Strengths and weakness for learning functions from quantum examples

Srinivasan Arunachalam (IBM Quantum)

A quick recap

PAC learning

We let $\mathcal{C} \subseteq \{c : \{0,1\}^n \to \{0,1\}\}$ be a concept class.

We let $D : \{0,1\}^n \to [0,1]$ be an unknown distribution.

- Classical PAC learning: obtained $(x, c(x))$ where $x \sim D$.
- Quantum PAC learning: obtained copies of $\sum_x \sqrt{D(x)}\, |x, c(x)\rangle$

Goal: Output h such that $\Pr_{x\sim D}[h(x) = c(x)] \geq 1 - \varepsilon$

Quantum sample complexity equals classical sample complexity of PAC learning

But. The distribution which witnessed the quantum lower bound was "unnatural".

1. What if *D* is nicer? Say the uniform distribution?
2. Do uniform quantum examples provide a speedup?
3. What happens if we query c and not just obtain examples?

Quantum examples help the coupon collector

Standard coupon collector

Problem: Suppose there are N coupons. How many coupons to draw (with replacement) before having seen each coupon at least once?

Answer: Simple probability analysis shows $\Theta(N \log N)$

Variation to coupon collector

Problem: Suppose there are N coupons. Fix unknown $i^* \in \{1, \ldots, N\}$. How many coupons to draw (with replacement) from $\{1, \ldots, N\} \setminus \{i^*\}$ before learning i^*?

Answer: Same analysis as earlier shows $\Theta(N \log N)$

What if we are given "quantum examples"

Suppose a quantum learner obtains quantum examples $\frac{1}{\sqrt{N-1}} \sum_{i \in (\{1,\ldots,N\} \setminus \{i^*\})} |i\rangle$. How many quantum examples before learning i^*?

Answer: Can learn i^* using $\Theta(N)$ quantum examples

Proof idea: Analyze the success probability using the pretty good measurement. Write down the Gram matrix observe that it's easily diagonalizable.

If $T = O(N)$, then $P_{opt} \geq P_{pgm} \geq 2/3$

Fourier sampling: a useful trick under uniform D

- Let $c : \{0,1\}^n \to \{-1,1\}$. Then the Fourier coefficients are

$$\widehat{c}(S) = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} c(x)(-1)^{S \cdot x} \quad \text{for all } S \in \{0,1\}^n$$

- Parseval's identity: $\sum_S \widehat{c}(S)^2 = \mathbb{E}_x[c(x)^2] = 1$
  So $\{\widehat{c}(S)^2\}_S$ forms a probability distribution
- Given quantum example under uniform D:

$$\frac{1}{\sqrt{2^n}} \sum_x |x, c(x)\rangle \xrightarrow{\text{Hadamard}} \sum_S \widehat{c}(S) \, |S\rangle$$

- Measuring allows to sample from the Fourier distribution $\{\widehat{c}(S)^2\}_S$
- Classically: sampling from $\{\widehat{c}(S)^2\}_S$ is hard given $(x, c(x))$ examples

Applications of Fourier sampling

- Consider the concept class of parities $\mathcal{C}_1 = \{c_S(x) = S \cdot x\}_{S \in \{0,1\}^n}$

  Classical: $\Omega(n)$ classical examples needed

  Quantum: 1 quantum example suffices to learn $\mathcal{C}_1$ (Bernstein-Vazirani'93)

- Consider $\mathcal{C}_2 = \{c \text{ is a } \ell\text{-junta}\}$, i.e., $c(x)$ depends only on ℓ bits of x

  Classical: Efficient learning is notoriously hard for $\ell = O(\log n)$ and uniform D

  Quantum: $\mathcal{C}_2$ can be *exactly* learnt using $\widetilde{O}(2^\ell)$ quantum examples and in time $\widetilde{O}(n2^\ell + 2^{2\ell})$ (Atıcı-Servedio'09)

Generalizing both these concept classes?

Definition: We say c is k-Fourier sparse if $|\{S : \widehat{c}(S) \neq 0\}| \leq k$.

