Learning à la Bernstein-Vazirani

Bernstein-Vazirani algorithm [2]:

Given \( |v_B\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0, 1\}^n} |x, q_x\rangle \), for \( a \in \{0, 1\}^n \), apply the \((n+1)\)-qubit quantum Fourier transform \( \mathcal{H}^{(n+1)} \). This produces the state

\[ \mathcal{H}^{(n+1)} |v_B\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0, 1\}^n} |x, f(x)\rangle, \]

Measure the last qubit in the computational basis. If 0 is observed, abort. If 1 is observed, measure the first \( n \) qubits in the computational basis and output the observed string.

→ Success probability \( \Omega(1) \) with success probability \( 1 - \frac{1}{2^n} \).

What does this have to do with learning?

Change of perspective: \( a \in \{0, 1\}^n \) unknown, given only quantum training data \( |v_B\rangle^{(m)} \)

Bernstein-Vazirani learns a \( n \times 1 \) matrix with success probability \( 1 - \frac{1}{2^n} \).

How easy is quantum learning linear functions?

Biased Quantum Fourier Sampling Linear Functions

We need the biased Fourier coefficients of Boolean linear functions:

\[ \hat{a} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0, 1\}^n} x |a\rangle \langle a| \]

Theorem 3: The function \( g^{(a)} = (-1)^{a(f(x))} \) with \( f^{(a)}(x) := \sum_{j=1}^n a_j x_j \) has \( \mu \)-biased Fourier coefficients \( \hat{g}^{(a)} \hat{a} \)

We can now understand the result of biased quantum Fourier sampling of linear functions:

Corollary 4: Biased quantum Fourier sampling with \( |v_B\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=1}^n \sqrt{D_j(x)} |x, f^{(a)}(x)\rangle \)

How hard is quantum learning linear functions?

Hardness of Learning Linear Functions

So far: How many copies of quantum training data are sufficient for learning? Now: How many copies of quantum training data are necessary for learning?

Classical: At most 1 bit per example → \( \Omega(n) \) training examples to learn \( n \) bits → By the above: Quantum learning does better! But by how much?

Quantum: Quantum exact learning \( \approx \) State identification for training data ensemble → Bound on success probability via Pretty Good Measurement [2] [4] [6] → Bound on PGM success probability using the square root of the Gram matrix of the ensemble, which can be computed inductively:

\[ \rho^{(n+1)} \leq \left( \sqrt{1 + \frac{1}{n}} \right)^n \]

with \( n \) the number of quantum examples.

This gives a quantum sample complexity lower bound for distributions with “large” bias:

Theorem 7: Exact learning of Boolean linear \( n \)-bit functions w.r.t. a distribution with bias \( \mu \geq 1 - \frac{1}{\sqrt{n}} \) requires \( \Omega(g(n)) \) quantum examples.

Using Theorem 5 for bias \( \mu \geq 1 - \frac{1}{\sqrt{n}} \), the comparison between our lower and upper bounds on the sample size \( m \) becomes \( \Omega(l(n)) \leq m \leq O(l(n)) \), so \( m = \Theta(polylog(n)) \). Thus we have an almost tight characterization for “large” bias.