CIS 800/002 The Algorithmic Foundations of Data Privacy November 1, 2011

Lecture 14

Lecturer: Aaron Roth *Scribe: Aaron Roth*

Query Release and Agnostic Learning

In this lecture, we will observe a simple connection between query release and agnostic learning, and interpret this connection as bad news for the problem of designing efficient non-interactive private query release algorithms – at least of the type that we have been considering so far.

Lets first introduce the problem of agnostic learning, and informally recall some results from learning theory.

Definition 1 *A concept class C is a collection of functions $f \in C$ of the form $f : X \to \{0,1\}$.*

Definition 2 *A labeled example is an element of $X \times \{0,1\}$. Write such an example as a pair $(x, \ell(x)), x \in X, \ell(x) \in \{0,1\}$.*

The learning problem is, given a distribution $\mathcal{D}$ over labeled examples, to find a function $f \in C$ that best labels the examples.

Definition 3 *The error of a hypothesis $f \in C$ over a distribution $\mathcal{D}$ over labelled examples is:*

$$err(f, \mathcal{D}) = \Pr_{x,\ell(x)\sim\mathcal{D}}[f(x) \neq \ell(x)]$$

The optimal error with respect to a class of queries C is:

$$OPT(C, \mathcal{D}) = \min_{f \in C} err(f, \mathcal{D})$$

An agnostic learning algorithm for a class of queries C is one which can find a function $f^* \in C$ of approximately minimum error, given any distribution over labeled examples.

Definition 4 *An α, γ-agnostic learning algorithm A for a class of queries C has the guarantee that given access to any distribution $\mathcal{D}$ over labeled examples, with probability at least $1 - \gamma$, $A(\mathcal{D}) = f \in C$ such that:*

$$err(f, \mathcal{D}) \leq OPT(C, \mathcal{D}) + \alpha$$

Non-agnostic learning corresponds to the case that $OPT(C, \mathcal{D}) = 0$.

We haven't yet specified how the algorithm gets access to the distribution. In practice, the algorithm will be given a dataset – i.e. a polynomially sized sample from the distribution. However, Kearns defines the *statistical query* model which characterizes a class of algorithms that in the end make only certain kinds of queries to the data set.

Definition 5 *A statistical query oracle $\mathcal{O}^\tau_\mathcal{D} : (X \times \{0,1\} \to [0,1]) \to \mathbf{R}$ with tolerance τ has the property that for every $f : X \times \{0,1\} \to [0,1]$:*

$$|\mathcal{O}^\tau_\mathcal{D}(f) - \mathrm{E}_{x,\ell(x)\sim\mathcal{D}}[f(x, \ell(x))]| \leq \tau$$

Note that access to a statistical query oracle with tolerance τ can be simulated with a sample from the distribution of size $\mathrm{poly}(1/\tau)$. The statistical query model will require algorithms to access the dataset only using a polynomial number of queries to a statistical query oracle with inverse polynomial tolerance.

Definition 6 *An algorithm is an (α, γ)-agnostic learning algorithm in the statistical query model if it is an (α, γ)-agnostic learning algorithm and accesses the data using only poly$(\log |X|)$ queries to $\mathcal{O}^\tau_\mathcal{D}$ for $\tau \geq 1/poly(\log |X|)$.*

Unfortunately, learning in the SQ model is limited even in the non-agnostic case. Let $X = \{0,1\}^d$, and define a parity function on a subset of variables $S \subseteq [d]$ to be:

$$P_S(x) = \frac{-1^{\sum_{i \in S} x_i}}{2} + \frac{1}{2}$$

i.e. the function that is 1 iff the sum of the variables indexed by S is even. Let $C_P = \{P_S : S \subseteq [d]\}$ be the set of all parity functions. Kearns proved that no SQ-algorithm can learn parity functions even in the non-agnostic setting.