Note that $\mathcal{C}_1$ is 1-Fourier sparse and $\mathcal{C}_2$ is 2^ℓ-Fourier sparse

Consider the concept class $\mathcal{C} = \{c : \{0,1\}^n \rightarrow \{-1,1\} : c \text{ is } k\text{-Fourier sparse}\}$

Observe that $\mathcal{C}_1 \subseteq \mathcal{C}$. $\mathcal{C}$ contains linear functions
Observe that $\mathcal{C}_2 \subseteq \mathcal{C}$. $\mathcal{C}$ contains $(\log k)$-juntas

Learning $\mathcal{C} = \{c \text{ is } k\text{-Fourier sparse}\}$

- Exact learning $\mathcal{C}$ under the uniform distribution D
- Classically (Haviv-Regev'15): $\widetilde{\Theta}(nk)$ classical examples $(x, c(x))$ are necessary and sufficient to learn the concept class $\mathcal{C}$
- Quantumly (ACLW'18): $\widetilde{O}(k^{1.5})$ quantum examples $\frac{1}{\sqrt{2^n}} \sum_x |x, c(x)\rangle$ are sufficient to learn $\mathcal{C}$ (independent of the universe size n)

Sketch of upper bound 1

- Structural property: if c is k-Fourier sparse, then $\widehat{c}(S)^2 \geq 1/k^2$
- Use Fourier sampling to sample $S \sim \{\widehat{c}(S)^2\}_S$
- Collect all the S using $O(k^2)$ samples.
- Estimate each $\widehat{c}(S)$ using classical examples. Sample, time complexity is $O(k^2)$

A more sophisticated analysis.

- Fourier sample and collect Ss until the learner learns $\mathcal{V} = \text{span}\{S : \widehat{c}(S) \neq 0\}$
- Suppose $\dim(\mathcal{V}) = r$, then $\widetilde{O}(rk)$ quantum examples suffice to find $\mathcal{V}$
- Use the result of [HR'15] to learn c' completely using $\widetilde{O}(rk)$ classical examples
- Since $r \leq \widetilde{O}(\sqrt{k})$, we get $\widetilde{O}(k^{1.5})$ upper bound

Learning Disjunctive normal Forms (DNF)

DNFs

Simply an OR of AND of variables. For example, $(x_1 \wedge x_4 \wedge \overline{x_3}) \vee (\overline{x_4} \wedge x_6 \wedge x_7 \wedge \overline{x_8})$

We say a DNF on n variables is an s-term DNF if number of clauses is $\leq s$

Learning $\mathcal{C} = \{c \text{ is an } s\text{-term DNF in } n \text{ variables}\}$ under uniform D

- Classically: Efficient learning using examples is a longstanding open question. Best known upper bound is $n^{O(\log n)}$ [Verbeurgt'90]
- Quantumly: Bshouty-Jackson'95 gave a polynomial-time quantum algorithm!

Proof sketch of quantum upper bound

- Structural property: if c is an s-term DNF, then there exists U s.t. $|\widehat{c}(U)| \geq \frac{1}{s}$
- Fourier sampling! Sample $T \sim \{\widehat{c}(T)^2\}_T$, poly$(s)$ many times to see such a U
- Construct a "weak learner" who outputs h s.t. $\Pr[h(x) = c(x)] = \frac{1}{2} + \frac{1}{s}$
- Not good enough! Want a h that agrees with c on most inputs x's
- Boosting: Run weak learner many times in some manner to obtain a strong learner who outputs h satisfying $\Pr[h(x) = c(x)] \geq 2/3$

Membership oracle model

Let $\mathcal{C} \subseteq \{c : \{0,1\}^n \to \{0,1\}\}$ be a concept class and $c^* \in \mathcal{C}$ be an unknown.

Classical model

Membership queries. Suppose we can query c^* as follows:

on input $x \in \{0,1\}^n$, the learning algorithm obtains $c^*(x)$.

