# small quantum computers and big classical data

Aram Harrow (MIT)
QC Theory in Practice
9 April 2019

# quantum algorithms models

| Model | Definition | Example |
| --- | --- | --- |
| Standard | Input: $x \in \{0,1\}^n$<br>Output: $y \in \{0,1\}$ | factoring / number theory<br>combinatorial optimization |
| Oracle | given access to O<br>$O \|i,a\rangle = \|i,a+x_i\rangle$ | Grover, NAND tree, collision, ...<br>Hidden subgroup problem, welded trees |
| Quantum data | Given $\sum_i x_i \|i\rangle$ | QFT, $e^{-iHt}$<br>SWAP test, Schur transform, tomography<br>linear systems |
| Quantum oracle | Given U | phase estimation<br>quantum sensing, process tomography<br>qubitization, singular value transform |

# quantum search

| i | f(i) |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 1 |
| 5 | 0 |
| … | … |
| n | 0 |

Given the ability to compute $f(i) := x_i$, find $i^*$ such that $f(i^*) = 1$.

classical: O(n) time needed

Grover's algorithm: (1996)

$O(\sqrt{n})$ on quantum computer

similar speedups for maximizing, approximate counting, collisions, triangle finding, game-tree evaluation, sampling, backtracking algorithms, rapidly mixing Markov chains, …

# oracle search?

usual model

"oracle" = subroutine
$O|i\rangle|0\rangle = |i\rangle|x_i\rangle$

→ provable speedups, sometimes exponential

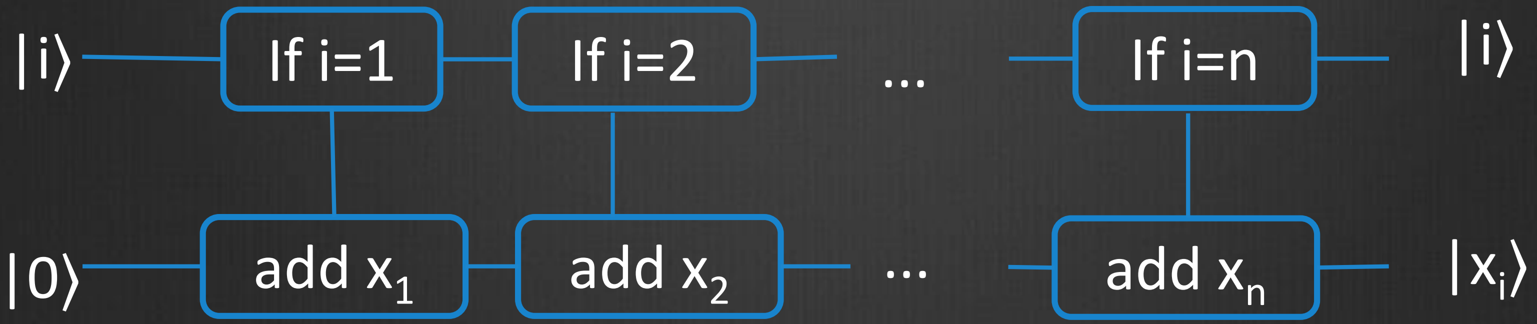
However:

- We cannot query a data center in superposition.
- Classical memory is ≈ parallel.
  For large enough n, n bits of memory ←→ O(n) CPUs.
- Proposed "quantum RAM" could be queried in superposition, but is not easy to build.

# oracles from classical memories

$x_1, \ldots, x_n$

$O|i\rangle|0\rangle = |i\rangle|x_i\rangle$



# Big data quantum speedups?

Queries with overhead O(n) mean no oracle speedup.

