning. SQ has become both an important tool and remains a foundational topic with many important questions.

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

Why Statistical Queries?

SQs have many connections to a variety of modern topics, including to evolvability, differential privacy, adaptive data analysis, and deep learning. SQ has become both an important tool and remains a foundational topic with many important questions.

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

Why Statistical Queries?

SQs have many connections to a variety of modern topics, including to evolvability, differential privacy, adaptive data analysis, and deep learning. SQ has become both an important tool and remains a foundational topic with many important questions.

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

Independently by:
Shai Ben-David, Alon Itai, Eyal Kushilevitz. Learning by Distances. *Information and Computation 117(2), pp. 240-250.* 1995.

Definitions

SQ as a restriction of PAC

Definition (efficient PAC learning)

Let C be a class of boolean functions $c : X \to \{-1, 1\}$. C is efficiently PAC-learnable if there exists an algorithm L such that for every $c \in C$, any probability distribution D_X over X, and any $0 < \epsilon, \delta < 1$, algorithm L takes a labeled sample S of size $m = \mathrm{poly}(1/\epsilon, 1/\delta, n, |c|)$ from[a] D, and in time polynomial in m, outputs a hypothesis h for which $\mathbf{Pr}_{S \sim D}[\mathrm{err}_D(h) \leq \epsilon] \geq 1 - \delta$.

[a] $n = |x|$

SQ learning is a variant PAC, which gives the learner access to an *oracle instead of labeled examples*.

SQ oracle

Definition (statistical query)

A statistical query is a pair (q, τ) with

- q : a function $q : X \times \{-1, 1\} \to \{-1, 1\}$.
- τ : a tolerance parameter $\tau \geq 0$.

Definition (statistical query oracle)

the statistical query oracle $\mathrm{SQ}(q, \tau)$ returns a value in the range:

$$\left[\mathbf{E}_{x \sim D}[q(x, c(x))] - \tau, \mathbf{E}_{x \sim D}[q(x, c(x))] + \tau\right].$$

Efficient SQ learning

Definition (efficient SQ learning)

Let C be a class of boolean functions $c : X \to \{-1, 1\}$. C is efficiently SQ-learnable if there exists an algorithm L such that for every $c \in C$, any probability distribution D, and any $\epsilon > 0$, there is a polynomial $p(\cdot, \cdot, \cdot)$ such that

- L makes at most $p(1/\epsilon, n, |c|)$ calls to the SQ oracle
- the smallest τ that L uses satisfies $\frac{1}{\tau} \leq p(1/\epsilon, n, |c|)$, and
- the queries q are evaluable in time $p(1/\epsilon, n, |c|)$,

and L outputs a hypothesis h satisfying $\mathrm{err}_D(h) \leq \epsilon$.

Note this definition has no failure parameter δ.

SQ vs. PAC

SQ learnability implies PAC learnability

SQ is a natural restriction of PAC.

Observation

If a class of functions is efficiently SQ-learnable, then it is efficiently learnable in the PAC model.

Proof.

You can simulate an SQ oracle in the PAC model by drawing $O\left(\frac{\log(k/\delta)}{\tau^2}\right)$ samples for each of the k statistical queries, and by the Hoeffding bound, the simulation will fail with probability $< \delta$. $\square$

SQ learnability implies noisy PAC learnability

SQ-learnability is also related to learnability under the classification noise model of Angluin and Laird ('87).

Definition (classification noise)

A PAC learning algorithm under random classification noise (η-PAC), aka "white-label noise," must meet the PAC requirements, but the label of each training sample is flipped with independently with probability η, for $0 \leq \eta < 1/2$. The sample size and running time also include a polynomial dependence on $1/(1-2\eta)$.

SQ learnability implies noisy PAC learnability

Theorem (Kearns '98)

If a class of functions is efficiently SQ-learnable, then it is efficiently learnable in the noisy PAC model.

Proof sketch.

1. Draw enough examples, $\text{poly}\left(\frac{1}{\tau}, \frac{1}{1-2\eta}, \log\frac{1}{\delta}\right)$ suffice.
2. Separate data into part on which q is affected by noise and part that's not.
3. Estimate q on both parts, then "undo" noise on noisy part.

   e.g. for the noisy part, $P = (P_\eta - \eta)/(1 - 2\eta)$. □

SQ Learnability Implies Noisy PAC Learnability – proof

Theorem (Kearns '98)

If a class of functions is efficiently SQ-learnable, then it is efficiently learnable in the PAC model under classification noise.

So, the SQ framework gives us a way to design algorithms that are also noise-tolerant.

SQ learnability also gives results for learning in Valiant's ('85) malicious noise model.

Theorem (Aslam and Decatur '98)

If a class of functions is efficiently SQ-learnable, then it is efficiently PAC learnable under malicious noise with noise rate $\eta = \tilde{O}(\epsilon)$.

Variants of SQs

Correlational and honest queries

Bshouty and Feldman ('01) defined correlational statistical queries:

Definition (correlational statistical query oracle)

Given a function $h = X \to \{-1, 1\}$ and a tolerance parameter τ, the correlational statistical query oracle $\mathrm{CSQ}(h, \tau)$ returns a value within τ of $\mathbf{E}_D[h(x)c(x)]$.

Note CSQ = "Learning by Distances" (Ben-David, Itai, Kushilevitz '95).

Correlational and honest queries

Bshouty and Feldman ('01) defined correlational statistical queries:

Definition (correlational statistical query oracle)

Given a function $h = X \rightarrow \{-1, 1\}$ and a tolerance parameter τ, the correlational statistical query oracle $\mathrm{CSQ}(h, \tau)$ returns a value within τ of $\mathbf{E}_D[h(x)c(x)]$.

Note CSQ = "Learning by Distances" (Ben-David, Itai, Kushilevitz '95).

Yang ('05) defined honest statistical queries:

Definition (honest statistical query oracle)

Given function $q : X \times \{-1, 1\} \rightarrow \{-1, 1\}$ and sample size s, the honest statistical query oracle $\mathrm{HSQ}(q, s)$ draws $x_1, \ldots, x_s \sim D$ and returns $\frac{1}{s} \sum_{i=1}^{s} q(x_i, c(x_i))$.

Bounds for SQ algorithms

Statistical query dimension

Limitations of SQ algorithms

A quantity called the statistical query dimension (Blum, Furst, Jackson, Kearns, Mansour, Rudich '94) controls the complexity of statistical query learning.

Definition (statistical query dimension)

For a concept class C and distribution D, the statistical query dimension of C with respect to D, denoted $\mathrm{SQ\text{-}DIM}_D(C)$, is the largest number d such that C contains d functions $f_1, f_2, \ldots, f_d$ such that for all $i \neq j$, $\left|\langle f_i, f_j \rangle_D\right| \leq 1/d$. Note: $\langle f_i, f_j \rangle_D = \mathbf{E}_D[f_i \cdot f_j]$.