Theorem 7 (Kearns 93) *Any α-learning algorithm for C_P even in the non-agnostic setting, for $\alpha \leq \frac{1}{2} - \left(\frac{1}{2^d}\right)^3$ must make at least $\frac{1}{2}2^{d/3}$ SQ queries with tolerance $\tau \leq \frac{1}{2^{d/3}}$.*

i.e. no algorithm which makes only polynomially many queries to inverse polynomial tolerance can learn parities to any non-trivial accuracy. In fact, the same is true for any super-polynomially sized subset of parity functions.

For non-agnostic learning, this is essentially the only limitation of the SQ model. Almost every other natural class of functions known to be PAC learnable is also learnable in the SQ model. However, for agnostic learning, the SQ model is much more restrictive, precluding even the relatively simple class of conjunctions. For each subset of variables $S \subseteq [d]$, define the conjunction (marginal) on subset S to be:

$$M_S(x) = \prod_{i \in S} x_i$$

i.e. the function that takes value 1 iff for each $i \in S$, $x_i = 1$. Write C_M for the set of all conjunctions.

Theorem 8 (Feldman 10) *Any α-learning algorithm for C_M in the agnostic setting for $\alpha \leq \frac{1}{2} - 1/poly(d)$ must make super-polynomially many SQ queries with tolerance $\tau \leq \frac{1}{poly(d)}$. The same is true for any super-polynomially sized subset of conjunctions.*

Note that these lower bounds are information theoretic, and not computational: they hold for algorithms that are allowed to perform arbitrary computations, and only bound the number of queries that the algorithms can make.

As we will see, an algorithm solving the non-interactive query release problem in the SQ model can also be used to solve the agnostic learning problem, and vice versa. i.e. the two problems are related in the sense that if there exists an algorithm making polynomially many queries to inverse polynomial tolerance that solves the one problem, then it can be used to solve the other problem as well in a black box manner. This means that if we hope to solve the query release problem for conjunctions with a polynomial time algorithm, it must involve some step that cannot be implemented only using polynomially many SQ queries – if there was such an algorithm it would violate the lower bound. This represents a barrier of sorts, since non-SQ techniques are very rare.

First we give a simple algorithm for the agnostic learning problem given an algorithm for the query release problem.

Observe that if M never makes more than m statistical queries, ReleaseToLearn never makes more than $2m$ since we run it twice...

Theorem 9 *If there exists an (α, β)-accurate query release mechanism for C making at most m statistical queries, then there exists a $(4\alpha, 2\beta)$-agnostic learning algorithm for C making at most $2m$ statistical queries.*

Proof It just remains to show that ReleaseToLearn is $(2\alpha, 2\beta)$-accurate for C. Condition on the event (which occurs except with probability 2β that $\max_{f \in C} |f(D_1) - f(\hat{D}_1)| \leq \alpha$ and $\max_{f \in C} |f(D_0) - f(\hat{D}_0)| \leq \alpha$ Let f^* be the function such that $\text{err}(f, D) = \text{OPT}(C, D)$ Observe that

$$f(D_1) - f(D_0) = n - \{x \in D_1 : f(x) \neq 1\} - \{x \in D_0 : f(x) = 0\} = \frac{1}{n}(1 - \text{OPT}(C, D))$$

Algorithm 1 Takes as input an (α, β)-accurate query release mechanism M for C that makes m SQ queries and a dataset $D \subseteq X \times 0, 1$. It is a $(2\alpha, 2\beta)$-agnostic learning algorithm for C making at most $2m$ SQ queries.