Goal: Learn c^* or output $h : \{0,1\}^n \to \{0,1\}$ such that $\Pr_x[c^*(x) = h(x)] \geq 2/3$

Complexity measure: Number of classical queries to c, call it $D(c)$.
Let $D(\mathcal{C}) = \max_{c \in \mathcal{C}} D(c)$ be query complexity of learning $\mathcal{C}$.

Quantum model

Quantum membership queries. Suppose we can quantumly query c^* in as follows:

$$O_{c^*} : |x, 0\rangle \to |x, c^*(x)\rangle .$$

In particular, these allow to obtain

$$\frac{1}{\sqrt{2^n}} \sum_x |x, 0\rangle \to \frac{1}{\sqrt{2^n}} \sum_x |x, c^*(x)\rangle$$

Goal: Learn c^* or output $h : \{0,1\}^n \to \{0,1\}$ such that $\Pr_x[c^*(x) = h(x)] \geq 2/3$

Complexity measure: Number of quantum queries to c, call it $Q(c)$.
Let $Q(\mathcal{C}) = \max_{c \in \mathcal{C}} Q(c)$ be query complexity of learning $\mathcal{C}$.

Membership oracle model

Classical model

Membership queries. Suppose we can query c^* as follows:

on input $x \in \{0,1\}^n$, the learning algorithm obtains $c^*(x)$.

Goal: Learn c^* or output $h : \{0,1\}^n \to \{0,1\}$ such that $\Pr_x[c^*(x) = h(x)] \geq 2/3$

Complexity measure: $D(\mathcal{C}) = \max_{c \in \mathcal{C}} D(c)$ be query complexity of learning $\mathcal{C}$.

Quantum model

Quantum membership queries. Suppose we can quantumly query c^* in as follows:

$$O_{c^*} : |x, 0\rangle \to |x, c^*(x)\rangle \, .$$

Goal: Learn c^* or output $h : \{0,1\}^n \to \{0,1\}$ such that $\Pr_x[c^*(x) = h(x)] \geq 2/3$

Complexity measure: $Q(\mathcal{C}) = \max_{c \in \mathcal{C}} Q(c)$ be query complexity of learning $\mathcal{C}$.

Question: Could $Q(\mathcal{C})$ be exponentially smaller than $D(\mathcal{C})$?

No. One can show that $Q(\mathcal{C}) \leq D(\mathcal{C}) \leq nQ(\mathcal{C})^3$ for every $\mathcal{C}$.

In the membership query model, quantum queries can give at most a polynomial speedup for learning over classical queries.

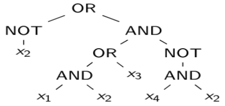
Shallow circuits

Gates: $\text{AND}(x) = 1$ iff $x = 1^n$, $\text{OR}(x) = 0$ iff $x = 0^n$, $\text{MAJ}(x) = 1$ iff $\sum_i x_i > n/2$

We say $c : \{0,1\}^n \to \{0,1\}$ is computed by a shallow circuit if:

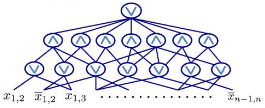
c can be computed by a constant-depth polynomial-sized circuit with:

bounded fan-in AND, OR, NOT gates (NC^0)



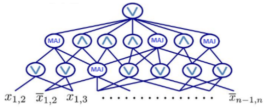
NC^0 Circuit

unbounded fan-in AND, OR, NOT gates (AC^0)



AC^0 Circuit

unbounded fan-in AND, OR, NOT, MAJ gates (TC^0)



TC^0 Circuit

Why consider NC^0 and AC^0 circuits?

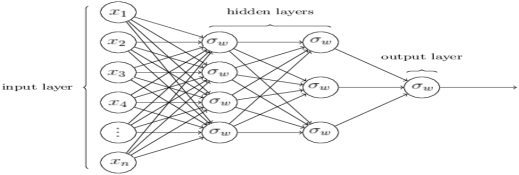
Shallow circuits have proven useful in exhibiting quantum advantage:

- BGK'18: A relational problem which can be solved using shallow quantum circuits but requires logarithmic-depth NC^0 circuit
- BKST'19: Improved the BGK'18 separation from NC^0 to AC^0
- CSV'18: Used BGK'18 for exponential certified randomness expansion
- G'18: Improved BGK'18 to give a separation in the average-case setting
- GNR'19: Used the construction of BGK'18 to show separations in the LOCAL model

Do shallow circuits give a quantum advantage for a learning task?