Sometimes, we leave out the distribution, in which case we mean:

$$\mathrm{SQ\text{-}DIM}(C) = \max_{D \in \mathcal{D}} \mathrm{SQ\text{-}DIM}_D(C).$$

Theorem (Blum, Furst, Jackson, Kearns, Mansour, Rudich '94)

Let C be a concept class and let $d = \mathrm{SQ\text{-}DIM}_D(C)$. Then any SQ learning algorithm that uses a tolerance parameter lower bounded by $\tau > 0$ must make at least $(d\tau^2 - 1)/2$ queries to learn C with accuracy at least τ. In particular, when $\tau = 1/d^{1/3}$, this means $(d^{1/3} - 1)/2$ queries are needed.

Corollary

Let C be a class with $\mathrm{SQ\text{-}DIM}_D(C) = \omega(n^k)$ for all k, then C is not efficiently SQ-learnable under D.

Theorem (Blum, Furst, Jackson, Kearns, Mansour, Rudich '94)

Let C be a concept class and let $d = \mathrm{SQ\text{-}DIM}_D(C)$. Then any ~~SQ~~ CSQ learning algorithm that uses a tolerance parameter lower bounded by $\tau > 0$ must make at least $(d\tau^2 - 1)/2$ queries to learn C with accuracy at least τ. In particular, when $\tau = 1/d^{1/3}$, this means $(d^{1/3} - 1)/2$ queries are needed.

Proof.

The original proof is a bit too technical to present here, so instead we'll see a clever, short proof of this lower bound for CSQs. □

proof (Szörényi '09).

Assume $f_1, \ldots, f_d$ realize the SQ-DIM. Let h be a query and $A = \{i \in [d] : \langle f_i, h\rangle \geq \tau\}$. Then by Cauchy-Schwartz, we have

$$\left\langle h, \sum_{i \in A} f_i \right\rangle^2 \leq \left|\left|\sum_{i \in A} f_i\right|\right|^2 = \sum_{i,j \in A} \langle f_i, f_j\rangle \leq \sum_{i \in A}\left(1 + \frac{|A| - 1}{d}\right),$$

so $\langle h, \sum_{i \in A} f_i\rangle^2 \leq |A| + \frac{|A|^2}{d}$. But by definition of A, we also have $\langle h, \sum_{i \in A} f_i\rangle \geq |A|\tau$. By algebra, $|A| \leq d/(d\tau^2 - 1)$, and the same bound holds for A' defined w.r.t. correlation $\leq -\tau$.

So no matter what h, an answer of 0 to CSQ(h, τ) eliminates at most $d/(|A| + |A'|) = (d\tau^2 - 1)/2$ functions. $\square$

Perhaps surprisingly, for distribution-specific learning, CSQ-learnability is equivalent to SQ-learnability.

Lemma (Bshouty, Feldman '02)

Any SQ can be answered by asking two SQs that are independent of the target and two CSQs.

$$\begin{aligned}\mathbf{E}_D[q(x, c(x))] &= \mathbf{E}_D\left[q(x,-1)\frac{1-c(x)}{2} + q(x,1)\frac{1+c(x)}{2}\right]\\ &= \frac{1}{2}\mathbf{E}_D[q(x,1)c(x)] - \frac{1}{2}\mathbf{E}_D[q(x,-1)c(x)]\\ &\quad + \frac{1}{2}\mathbf{E}_D[q(x,1)] + \frac{1}{2}\mathbf{E}_D[q(x,-1)].\end{aligned}$$

Perhaps surprisingly, for distribution-specific learning, CSQ-learnability is equivalent to SQ-learnability.

Lemma (Bshouty, Feldman '02)

Any SQ can be answered by asking two SQs that are independent of the target and two CSQs.

$$\begin{aligned}\mathbf{E}_D[q(x,c(x))] &= \mathbf{E}_D\left[q(x,-1)\frac{1-c(x)}{2} + q(x,1)\frac{1+c(x)}{2}\right] \\ &= \frac{1}{2}\mathbf{E}_D[q(x,1)c(x)] - \frac{1}{2}\mathbf{E}_D[q(x,-1)c(x)] \\ &\quad + \frac{1}{2}\mathbf{E}_D[q(x,1)] + \frac{1}{2}\mathbf{E}_D[q(x,-1)].\end{aligned}$$

On the other hand, Feldman (2011) showed that CSQs are strictly weaker than SQs for distribution-independent learning. E.g. half-spaces are not distribution-independently CSQ learnable, but are SQ learnable.

Theorem (Blum et al. '94; Szörényi '09)

Let C be a concept class and let $d = \mathrm{SQ\text{-}DIM}(C)$. Then any SQ ($\therefore$ CSQ) learning algorithm that uses a tolerance parameter lower bounded by $\tau > 0$ must make at least $(d\tau^2 - 1)/2$ queries to learn C with accuracy at least τ. In particular, when $\tau = 1/d^{1/3}$, this means $(d^{1/3} - 1)/2$ queries are needed.

Theorem (Yang '05; Feldman, Grigorescu, Reyzin, Vempala, Xiao '17)

Let C be a concept class and let $d = \mathrm{SQ\text{-}DIM}(C)$. Then any HSQ learning algorithm must use a total sample complexity at least $\Omega(d)$ to learn C (to constant accuracy and probability of success).

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
  - known from orthogonality of Fourier characters under the uniform distribution; see O'Donnell ('09)

Classes that are not SQ learnable

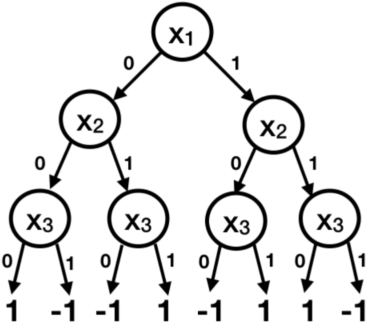
- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
  - known from orthogonality of Fourier characters under the uniform distribution; see O'Donnell ('09)
  - parities are PAC-learnable, so SQ $\subsetneq$ PAC

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
  - known from orthogonality of Fourier characters under the uniform distribution; see O'Donnell ('09)
  - parities are PAC-learnable, so SQ $\subsetneq$ PAC
  - this implies: VC-DIM$(C) \leq$ SQ-DIM(C), but SQ-DIM(C) can also be exponentially large in VC-DIM(C) (Blum, Furst, Jackson, Kearns, Mansour, Rudich '94)

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)

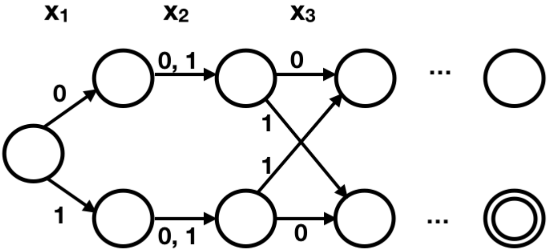


Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)
- DNF (SQ-DIM $\geq n^{c \log n}$)
  $(x_1 \wedge x_2 \wedge x_3) \vee (\bar{x}_1 \wedge \bar{x}_2 \wedge \bar{x}_3) \vee (\bar{x}_1 \wedge x_2 \wedge \bar{x}_3) \vee (\bar{x}_1 \wedge \bar{x}_2 \wedge x_3)$

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)
- DNF (SQ-DIM $\geq n^{c \log n}$)
- finite automata (SQ-DIM $\geq 2^{cn}$)



Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)
- DNF (SQ-DIM $\geq n^{c \log n}$)
- finite automata (SQ-DIM $\geq 2^{cn}$)
- etc.