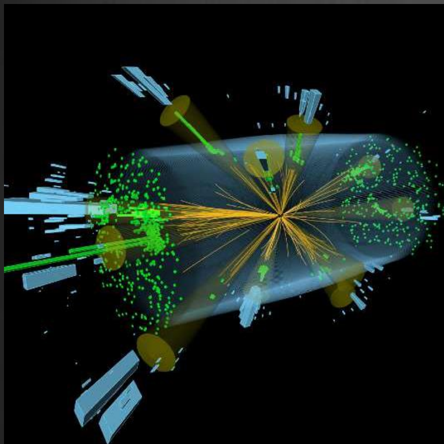
Usual solutions:

1. QRAM: quantumly-accessible database
2. $x_i = f(i)$ for efficient f, e.g. $f(i) = a^i \bmod N$.

This talk:

1. Problems where data access doesn't dominate:
   -example from computing "thrust" in collider data.

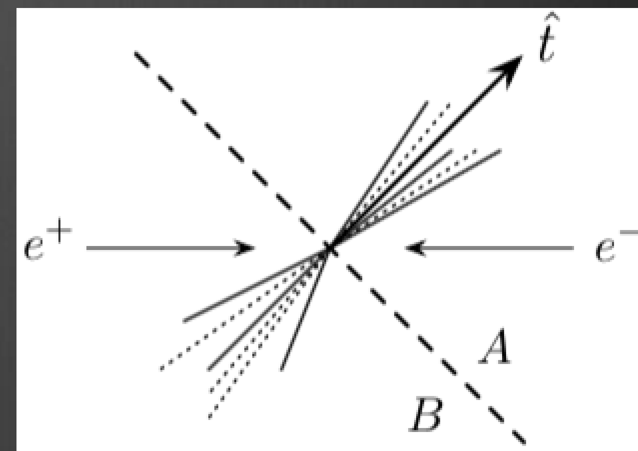
2. Problems where we can reduce the size of the data set
   -clustering, Bayesian inference, saddle-point optimization.



# Jet clustering

joint work with
Naik, Thaler, Wei

How "jet-like" is an LHC event?




[Farhi, 1977]

Def: Given particle momenta $p_1, \ldots, p_n \in \mathbb{R}^3$.
thrust = $\max_t \sum_k |t \cdot p_k|$ over unit vectors $t$.

Fact: optimal $t$ is $t_{ij} := f(p_i \times p_j)$ for some i,j.

O(n³) classical algorithm:
For all i,j evaluate $\sum_k |t_{ij} \cdot p_k|$.

O(n²) quantum algorithm:
Grover over i,j with O(n) iterations
- Compute $p_i, p_j$ and $t_{ij}$ in time O(n).
- evaluate $\sum_k |t_{ij} \cdot p_k|$ in time O(n).