Why consider TC^0 circuits?

A theoretical way to model neural networks: A simple feed-forward neural network



where σ_w is the sigmoid function associated with weights $w = (w_0, w_1, \ldots, w_n)$. The weights could be exponential in n

A sequence of results in the 90s showed that constant-depth polynomial-sized feed-forward neural networks can be implemented by TC^0 circuits

Do quantum resources help learning a class of neural networks faster?

Learning NC^0 efficiently: a simple observation

The circuit class NC^0

Recall: Class of functions $c : \{0,1\}^n \to \{0,1\}$ such that c can be computed by an $O(1)$-depth circuit with $\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}$ gates on at most 2 bits

Observation: If c is computed by a depth-d NC^0 circuit (denoted NC^0_d), then $c(x)$ depends on at most 2^d input bits of x

Learning juntas quantumly

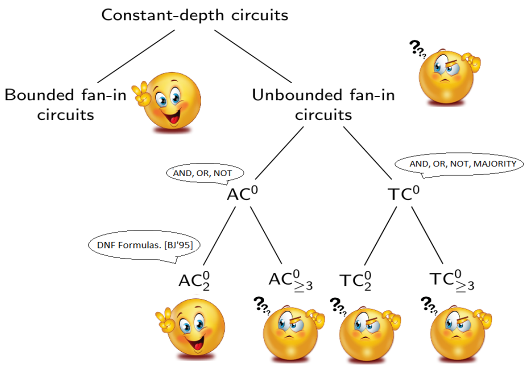
Consider $\mathcal{C} = \{c \text{ is a } \ell\text{-junta}\}$, i.e., $c(x)$ depends only on ℓ bits of x

Classical: Efficient learning is notoriously hard for $\ell = O(\log n)$ and uniform D

Quantum: $\mathcal{C}$ can be *exactly* learned using $\widetilde{O}(2^\ell)$ quantum examples and in time $\widetilde{O}(n2^\ell + 2^{2\ell})$ (Atıcı-Servedio'09)

Observation. NC^0_d consists of 2^d-juntas. Can be learned in time $O(n2^{2^d} + 2^{2^{d+1}})$.
If $d = O(1)$, then NC^0 can be learned in quantum polynomial-time using only uniform quantum examples. Also bounded fan-in circuits can be learned quantum-efficiently

Motivation question for this talk: Learn constant-depth circuits?



Clasically what is known?

Learning AC^0 under the uniform distribution

- **Upper bounds**: Linial, Mansour, Nisan'89 showed how to learn AC^0 circuits in quasi-polynomial time i.e., $n^{O(\log n)}$ time
- Crucial idea: Learn the Fourier spectrum of AC^0 circuits
- **Lower bound**: Kharitonov'93 (conditionally) showed that the quasi-polynomial time bound of LMN'89 is optimal
- AC^0 hardness assumed that factoring is hard for sub-exponential time algorithms

Learning TC^0

- Not much is known about learning even depth-2 TC^0 circuits under the uniform distribution
- Kharitonov'93 ruled out polynomial-time learners for TC^0 assuming factoring is polynomial-time hard
- Klivans-Sherstov'09 showed PAC learning depth-2 TC^0 circuits is hard based on hardness of breaking LWE-cryptosystem

Quantum hardness of learning

"*Can* AC^0 *and* TC^0 *be quantum PAC learned?*"

Strong negative answer: under the uniform distribution setting given queries

(1.) If we can learn $\mathsf{AC}^0, \mathsf{TC}^0$, then we can break Learning with Errors cryptosytem (which is the basis of post-quantum cryptographic systems):

(2.) If we can learn TC^0_2, then we would obtain a breakthrough in complexity theory.