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)
- DNF (SQ-DIM $\geq n^{c \log n}$)
- finite automata (SQ-DIM $\geq 2^{cn}$)
- etc.
- even *uniformly random* decision trees, DNF, and automata (Angluin, Eisenstat, Kontorovich, Reyzin '10)

Classes that are not SQ learnable

- parity functions, $\chi_c(x) = (-1)^{c \cdot x}$ (SQ-DIM $= 2^n$)
- decision trees (SQ-DIM $\geq n^{c \log n}$)
- DNF (SQ-DIM $\geq n^{c \log n}$)
- finite automata (SQ-DIM $\geq 2^{cn}$)
- etc.
- even *uniformly random* decision trees, DNF, and automata (Angluin, Eisenstat, Kontorovich, Reyzin '10)

Note that only the first of these are known to be PAC learnable. We'll come back to this later.

Weak learning

Theorem

Let C be a concept class and let $\mathrm{SQ\text{-}DIM}_D(C) = \mathrm{poly}(n)$, then C is weakly learnable under D.

Proof.

Let $S = \{f_1, \dots f_d\} \subseteq C$ realize the SQ bound. For each $f_i \in S$, query its correlation with c^*. At least one has a correlation $> 1/d$ (otherwise we could add c^* to S, contradicting S's maximality). □

Because of this observation, $\mathrm{SQ\text{-}DIM}$ is sometimes referred to as the *weak* statistical query dimension.

Strong vs weak SQ learning

Schapire ('90) showed that "weak learning" = "strong learning" in the PAC setting. Is the same true in the SQ setting?

Strong vs weak SQ learning

Schapire ('90) showed that "weak learning" = "strong learning" in the PAC setting. Is the same true in the SQ setting?

Yes! Aslam and Decator ('98) showed SQ boosting is possible.

Theorem (Aslam, Decatur '98)

Let $d = \mathrm{SQ\text{-}DIM}(C)$, then C is SQ-learnable to error $\epsilon > 0$ using $O(d^5 \log^2 \frac{1}{\epsilon})$ queries with tolerances bounded by $\tau = \Omega(\frac{\epsilon}{3d})$.

Strong vs weak SQ learning

Schapire ('90) showed that "weak learning" = "strong learning" in the PAC setting. Is the same true in the SQ setting?

Yes! Aslam and Decator ('98) showed SQ boosting is possible.

Theorem (Aslam, Decatur '98)

Let $d = \mathrm{SQ\text{-}DIM}(C)$, then C is SQ-learnable to error $\epsilon > 0$ using $O(d^5 \log^2 \frac{1}{\epsilon})$ queries with tolerances bounded by $\tau = \Omega(\frac{\epsilon}{3d})$.

But this is for distribution *independent* learning.

strong statistical query dimension

In the distribution-dependent case, (weak) SQ dimension does not characterize strong learnability.

For this reason, there exists the notion of strong SQ dimension (Simon '07; Feldman '09; Szörényi '09).

Definition (strong statistical query dimension)

For a concept class C and distribution D, let the strong statistical query dimension $\mathrm{SSQ\text{-}DIM}_D(C, \gamma)$ be the largest d such that some $f_1, \ldots, f_d \in C$ fulfill

- $|\langle f_i, f_j \rangle_D| \leq \gamma$ for $1 \leq i < j \leq d$, and
- $|\langle f_i, f_j \rangle_D - \langle f_k, f_\ell \rangle_D| \leq 1/d$ for $1 \leq i < j \leq d$, $1 \leq k < \ell \leq d$.

Strong SQ learning

Definition (strong statistical query dimension)

For a concept class C and distribution D, let the strong statistical query dimension $\mathrm{SSQ\text{-}DIM}_D(C, \gamma)$ be the largest d such that some $f_1, \ldots, f_d \in C$ fulfill

- $|\langle f_i, f_j \rangle_D| \leq \gamma$ for $1 \leq i < j \leq d$, and
- $|\langle f_i, f_j \rangle_D - \langle f_k, f_\ell \rangle_D| \leq 1/d$ for $1 \leq i < j \leq d$, $1 \leq k < \ell \leq d$.

Roughly, $\mathrm{SSQ\text{-}DIM}_D(C, 1-\epsilon)$, controls the complexity of learning C to error ϵ under D.

Strong SQ learning

Definition (strong statistical query dimension)

For a concept class C and distribution D, let the strong statistical query dimension $\mathrm{SSQ\text{-}DIM}_D(C, \gamma)$ be the largest d such that some $f_1, \ldots, f_d \in C$ fulfill

- $|\langle f_i, f_j \rangle_D| \leq \gamma$ for $1 \leq i < j \leq d$, and
- $|\langle f_i, f_j \rangle_D - \langle f_k, f_\ell \rangle_D| \leq 1/d$ for $1 \leq i < j \leq d$, $1 \leq k < \ell \leq d$.

Roughly, $\mathrm{SSQ\text{-}DIM}_D(C, 1-\epsilon)$, controls the complexity of learning C to error ϵ under D.

For $\epsilon = 1/10$, the gap between strong and weak SQ dimension can be as large as possible, e.g. consider $\mathcal{F} = \{v_1 \vee \chi_c \mid c \in \{0,1\}^n\}$; then $\mathrm{SQ\text{-}DIM}_U(\mathcal{F}) = 0$ but $\mathrm{SSQ\text{-}DIM}_U(\mathcal{F}, 9/10) = 2^n$.

Strong SQ learning

Definition (strong statistical query dimension)

For a concept class C and distribution D, let the strong statistical query dimension $\mathrm{SSQ\text{-}DIM}_D(C, \gamma)$ be the largest d such that some $f_1, \ldots, f_d \in C$ fulfill

- $|\langle f_i, f_j \rangle_D| \leq \gamma$ for $1 \leq i < j \leq d$, and
- $|\langle f_i, f_j \rangle_D - \langle f_k, f_\ell \rangle_D| \leq 1/d$ for $1 \leq i < j \leq d$, $1 \leq k < \ell \leq d$.

Roughly, $\mathrm{SSQ\text{-}DIM}_D(C, 1-\epsilon)$, controls the complexity of learning C.

Feldman ('12) showed that a variant of $\mathrm{SSQ\text{-}DIM}$ captures the complexity of agnostic learning of a hypothesis class, which implies that even agnostically learning conjunctions is not possible with statistical queries

SQ and Learnability

PAC, η-PAC, and SQ

We've seen the following:

- efficient SQ $\subseteq$ efficient η-PAC $\subseteq$ efficient PAC
- parity functions are efficiently PAC learnable, but not efficiently SQ learnable.

Are parity functions learnable in η-PAC?

Noisy parity

Are parity functions learnable in η-PAC?