O(n) memory cost doesn't hurt us

# statistics using data

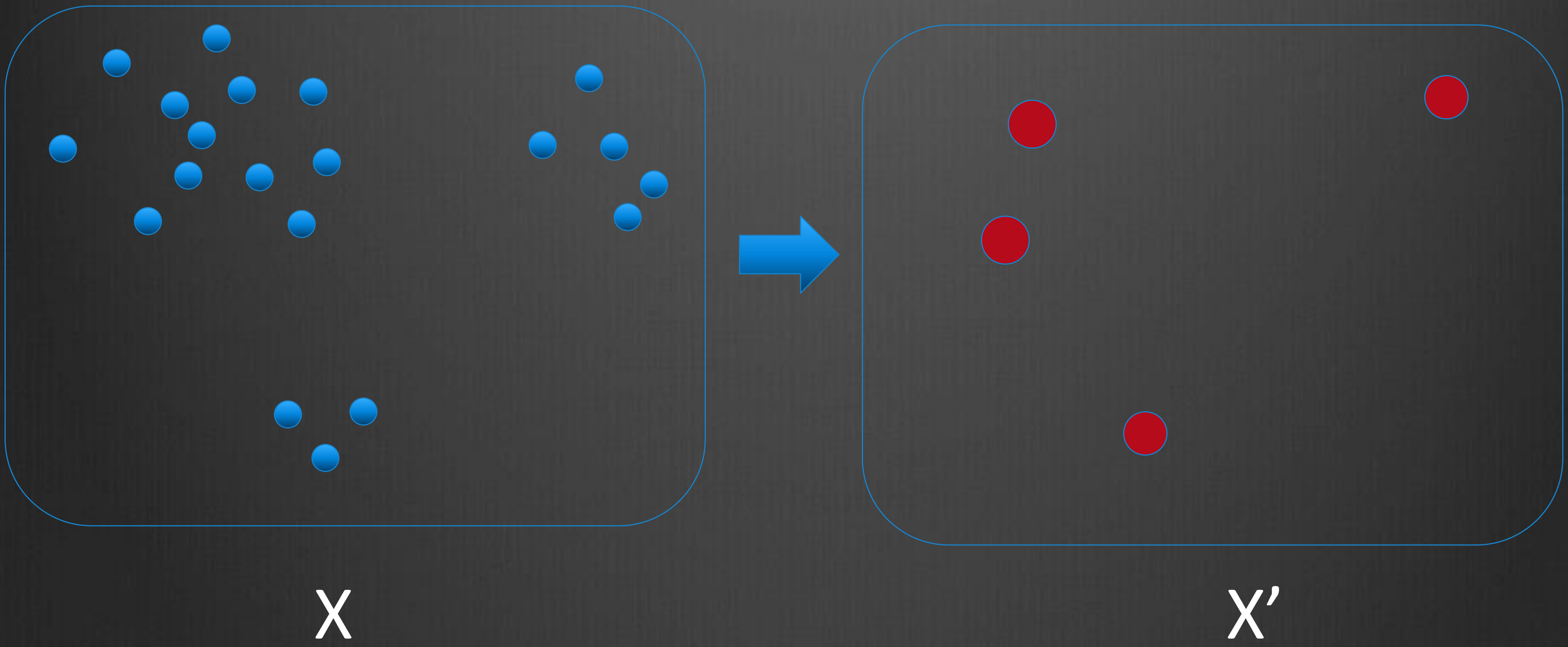
Maximum-likelihood estimation:
Given a set of models Y and a data set X,
Compute $\text{argmax}_y \, [r(y) + \Sigma_{x \in X} f(x,y)]$
where $f(x,y) = \log p(x|y)$.

X, Y appear almost symmetrically. (max vs ∑ don't matter for Grover)
but
X lives on a hard drive → no oracle access or superposition
Y is synthetic (e.g. weights and centers of clusters) → can superpose

Classical computer controlling a quantum computer:
Can run Grover with $O(|X| \cdot |Y|^{1/2})$ quantum evaluations of f.

key question: Can we reduce the dependence on |X|?

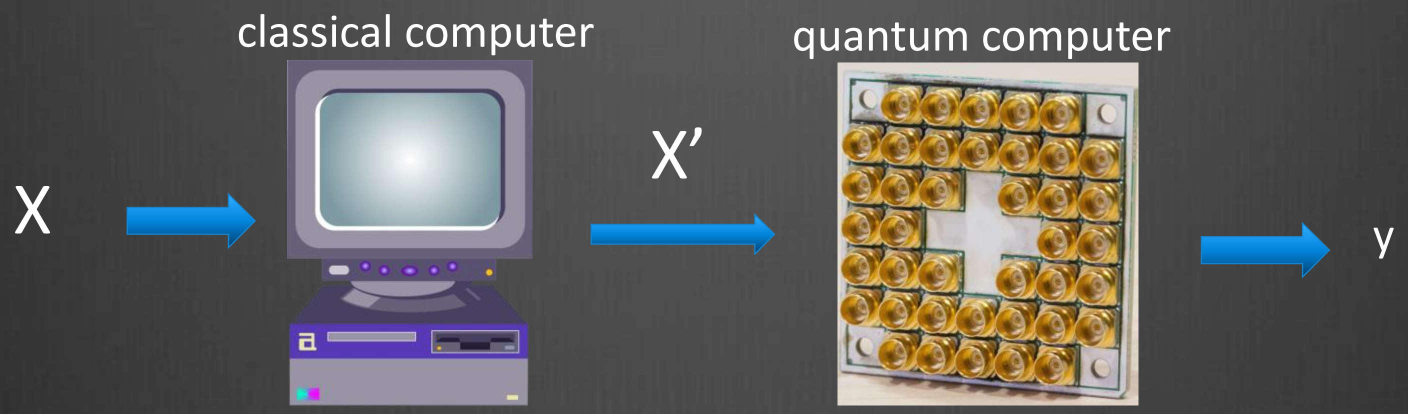
# data reduction: "coresets"