Tools of interest

Pseudo-random functions (PRF)

A family $\mathcal{F} = \{F_s : \{0,1\}^n \to \{0,1\}^\ell : s \in \{0,1\}^k\}$ where s is a key

We use the notation $\mathcal{A}^F$ meaning $\mathcal{A}$ can make queries to F at unit cost.

Important property: A PRF is said to be secure if:

There exists no polynomial-time algorithm $\mathcal{A}$ such that

$$\left| \Pr_{s \in \{0,1\}^k}[\mathcal{A}^{F_s}(\cdot) = 1] - \Pr_U[\mathcal{A}^U(\cdot) = 1] \right| \geq \frac{1}{\text{poly}(n)},$$

where U is a uniformly random oracle $U : \{0,1\}^n \to \{0,1\}^{\ell(n)}$

$\mathcal{A}$ cannot distinguish between truly random oracle U and "fake" random oracle $F_s \in \mathcal{F}$

We say the PRF $\mathcal{F}$ is quantum-secure if $\mathcal{A}$ was a quantum polynomial-time algorithm

Learning with Errors (LWE)

One of the leading candidates for post-quantum cryptographic schemes:
Learning with Errors [Regev'05]

Important property: Best known quantum algorithm run in exponential time. A sub-exponential time quantum algorithm would already be a breakthrough

Our hardness is based on hardness for poly-time/ subexp-time algorithms for LWE

Pseudo-random functions and learning

Let $\mathcal{F} = \{F_s : \{0,1\}^n \to \{0,1\}^\ell\}_s$ be a quantum-secure PRF, i.e., **no** efficient quantum distinguisher $\mathcal{A}$ such that

$$\left| \Pr_{s \in \{0,1\}^k}[\mathcal{A}^{F_s}(\cdot) = 1] - \Pr_U[\mathcal{A}^U(\cdot) = 1] \right| \geq \frac{1}{\text{poly}(n)}$$

In particular, no efficient quantum algorithm can distinguish if it was given oracle access to $F \in \mathcal{F}$ or uniformly random U

Let $\mathcal{C}_\mathcal{F} = \{F'_s : \{0,1\}^n \to \{0,1\} : F'_s(x) = \text{FBIT}(F_s(x))\}_{F_s \in \mathcal{F}}$ be a concept class.

Assume $\mathcal{B}$ is an efficient quantum learner for $\mathcal{C}_\mathcal{F}$. Consider an algorithm $\mathcal{A}$:

$\mathcal{A}$ is given oracle O s.t.: $O \in \mathcal{C}_\mathcal{F}$ or O is uniformly random oracle $U : \{0,1\}^n \to \{0,1\}$

- $\mathcal{A}$ prepares copies of $\frac{1}{\sqrt{2^n}} \sum_x |x\rangle |O(x)\rangle$ efficiently and passes it to $\mathcal{B}$. Similarly queries made by $\mathcal{B}$ can be simulated by $\mathcal{A}$
- $\mathcal{B}$ outputs a hypothesis $h : \{0,1\}^n \to \{0,1\}$.
  $\mathcal{A}$ says $O \in C_\mathcal{F}$ iff $h(x) = O(x)$ for uniformly random $x \in \{0,1\}^n$

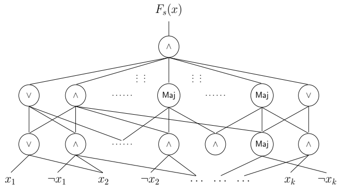
Technical lemma: If the learning algorithm $\mathcal{B}$ has bias β, then the bias of $\mathcal{A}$ is $\geq \beta/2$

Contradiction. Let $\beta = \frac{1}{\text{poly}(n)} \implies \mathcal{A}$ serves as a quantum distinguisher since $\mathcal{B}$ was efficient. Contradicts that G was quantum-secure, hence $\mathcal{B}$ couldn't have been efficient