- Blum, Kalai, and Wasserman ('00) gave a $2^{n/\log n}$ algorithm for learning parities in η-PAC.[1]
- This at least means that the class of parities on the first $k = \log n \log \log n$ bits are efficiently learnable in η-PAC, but not efficiently SQ learnable.

[1] for η constant

Noisy parity

Are parity functions learnable in η-PAC?

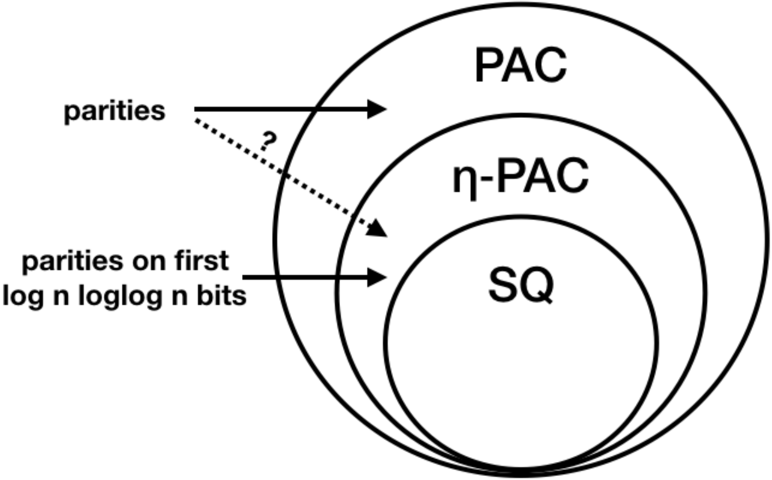
- Blum, Kalai, and Wasserman ('00) gave a $2^{n/\log n}$ algorithm for learning parities in η-PAC.[1]
- This at least means that the class of parities on the first $k = \log n \log\log n$ bits are efficiently learnable in η-PAC, but not efficiently SQ learnable.

This question is the (notorious) "noisy parity problem" (LPN).

- It is assumed there is no efficient algorithm. Variants have been proposed for public-key cryptography (Peikart '14).
- Some progress, but far from efficient algorithms. (Blum, Kalai, Wasserman '00; Grigorescu, Reyzin, Vempala '11; Valiant '12)

[1]for η constant

The big picture



SQ algorithms

On the other hand, we have many methods that can be implemented via the SQ oracle:

SQ algorithms

On the other hand, we have many methods that can be implemented via the SQ oracle:

- gradient descent (Robbins, Monro '51)
- EM (Dempster, Laird, Rubin '77)
- SVM (Cortes, Vapnik '95; Mitra, Murthy, Pal '04)
- linear/convex optimization (Dunagan, Vempala '08)
- MCMC (Tanner, Wong '87; Gelfand, Smith '90)
- simulated annealing (Kirkpatrick, Gelatt, Vecchi '83; Černý '85)
- etc., etc.

SQ algorithms

On the other hand, we have many methods that can be implemented via the SQ oracle:

- gradient descent (Robbins, Monro '51)
- EM (Dempster, Laird, Rubin '77)
- SVM (Cortes, Vapnik '95; Mitra, Murthy, Pal '04)
- linear/convex optimization (Dunagan, Vempala '08)
- MCMC (Tanner, Wong '87; Gelfand, Smith '90)
- simulated annealing (Kirkpatrick, Gelatt, Vecchi '83; Černý '85)
- etc., etc.
- pretty much everything, incl. PCA, ICA, Naïve Bayes, neural net algorithms, k-means (Blum, Dwork, McSherry, Nissim '05)

Non-SQ algorithms

In fact, we basically have only a couple non-SQ algorithms

1. Gaussian elimination
2. hashing/bucketing

Most everything else seems to be SQ.

Non-SQ algorithms

In fact, we basically have only a couple non-SQ algorithms

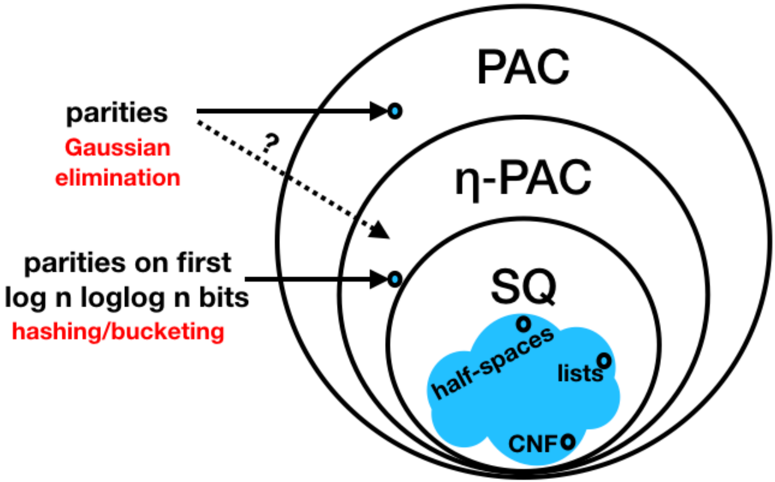
1. Gaussian elimination
2. hashing/bucketing

Most everything else seems to be SQ.

This helps explain why we don't have algorithms for many natural classes, including decision trees and DNF.

To tackle these, we need to invent fundamentally different techniques!

The actual picture



So, what to do for e.g. decision trees?

Applications

Optimization and search over distributions

Introduction to optimization over distributions

Statistical algorithms apply to optimization problems over an unknown distribution D. These are normally solved by working over a sample from D.

As a motivating example, consider the problem of finding the direction that maximizes the rth moment over a distribution D,

$$\mathrm{argmax}_{u:|u|=1} \mathbf{E}_{x\sim D}[(u \cdot x)^r].$$

(This is easy for $r = 1$ and $r = 2$ and probably hard otherwise.)

Introduction to statistical algorithms

Feldman, Grigorescu, Reyzin, Vempala, and Xiao ('17) extended SQs to outside learning. Any problem with instances coming from a distribution D (over X) can be analyzed via a "statistical oracle."

Let $q : X \to \{0, 1\}$, $\tau > 0$ a tolerance, and $t > 0$ a sample size.

Definition (statistical oracles)

- STAT(q, τ): returns a value in: $[\mu - \tau, \mu + \tau]$,
- 1-STAT(q): draws 1 sample, $x \sim D$, and returns $q(x)$,
- VSTAT(q, t): returns a value $[\mu - \tau', \mu + \tau']$,

where $\mu = \mathbf{E}_{x \sim D}[q(x)]$ and $\tau' = \max\left\{1/t, \sqrt{\mu(1-\mu)/t}\right\}$.

Statistical dimension

Definition (pairwise correlation of two distributions)

Define the pairwise correlation of D_1, D_2 with respect to D is

$$\chi_D(D_1, D_2) = \left| \left\langle \frac{D_1}{D} - 1, \frac{D_2}{D} - 1 \right\rangle_D \right|.$$

Note that $\chi_D(D_1, D_1) = \chi^2(D_1, D)$, the chi-squared distance between D_1 and D (Pearson '00).