ReleaseToLearn(M, D)

Let $D_1 = \{x \in D : \ell(x) = 1\}$ and let $D_0 = \{x \in D : \ell(x) = 0\}$.
Let $\hat{D}_1 = M(D_1)$ and let $\hat{D}_0 = M(D_0)$. (Note that any statistical query f that M makes to $\mathcal{O}^\tau_{D_i}$ can be made to $\mathcal{O}^\tau_D$ by using the query $\hat{f}$ such that $\hat{f}(x) = f(x)$ if $\ell(x) = i$ and $\hat{f}(x) = 0$ otherwise.)
Output the function $f^* \in C$ that maximizes $f^*(\hat{D}_1) - f^*(\hat{D}_0)$.

and so the algorithm outputs some f such that $f(\hat{D}_1) - f(\hat{D}_0) \geq n(1 - \mathrm{OPT}(C, D) - 2\alpha)$. By a similar argument, any f output such that $f(\hat{D}_1) - f(\hat{D}_0) \geq n(1 - \mathrm{OPT}(C, D) - 2\alpha)$ must have $f(D_1) - f(D_0) \geq n(1 - \mathrm{OPT}(C, D) - 4\alpha)$, and therefore must have $\mathrm{err}(f, D) \leq \mathrm{OPT}(C, D) + 4\alpha$, proving the claim. ■

Now we have to show the reverse direction. But actually, we have already done this in previous lectures!

Recall our definition of a distinguisher:

Definition 10 *An (α, γ) distinguisher with respect to a class of queries $\mathcal{C}$ is an algorithm $A : \mathbf{N}^{|\mathcal{X}|} \times \mathbf{N}^{|\mathcal{X}|} \to \mathcal{C}$ with the following property. For any pair $D, D' \in \mathbf{N}^{|\mathcal{X}|}$ such that there exists a $Q^* \in \mathcal{C}$ such that $|Q^*(D) - Q^*(D')| \geq 3\alpha n$, $A(D, D') = Q$ such that $|Q(D) - Q(D')| \geq \alpha n$, except with probability at most γ.*

A distinguisher for C can be called a small number of times (when paired with a database update algorithm) to produce a dataset that is α-accurate for all queries in C. But we can implement a distinguisher with an agnostic learner! Suppose we have an $(2\alpha, \gamma)$- agnostic learner, and two databases $D, D' \in \mathbf{N}^{|\mathcal{X}|}$. We can define obtain a distinguisher as follows: Note that LearnToDistinguish makes

Algorithm 2 LearnToDistinguish takes as input a $(2\alpha, \gamma)$-agnostic learning algorithm for C that makes at most m statistical queries, and two databases D and D'. It is an $(\alpha, 2\gamma)$-distinguisher for $\mathcal{C}$ that makes at most $2m + 2$ statistical queries.

LearnToDistinguish(A, D, D')

Let

$$D^+ = \{(x, 1) : x \in D\} \cup \{(x, 0) : x \in D'\}$$

and

$$D^- = \{(x, 0) : x \in D\} \cup \{(x, 1) : x \in D'\}$$

Let $f^+ = A(D^+)$ and $f^- = A(D^-)$
Compute $a^+ = |f^+(D) - f^+(D')|$ and $a^- = |f^-(D) - f^-(D')|$ (Both are just statistical queries).
if $a^+ > a^-$ **then**
  Return f^+
else
  Return f^-
end if

just $O(m)$ statistical queries: it runs A twice, and them makes $O(1)$ more queries to compute a^+ and a^-.

Theorem 11 *If there exists a (α, γ)-agnostic learning algorithm for C making at most m statistical queries, then there exists an $(\alpha, 2\gamma)$-distinguisher for C making at most $O(m)$ statistical queries.*

Proof Condition on the event that neither run of the agnostic learning algorithm fails, which occurs with probability at least $1 - 2\gamma$. Suppose that there is some query $f^* \in C$ such that $|f^*(D) - f^*(D')| \geq 3\alpha n$. Recall that $n \cdot \text{OPT}(C, D^+) \leq |\{x \in D : f(x) = 0\}| + |\{x \in D' : f(x) = 1\}| = n + f(D') - f(D)$. Similarly, $n \cdot \text{OPT}(C, D^-) = n + f(D) - f(D')$. Hence, it must be that $\frac{1}{n}\max(a^+, a^-) \geq \frac{1}{2} + 2\alpha$ – i.e. that the returned query f^* is such that $|f^*(D) - f^*(D')| \geq 2\alpha$. But this is what we wanted from a distinguisher! ■