Hardness of TC^0, AC^0

Starting point: PRF $\mathcal{F}$ constructed by [BPR'12] which is secure assuming the LWE problem is hard

Show that for every s, the function F_s can be computed by TC^0 circuit



In particular, every concept $c \in \mathcal{C}_{\mathcal{F}}$ can be computed by a TC^0 circuit

Main theorem 1. If there exists an efficient quantum learner for $\mathcal{C}_{\mathcal{F}} \subseteq TC^0$, then there exists a polynomial-time quantum algorithm for the LWE problem

Similar idea doesn't work for AC^0: PRFs constructed from LWE cannot naturally be computed in TC^0. Overcome by reducing key-size and relaxing the security of LWE

Main theorem 2. If there exists a quasi-polynomial quantum learner for $\mathcal{C}'_{\mathcal{F}} \subseteq AC^0$, then there exists a sub-exponential time quantum algorithm for the LWE problem

Hardness of learning depth-2 TC0 circuits

Drawback. 1. Using the PRF approach we had above, we are not able to say anything about lower bounds for circuit families with *very small* depth.

2. None of these PRFs are known to be implementable in depth ≤ 6

Main result. If a class $\mathcal{C}$ of polynomial-size concepts can be learned under the uniform distribution with membership queries and with error $\varepsilon \leq 1/2 - \gamma$ and in quantum time $o(\gamma^2 \cdot 2^n)$, then BQE $\notin \mathcal{C}$ (i.e., bounded quantum exponential time $\notin \mathcal{C}$).

Two trivial algorithms

1. Query everything: Query/time complexity is 2^n, error $\varepsilon = 0$
2. Fourier sample: Time complexity is poly(n), error is $1/2 - \Omega(2^{-n/2})$

Concrete application

Consider $\mathcal{C} = \mathsf{TC}_2^0$ (the class of depth-2 threshold circuits). If there exists a non-trivial learning algorithm for TC_2^0, then new circuit lower bounds. In particular BQE $\notin \mathsf{TC}_2^0$.

Conceptually

(i) Explains why devising new quantum learning algorithms is hard

(ii) Gives a new motivation for providing new quantum speedups

Learning parities agnostically?

Let $\mathcal{C} = \{c_S : \{0,1\}^n \to \{0,1\} : c_S(x) = \langle S, x\rangle\}_S$.

1. In uniform PAC learning we are given

$$|\psi_S\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x, c_S(x)\rangle\,,$$

learn S. Using $O(1)$ copy of $|\psi_S\rangle$.

2. In random-classification noise PAC learning we are given

$$|\psi_S\rangle = \frac{1}{\sqrt{2^n}} \sum_x \sqrt{1-\eta}\,|x, c_S(x)\rangle + \sqrt{\eta}\,|x, 1 \oplus c_S(x)\rangle\,,$$

learn S. Using poly$(1/(1-2\eta)^2)$ copies $|\psi_S\rangle$.

3. The "hardest" **agnostic model.** A quantum learning algorithm obtains

$$|\psi_g\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes \Big(\sqrt{\frac{1+g(x)}{2}}\,|1\rangle + \sqrt{\frac{1-g(x)}{2}}\,|0\rangle\Big)$$

for an *arbitrary* $g : \{0,1\}^n \to [-1,1]$. Find S such that

$$|\widehat{g}(S)| \in [\max_T |\widehat{g}(T)| - \varepsilon, \max_T |\widehat{g}(T)| + \varepsilon].$$

Can we learn parities in the agnostic model?

Learning parities agnostically?