E.g., let $X = \{0,1\}^n$ and D_{c_1}, D_{c_2} be uniform over the examples labeled -1 by χ_{c_1}, χ_{c_2} resp. It turns out $\chi_U(D_{c_1}, D_{c_2}) = 0$.

Let us compute $\chi_U(D_{010}, D_{011}) = \left\langle \frac{D_{010}}{U} - 1, \frac{D_{011}}{U} - 1 \right\rangle_U$ for $n = 3$.

X	U	D_{010}	D_{011}	$\frac{D_{010}}{U}$	$\frac{D_{011}}{U}$	$\frac{D_{010}}{U} - 1$	$\frac{D_{011}}{U} - 1$
000	1/8	0	0	0	0	-1	-1
001	1/8	0	1/4	0	2	-1	1
010	1/8	1/4	1/4	2	2	1	1
011	1/8	1/4	0	2	0	1	-1
100	1/8	0	0	0	0	-1	-1
101	1/8	0	1/4	0	2	-1	1
110	1/8	1/4	1/4	2	2	1	1
111	1/8	1/4	0	2	0	1	-1

$$\begin{aligned}\left\langle \frac{D_{010}}{U} - 1, \frac{D_{011}}{U} - 1 \right\rangle_U = &\ \frac{(-1)(-1)}{8} + \frac{(-1)(1)}{8} + \frac{(1)(1)}{8} + \frac{(1)(-1)}{8} \\ &+ \frac{(-1)(-1)}{8} + \frac{(-1)(1)}{8} + \frac{(1)(1)}{8} + \frac{(1)(-1)}{8} \\ = 0\end{aligned}$$

Average correlation

Definition (pairwise correlation of two distributions)

Define the pairwise correlation of D_1, D_2 with respect to D is

$$\chi_D(D_1, D_2) = \left| \left\langle \frac{D_1}{D} - 1, \frac{D_2}{D} - 1 \right\rangle_D \right|.$$

Definition (average correlation of a set of distributions)

The average correlation of a set of distributions $\mathcal{D}'$ relative to D is

$$\rho(\mathcal{D}', D) = \frac{1}{|\mathcal{D}'|^2} \sum_{D_1, D_2 \in \mathcal{D}'} \chi_D(D_1, D_2).$$

Definition (statistical dimension with average correlation)

For $\bar{\gamma} > 0$, a domain X, a set of distributions $\mathcal{D}$ over X and a reference distribution D over X, the statistical dimension of $\mathcal{D}$ relative to D with average correlation $\bar{\gamma}$ is defined to be the largest value d such that for any subset $\mathcal{D}' \subseteq \mathcal{D}$ for which $|\mathcal{D}'| \geq \mathcal{D}/d$, we have $\rho(\mathcal{D}', D) \leq \bar{\gamma}$. This is denoted $\mathrm{SDA}_D(\mathcal{D}, \bar{\gamma})$.

For a search problem $\mathcal{Z}$ over distributions, we use: $\mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$

Later strengthened to use discrimination norm (Feldman, Perkins, Vempala '15) and then extended to "Randomized Statistical Dimension" (Feldman '17).

Intuitively, largest such d for which $1/d$ fraction of the set of distributions has low pairwise correlation is the statistical dimension.

Theorem (Feldman, Grigorescu, Reyzin, Vempala, Xiao '17)

Let X be a domain and $\mathcal{Z}$ be a search problem over a class of distributions D over X. For $\bar{\gamma} > 0$, let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$. To solve $\mathcal{Z}$ with probability $\geq 2/3$, any SQ algorithm requires at least:

1. d *calls to* $VSTAT(.\,, c_1/\bar{\gamma})$
2. $\min(d/4, c_2/\bar{\gamma})$ *calls to* $1\text{-}STAT(.)$
3. d *calls to* $STAT(.\,, c_3\sqrt{\bar{\gamma}})$.

Theorem (Feldman, Grigorescu, Reyzin, Vempala, Xiao '17)

Let X be a domain and $\mathcal{Z}$ be a search problem over a class of distributions D over X. For $\bar{\gamma} > 0$, let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$. To solve $\mathcal{Z}$ with probability $\geq 2/3$, any SQ algorithm requires at least:

1. *d calls to VSTAT$(. \, , c_1/\bar{\gamma})$*
2. $\min(d/4, c_2/\bar{\gamma})$ *calls to* 1-*STAT*$(.)$
3. *d calls to STAT$(. \, , c_3\sqrt{\bar{\gamma}})$.*

Szörényi's ('09) proof of the SQ-DIM lower bound for CSQs gives intuition. Recall, for query h and $f_1, \ldots, f_d$ realizing the SQ-DIM,

$$\left\langle h, \sum_{i \in A} f_i \right\rangle^2 \leq \left|\left|\sum_{i \in A} f_i\right|\right|^2 = \sum_{i,j \in A} \langle f_i, f_j \rangle \leq \sum_{i \in A} \left(1 + \frac{|A| - 1}{d}\right).$$

Theorem (Feldman, Grigorescu, Reyzin, Vempala, Xiao '17)

Let X be a domain and $\mathcal{Z}$ be a search problem over a class of distributions D over X. For $\bar{\gamma} > 0$, let $d = \mathrm{SDA}(\mathcal{Z}, \bar{\gamma})$. To solve $\mathcal{Z}$ with probability $\geq 2/3$, any SQ algorithm requires at least:

1. *d calls to VSTAT$(. \,, c_1/\bar{\gamma})$*
2. $\min(d/4, c_2/\bar{\gamma})$ *calls to 1-STAT(.)*
3. *d calls to STAT$(. \,, c_3\sqrt{\bar{\gamma}})$.*

differences from / extensions to SQ-DIM.

1. no need for labels.
2. $\bar{\gamma}$ instead of γ
3. disconnecting d from γ
4. the VSTAT oracle

Applications of SDA bounds

Consider the planted clique problem of detecting a k-clique randomly induced in a $G(n, \frac{1}{2})$ Erdös-Rényi random random graph instance.

- Information-theoretically, this is possible for $k > 2\log(n) + 1$.
- The state-of-the-art polynomial-time algorithm recovers cliques of size $k > \Omega(\sqrt{n})$ (Alon, Krivelevich, Sudakov '98).

Applications of SDA bounds

Consider the planted clique problem of detecting a k-clique randomly induced in a $G(n, \frac{1}{2})$ Erdös-Rényi random random graph instance.

- Information-theoretically, this is possible for $k > 2\log(n) + 1$.
- The state-of-the-art polynomial-time algorithm recovers cliques of size $k > \Omega(\sqrt{n})$ (Alon, Krivelevich, Sudakov '98).

SDA lower bounds show that statistical algorithms cannot efficiently recover cliques of size $O(n^{1/2-\epsilon})$.

Statistical variant of planted clique

To use SDA machinery, we first need to define a distributional version of planted clique.