X' is a coreset if for all y,

$\Sigma_{x \in X} f(x,y) \approx \Sigma_{x \in X'} w(x)\, f(x,y)$.

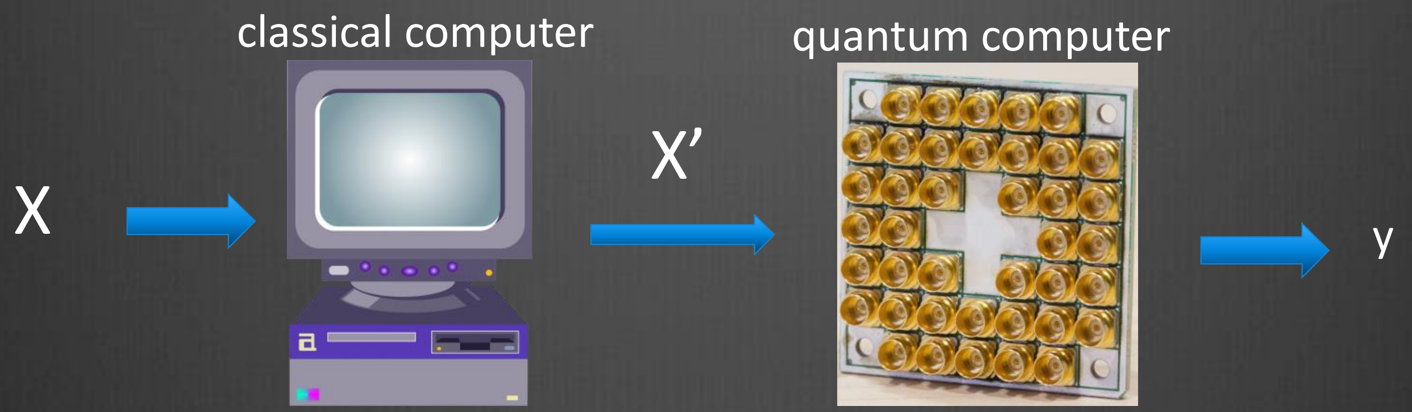
# hybrid algorithms for machine learning