Let $\mathcal{C} = \{c_S : \{0,1\}^n \to \{0,1\} : c_S(x) = \langle S, x \rangle\}_S$. In the agnostic model. a quantum learning algorithm obtains

$$|\psi_g\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes \left(\sqrt{\frac{1+g(x)}{2}}\,|1\rangle + \sqrt{\frac{1-g(x)}{2}}\,|0\rangle \right)$$

for an *arbitrary* $g : \{0,1\}^n \to [-1,1]$. Find S such that

$$|\widehat{g}(S)| \in [\max_T |\widehat{g}(T)| - \varepsilon, \max_T |\widehat{g}(T)| + \varepsilon].$$

A classical algorithm

1. Measure $|\psi_g\rangle$ to obtain (x, b) where $b = 1$ with probability $(1 + g(x))/2$ and $b = 0$ with probability $(1 - g(x))/2$.
2. [FGKP'06] showed classical agnostic learning reduces to random classification noise model *for parities*
3. LPN is solvable using $O(n)$ samples and time $2^{n/\log n}$.

Can we learn parities agnostically quantum time efficiently?

Why care? Give a quantum polynomial time for AC^0_3 under the uniform distribution (the classical analogue is an open question)

Learning parities agnostically?

Let $\mathcal{C} = \{c_S : \{0,1\}^n \to \{0,1\} : c_S(x) = \langle S, x\rangle\}_S$. In the agnostic model for an *arbitrary* $g : \{0,1\}^n \to [-1,1]$, a quantum learning algorithm obtains

$$|\psi_g\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes \left(\sqrt{\frac{1+g(x)}{2}}\,|1\rangle + \sqrt{\frac{1-g(x)}{2}}\,|0\rangle \right)$$

Find S in $|\widehat{g}(S)| \in [\max_T |\widehat{g}(T)| - \varepsilon, \max_T |\widehat{g}(T)| + \varepsilon]$.

Harder task: given $|\psi_g\rangle^{\otimes t}$ sample from a distribution $D_g : \{0,1\}^n \to [0,1]$ satisfying

$$\sum_S \left| D_g(S) - \frac{\widehat{g}(S)^2}{\sum_S \widehat{g}(S)^2} \right| \leq \varepsilon.$$

Unclear if possible even sample efficiently! One approach for showing hardness.

Consider the hard instance

$$\mathcal{E}_1 = \{g : \{0,1\}^n \to \{1, 2/\sqrt{N} - 1\} : |g^{-1}(1)| = N/2 - \sqrt{N}\},$$
$$\mathcal{E}_2 = \{g : \{0,1\}^n \to \{1, 2/\sqrt{N} - 1\} : |g^{-1}(1)| = N/2 + \sqrt{N}\}.$$

Learning parities agnostically?

Given copies of $|\psi_g\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes \left(\sqrt{\frac{1+g(x)}{2}} |1\rangle + \sqrt{\frac{1-g(x)}{2}} |0\rangle \right)$ can we sample from a distribution $D_g : \{0,1\}^n \to [0,1]$ satisfying

$$\sum_S \left| D_g(S) - \frac{\widehat{g}(S)^2}{\sum_S \widehat{g}(S)^2} \right| \leq \varepsilon.$$

Hard instance. Consider a set of $g : \{0,1\}^n \to \{1, \frac{2}{\sqrt{N}} - 1\}$.
$\mathcal{E}_1$ is set of gs s.t. $|g^{-1}(1)| = N/2 - \sqrt{N}$
$\mathcal{E}_2$ is set of gs s.t. $|g^{-1}(1)| = N/2 + \sqrt{N}$.

Using a technique of Aaronson-Ambainis reduces to the following task. Let

$$\mathcal{C} = \Big\{ |\psi_z\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle \otimes \left(\sqrt{z_x}\, |0\rangle + \sqrt{1 - z_x}\, |1\rangle \right) : \\ z \in \{1, 1/\sqrt{N}\}^N, |z| = \sum_i z_i \in \{N/2, N/2 + \sqrt{N}\} \Big\}.$$

Let $\mathcal{A}$ be an algorithm that is given T copies of $|\psi_z\rangle \in \mathcal{C}$ and satisfies the following:

- if $|z^{-1}(1)| = N/2 - \sqrt{N}$, accepts with probability $< 0.2/N$ and
- if $|z^{-1}(1)| = N/2 + \sqrt{N}$, accepts with probability $\in [3.8/N, 4.2/N]$.

Then $T = \Omega(N^c)$ for some $c < 1$.