Problem (distributional planted k-biclique)

For k, $1 \le k \le n$, and a subset of k indices $S \subseteq \{1, 2, \ldots, n\}$. The input distribution D_S on vectors $x \in \{0, 1\}^n$ is defined as follows: w.p. $1 - k/n$, x is uniform over $\{0, 1\}^n$; and w.p. k/n, x is such that its k coordinates from S are set to 1, and the remaining coordinates are uniform in $\{0, 1\}$. The problem is to find the unknown subset S.

an example:

coordinates of S:	(0,1,0,0,1,0,0,...,0,1)
w.p. k/n :	(U,1,U,U,1,U,U,...,U,1)
w.p. $(n-k)/n$:	(U,U,U,U,U,U,U,...,U,U)

Lower bounds for the planted clique problem

Theorem (Feldman, Grigorescu, Reyzin, Vempala, Xiao '17)

For $\epsilon \geq 1/\log n$ and $k \leq n^{1/2-\epsilon}$, let $\mathcal{D}$ be the set of all planted k-clique distributions. Then $SDA_U(\mathcal{D}, 2^{\ell+1}k^2/n^2) \geq n^{2\ell\delta}/3$

Corollary

For any constant $\epsilon > 0$ and any $k \leq n^{1/2-\epsilon}$, and $r > 0$, to solve distributional planted k-biclique with probability $\geq 2/3$, any statistical algorithm requires

- *at least $n^{\Omega(\log r)}$ queries to $VSTAT(. , n^2/(rk^2))$, or*
- *at least $\Omega(n^2/k^2)$ queries to 1-STAT(.).*

Evolvability

Evolutionary algorithms

Valiant ('09) defined the evolvability framework to model and formalize Darwinian evolution, with the goal of understanding what is "evolvable."

Definition (evolutionary algorithm)

An evolutionary algorithm A is defined by a pair (R, M) where

- R, the representation, is a class of functions from X to $\{-1, 1\}$.
- M, the mutation, is a randomized algorithm that, given $r \in R$ and an $\epsilon > 0$, outputs an $r' \in R$ with probability $\mathbf{Pr}_A(r, r')$.

$\mathtt{Neigh}_A(r, \epsilon) =$ set of r' that $M(r, \epsilon)$ may output (w.p. $1/p(n, 1/\epsilon)$).

performance of a representation

Definition (performance and empirical performance)

The performance of $r \in R$ w.r.t. an ideal function $f : X \to \{-1, 1\}$ is

$$\mathtt{Perf}_{f,D}(r) = \mathbf{E}_{x \sim D}[f(x)r(x)].$$

The empirical performance of r on s samples $x_1, \ldots, x_s$ from D is

$$\mathtt{Perf}_{f,D}(r, s) = \frac{1}{s} \sum_{i}^{t} f(x_i)r(x_i).$$

Natural? selection

Definition (selection)

Selection $\mathtt{Sel}[\tau, p, s](f, D, A, r)$ with parameters: tolerance τ, pool size p, and sample size s operating on $f, D, A = (R, M), r$ defined as before, outputs r^+ as follows.

1. Run $M(r, \epsilon)$ p times and let Z be the set of r's obtained.
2. For $r' \in Z$, let $\mathbf{Pr}_Z(r')$ be the frequency of r'.
3. For each $r' \in Z \cup \{r\}$ compute $v(r') = \mathtt{Perf}_{f,D}(r', s)$
4. Let $\mathtt{Bene}(Z) = \{r' \mid v(r') \geq v(r) + \tau\}$ and $\mathtt{Neut}(Z) = \{r' \mid |v(r') - v(r)| + \tau\}$
5. if $\mathtt{Bene} \neq \emptyset$, output r^+ proportional to $\mathbf{Pr}_Z(r^+)$ in $\mathtt{Bene}$
   else if $\mathtt{Neut} \neq \emptyset$, output r^+ proportional to $\mathbf{Pr}_Z(r^+)$ in $\mathtt{Neut}$
   else output $\perp$

Evolvability

Definition (evolvability by an algorithm)

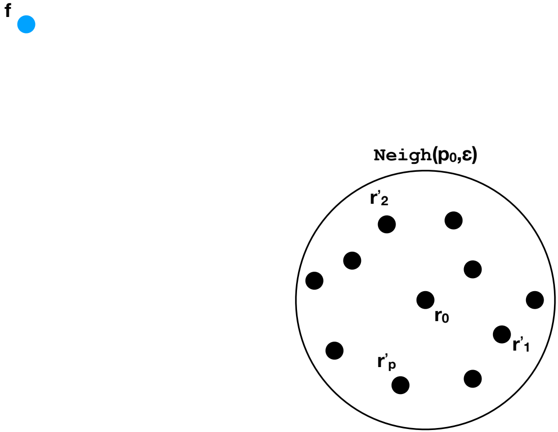
For concept class C over X, distribution D, and evolutionary algorithm A, we say that the class C is evolvable over D by A if there exist polynomials, $\tau(n, 1/\epsilon)$, $p(n, 1/\epsilon)$, $s(n, 1/\epsilon)$, and $g(n, 1/\epsilon)$ such that for every n, $c^* \in C$, $\epsilon > 0$, and every $r_0 \in R$, with probability at least $1 - \epsilon$, the random sequence $r_i \leftarrow \texttt{Sel}[\tau, p, s](c^*, D, A, r_{i-1})$ will yield a r_g s.t. $\texttt{Perf}_{c^*, D}(r_g) \geq 1 - \epsilon$.

Definition (evolvability of a concept class)

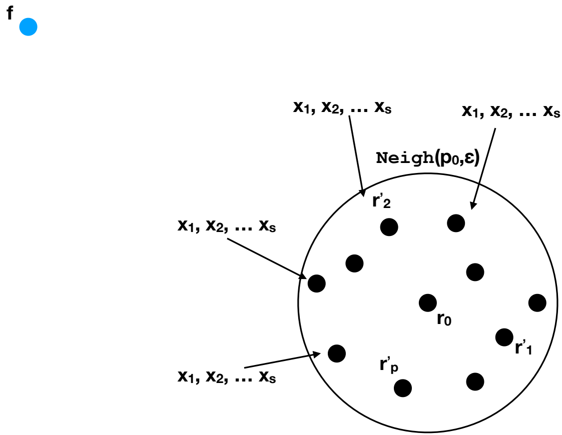
A concept class C is evolvable (over $\mathcal{D}$) if there exists an evolutionary algorithm A so that for any for any $D(\in \mathcal{D})$ over X, C is evolvable over D by A.