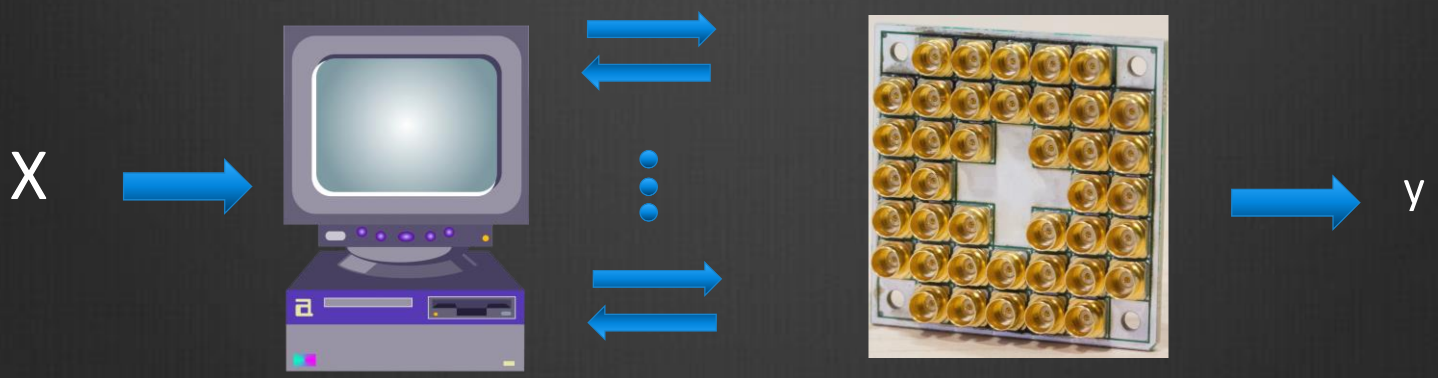

- Finds quick, crude, guess of y’.
- Use y’ to construct importance sampling weights.
- Draw X’ using importance sampling.

- Use Grover / quantum walks / adiabatic / etc to find y.
- Inner loop cost is $O(|X'|)$.
  Total cost for Grover $O(|X'|\cdot|Y|^{1/2})$.
- Provable speedups include clustering.

# adaptive data reduction




Data set is already small by the time we turn on the quantum computer.




Are there benefits to using the quantum computer interactively?

# adaptive coresets

[Campbell, Broderick 2017] [Bugalo, Elvira, Martino, Luengo, Míguez, Djuric 2017]

Bayesian inference: sample y from

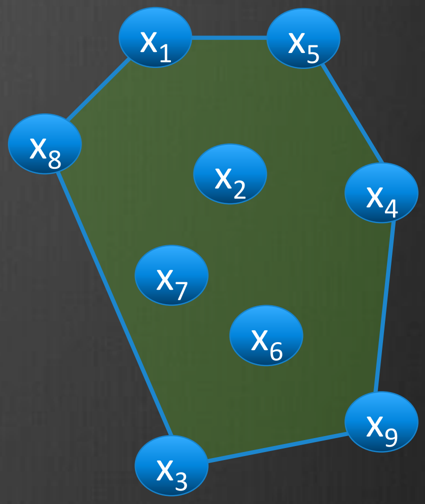
$$\pi(y) = \frac{\pi_0(y)\exp\left(\sum_{x\in X} f(x,y)\right)}{Z}$$

data $x_1, \ldots, x_t$
weights $w(x_1), \ldots, w(x_t)$

sample $y_t$ from $\pi_t(y_t) = \frac{\pi_0(y)\exp\left(\sum_{i=1}^{t} w(x_i) f(x_i, y)\right)}{Z}$

Choose $x_{t+1}$, $w(x_{t+1})$, using $y_1, \ldots, y_t$, $x_1, \ldots, x_t$, $w(x_1), \ldots, w(x_t)$

# saddle-point optimization




X = convex hull of $\{x_1, ..., x_n\}$
Y = probability distributions over m items
f(x,y) linear in x,y

"minimax": $\max_y \min_x f(x,y)$
$= \min_x \max_y f(x,y)$

Family includes:

- zero-sum games
- linear programming (X = linear constraints on Y)
- ≈max. likelihood est. (when ∑ is dominated by the min)
- $L_1$ kernel SVM (ask me in person)

# hybrid algorithm for minimax

Thm [Grigoriadis-Khachiyan '95]:
Error is $\leq \varepsilon \max_{xy}|A_{xy}|$ after
$T = O(\log(|X|\cdot|Y|)/\varepsilon^2)$ rounds

Runs on hybrid classical-quantum computer.

- T rounds of alternation.
- Classical computer iterates through data set in each round.
- Quantum computer uses Grover (or other q algorithm) with O(T) calculations in inner loop.

# Perspective

- I used Grover for concreteness but same ideas apply to adiabatic algorithm, QAOA, quantum walks, etc.

- Useful lower bounds in this data model don't look easy. May need conjectures like 3SUM.

- QCs will augment not replace classical computers.
Let's design algorithms as though we believe this!