Finally, we need only recall that we can pair any $(3\alpha, 2\gamma)$-distinguisher for C with a $B(\alpha)$ database update algorithm to obtain an $(\alpha, 2B(\alpha)\gamma)$-accurate release algorithm for C that makes only $B(\alpha)$ black box calls to the distinguisher, and accesses the database in no other way. Therefore, if the distinguisher never makes more than m statistical queries to the database, the query release mechanism never makes more than $B(\alpha) \cdot m$ statistical queries to the database. Recall also that we have database update algorithms (e.g. the multiplicative weights algorithm) for which $B(\alpha)$ is $\text{poly}(|X|, 1/\alpha)$. For the MW algorithm, we have $B(\alpha) = \frac{4\log|X|}{\alpha^2}$. Therefore, putting this all together, we have the following theorem:

Theorem 12 *If there exists an (α, γ)-agnostic learning algorithm for C making at most m statistical queries, then there exists an $(\alpha, 8\log|X|\gamma/\alpha^2)$-accurate query release algorithm for C making at most $(8m + 8)\log|X|/\alpha^2$ statistical queries.*

So what have we shown? A-priori, agnostic learning seems like an easier task than query release. It requires only revealing the value of a single function $f \in C$ on the dataset, whereas the query release problem involves revealing the answers to every function in the class. Nevertheless, we have shown that the two problems are information-theoretically equivalent up to polynomial factors. That is, given an algorithm for the one problem, we can run it in a black box way only a polynomial number of times, and obtain an algorithm for the other problem. This means, in particular, that there is no algorithm for releasing conjunctions to non-trivial error that makes only polynomially many SQ queries.

Who cares? We would like a polynomial time algorithm for non-interactively releasing the very natural class of conjunctions. This result does not preclude such an algorithm, but does imply that any such algorithm must be inherently impossible to implement using only SQ queries. But our basic algorithmic technique of adding laplace noise to linear queries is inherently SQ! Even many ways you might think of to sample from the exponential mechanism (such as sampling from a random walk using the metropolis method) can be implemented with SQ queries, and so this result implies in particular that such random walks do not mix quickly. In fact, we do not currently seem to have (almost) any methods that are simultaneously efficient, and which cannot be implemented using statistical queries. This means that to make major progress on the problem of non-interactive query release, we need fundamentally new techniques.

This reduction also possibly sheds some light on just *why* agnostic learning is so hard: although it seems not to reveal much more information about the distribution than non-agnostic learning, in fact, an agnostic learning algorithm has the power to recover the approximate values for *every* function in C evaluated on the distribution. This observation can be used to recover Feldman's hardness result for agnostically learning conjunctions in the SQ model. An agnostic SQ learning algorithm for conjunctions could actually be used to recover the approximate values of all conjunctions evaluated on the data set. But the set of all 2^d conjunctions forms a basis for the set of all boolean functions: that is, every boolean function can be written as a linear combination of conjunctions. So the answers to all 2^d conjunctions actually reveal the answers to all 2^d parity functions as well (up to somewhat higher error). And the answers to all 2^d parity queries easily allow you to learn parities (Just pick the one with the largest value). So an agnostic learning algorithm for conjunctions could be used as a (computationally inefficient, but SQ query efficient) learning algorithm for parities, which Kearns proved does not exist...

Bibliographic Information The statistical query model was introduced by Michael Kearns in "Efficient noise-tolerant learning from statistical queries", 1993, which also proved that there is no SQ learning algorithm for parities. That conjunctions are hard to agnostically learn in the SQ model was

proven by Feldman in "A complete characterization of statistical query learning with applications to evolvability", 2009. The first connection between data privacy and the SQ model of learning was drawn by Kasiviswanathan, Lee, Nissim, Raskhodnikova, and Smith, in "What can we learn privately", 2008. The connection between agnostic learning and query release was shown by Gupta, Hardt, Roth, and Ullman, in "Privately Releasing Conjunctions and the Statistical Query Barrier", 2011.