An illustration of evolvability

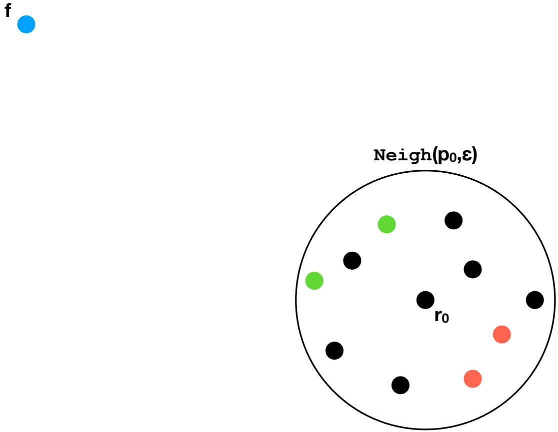
An illustration of evolvability



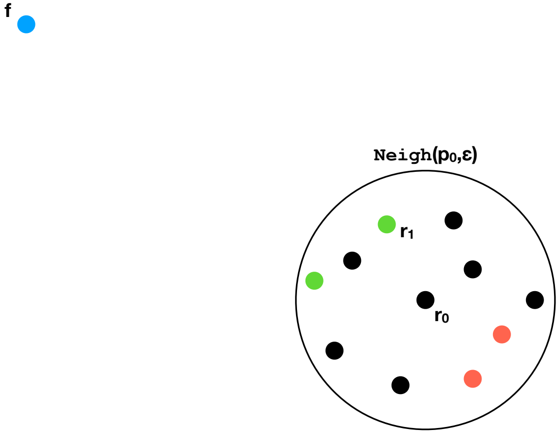
An illustration of evolvability



An illustration of evolvability

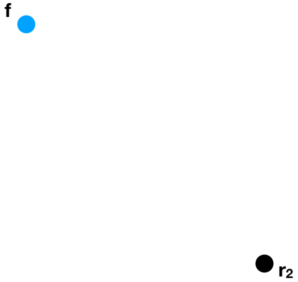


An illustration of evolvability

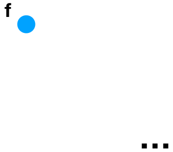


An illustration of evolvability

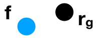
An illustration of evolvability



An illustration of evolvability



An illustration of evolvability



A characterization of evolvability

It turns out that evolvability is equivalent to learnability with CSQs!

Theorem (Feldman '08)

C is evolvable if and only if C is learnable with CSQs (over $\mathcal{D}$).

A characterization of evolvability

It turns out that evolvability is equivalent to learnability with CSQs!

Theorem (Feldman '08)

C is evolvable if and only if C is learnable with CSQs (over $\mathcal{D}$).

That EVOLVABLE $\subseteq$ CSQ is immediate (Valiant '09).
The other direction involves first showing that

$$\mathrm{CSQ}_{>}(r, \theta, \tau) = \begin{cases} 1 & \text{if } \mathbf{E}_D[r(x)c^*(x)] \geq \theta + \tau \\ 0 & \text{if } \mathbf{E}_D[r(x)c^*(x)] \leq \theta - \tau \\ 0 \text{ or } 1 & \text{otherwise} \end{cases}$$

can simulate CSQs. Then an evolutionary algorithm is made that simulates queries to a $\mathrm{CSQ}_{>}$ oracle.

What about sex?

Valiant's model of evolvability is asexual.

Kanade ('11) extended evolvability to include recombination by replacing `Neigh` (neighborhood) with `Desc` (descendants).

What about sex?

Valiant's model of evolvability is asexual.

Kanade ('11) extended evolvability to include recombination by replacing `Neigh` (neighborhood) with `Desc` (descendants).

Definition (recombinator)

For polynomial $p(,)$, a <u>p-bounded recombinator</u> is a randomized algorithm that takes as input two representations $r_1, r_2 \in R$ and ϵ and outputs a set of representations $\texttt{Desc}(r_1, r_2, \epsilon) \subseteq R$. Its running time is bounded by $p(n, 1/\epsilon)$. $\texttt{Desc}(r_1, r_2, \epsilon)$ is allowed to be empty which is interpreted as r_1 and r_2 being unable to mate.

Evolution under recombination

Definition (parallel CSQ)

A parallel CSQ learning algorithm uses p (polynomially bounded) processors and we assume that there is a common clock which defines parallel time steps. During each parallel time step a processor can make a CSQ query, perform polynomially-bounded computation, and write a message that can be read by every other processor. We assume that communication happens at the end of each parallel time step and on the clock. The CSQ oracle answers all queries in parallel.

Sexual evolution is equivalent to parallel CSQ learning.

Theorem (Kanade '11)

If C is parallel CSQ learnable in T query steps, then C is evolvable under recombination in $O(T\log^2(n/\epsilon))$ generations.

Differential privacy and adaptive data analysis

Differential privacy

The differential privacy of an algorithm captures an individual's "exposure" of being in a database when that algorithm is used (Dwork, McSherry, Nissim, Smith '06).

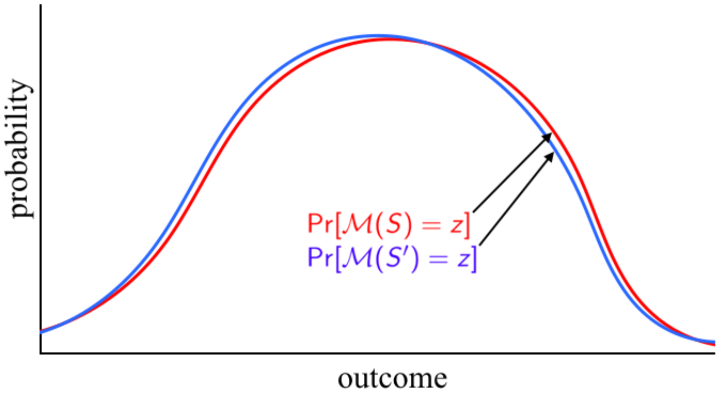
Definition (differential privacy)

A probabilistic mechanism $\mathcal{M}$ satisfies (α, β)-differential privacy if for any two samples S, S' that differ in just one example, for any outcome z

$$\mathbf{Pr}[\mathcal{M}(S) = z] \leq e^{\alpha}\mathbf{Pr}[\mathcal{M}(S') = z] + \beta.$$

If $\beta = 0$, we simply call $\mathcal{M}$ α-differentially private.

Differential privacy



Laplace mechanism

Definition (Laplace mechanism)

Given n inputs in $[0,1]$, the Laplace mechanism for outputting their average computes the true average value a and then outputs $a+x$ where x is drawn from the Laplace density with parameter $1/(\alpha n)$:

$$\mathrm{Lap}_{(0,\frac{1}{\alpha n})}(x) = \left(\frac{\alpha n}{2}\right) e^{-|x|\alpha n}.$$

Theorem (Dwork, McSherry, Nissim, Smith '06)

The Laplace mechanism satisfies α-differential privacy, and moreover has the property that with probability $\geq 1-\delta$, the error added to the true average is $O\left(\frac{\log(1/\delta')}{\alpha n}\right)$.

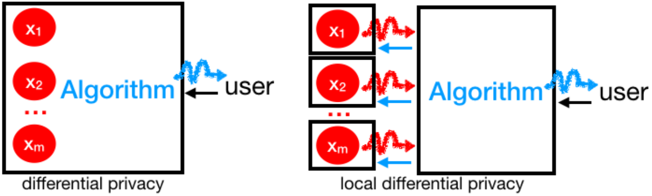
Differentially private learning

Theorem (Blum, Dwork, McSherry, Nissim '05)

If class C is efficiently SQ learnable, then it is also efficiently PAC learnable while satisfying α-differential privacy, with time and sample size polynomial in $1/\alpha$. In particular, if there is an algorithm that makes M queries of tolerance τ to learn C to error ϵ in the SQ model, then a sample of size $m = O\left(\left[\frac{M}{\alpha\tau} + \frac{M}{\tau^2}\right] \log\left(\frac{M}{\delta}\right)\right)$ is sufficient to PAC learn C to error ϵ with probability $1-\delta$ while satisfying α-differential privacy.

This is achieved by taking large enough sample and adding Laplace noise with scale parameter as to satisfy $\frac{\alpha}{M}$-differential privacy per query while staying within τ of the expectation of each query.

SQ equivalence to local differential privacy



Theorem (Kasiviswanaathan, Lee, Nissim, Raskhodnikova, Smith '11)

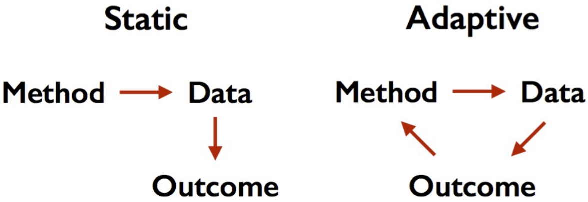
Concept class C is locally differentially privately learnable if and only if C is learnable using statistical queries.

Adaptive data analysis

Interestingly, differential privacy has applications to a new area of study called "adaptive data analysis."

Adaptive data analysis

Interestingly, differential privacy has applications to a new area of study called "adaptive data analysis."



– Illustration from blog post by Hardt ('15)

Adaptive data analysis was defined by Dwork, Feldman, Hardt, Pitassi, Reingold, and Roth (15).

Definition (adaptive accuracy)

A mechanism $\mathcal{M}$ is (α, β)-accurate on a distribution D and on queries $q_1, \ldots, q_k$, if for its responses $a_1, \ldots a_k$ we have

$$\mathbf{Pr}_{\mathcal{M}}[\max |q_i(D) - a_i| \leq \alpha] \geq 1 - \beta.$$

Note: there is also an analogous notion of (α, β) accuracy on a sample S.

A natural question is how many samples from D are needed to answer k queries adaptively with (α, β)-accuracy.

Adaptive data analysis was defined by Dwork, Feldman, Hardt, Pitassi, Reingold, and Roth (15).

Definition (adaptive accuracy)

A mechanism $\mathcal{M}$ is (α, β)-accurate on a distribution D and on queries $q_1, \ldots, q_k$, if for its responses $a_1, \ldots a_k$ we have

$$\mathbf{Pr}_{\mathcal{M}}[\max |q_i(D) - a_i| \leq \alpha] \geq 1 - \beta.$$

Note: there is also an analogous notion of (α, β) accuracy on a sample S.

A natural question is how many samples from D are needed to answer k queries adaptively with (α, β)-accuracy.

Note that there is no assumption about the complexity of the class from which the q_is come. So, standard techniques don't apply.

Differential privacy offers a notion of stability that "transfers" to adaptive accuracy. The following is an adapted *transfer theorem*.

Theorem (Dwork, Feldman, Hardt, Pitassi, Reingold, and Roth '15)

Let $\mathcal{M}$ be a mechanism that on sample $S \sim D^n$ answers k adaptively chosen statistical queries, is $(\frac{\alpha}{64}, \frac{\alpha\beta}{32})$-private for some $\alpha, \beta > 0$ and $(\frac{\alpha}{8}, \frac{\alpha\beta}{16})$-accurate on S. Then $\mathcal{M}$ is (α, β)-accurate on D.

Differential privacy offers a notion of stability that "transfers" to adaptive accuracy. The following is an adapted *transfer theorem*.

Theorem (Dwork, Feldman, Hardt, Pitassi, Reingold, and Roth '15)

Let $\mathcal{M}$ be a mechanism that on sample $S \sim D^n$ answers k adaptively chosen statistical queries, is $(\frac{\alpha}{64}, \frac{\alpha\beta}{32})$-private for some $\alpha, \beta > 0$ and $(\frac{\alpha}{8}, \frac{\alpha\beta}{16})$-accurate on S. Then $\mathcal{M}$ is (α, β)-accurate on D.

Putting together the Laplace mechanism with the transfer theorem, and doing some careful analysis to improve the bounds, one can get an adaptive algorithm for SQs.

Adaptively answering SQs

Theorem (Bassily, Nissim, Smith, Steinke, Stemmer, Ullman '16)

There is a polynomial-time mechanism that is (α, β)-accurate with respect to any distribution D for k adaptively chosen statistical queries given

$$m = \tilde{O}\left(\frac{\sqrt{k}\log^{3/2}(1/\beta)}{\alpha^2}\right)$$

samples from D.

Subsampling (Kasiviswanathan, Lee, Nissim, Raskhodnikova, Smith '08) can exponentially speed up the Laplace mechanism per-query without increasing the sample complexity (Fish, Reyzin, Rubinstein '18).

Other applications

A few other applications

Theorem (Sherstov '08)

Let C be the class of functions $\{-1,1\}^n \to \{-1,1\}$ computable in AC^0. If $\mathrm{SQ\text{-}DIM}(C) \leq O\left(2^{2^{(\log n)^\epsilon}}\right)$ for every constant $\epsilon > 0$, then $\mathrm{IP} \in \mathrm{PSPACE}^{\mathrm{cc}} \setminus \mathrm{PH}^{\mathrm{cc}}$.

Result (Chu, Kim, Lin, Yu, Bradski, Ng, Olukotun '06)

SQ algorithms can be put into "summation form" and automatically parallelized in MapReduce, giving nearly-linear speedups in practice.

Theorem (Steinhardt, Valiant, Wager '16)

Any class $\mathcal{C}$ that is learnable with m statistical queries of tolerance $1/m$, it is learnable from a stream of $\mathrm{poly}(m, \log|\mathcal{C}|)$ examples and $b = O(\log|\mathcal{C}| \log(m))$ bits of memory.

Summary

- SQs originate from a framework motivated, in part, for producing noise-tolerant algorithms.
- It turned out that most of our algorithms can work in the SQ framework.
- SQ dimension gives a serious impediment for learning and for optimization.
- Novel applications of SQs have allowed us to shed light on the difficulty of some problems.
- There are also perhaps unexpected applications, to differential privacy, adaptive data analysis, evolvability, among other areas.

Open problems

- Can we formally separate η-PAC from SQ?
  - Blum, Kalai, and Wasserman's ('00) result fails non constant η.
- Can we give evidence for the hardness of other classical problems using statistical dimension?
- Can we design/analyze faster or natural algorithms for evolvability.
  - e.g. the swapping algorithm (Valiant '09; Diochnos, Turán '09)
- What is the sample complexity of adaptively answering SQs?
  - best l.b.: $\Omega(\sqrt{k}/\alpha)$ (Hardt, Ullman '14) and u.b.: $O(\sqrt{k}/\alpha^2)$ (Bassiliy, Nissim, Smith, Steinke, Stemmer, Ullman '16)
- **Where else can SQ have an impact?**

Thank You!

Any questions?