QUANTUM RECOMMENDATION SYSTEMS

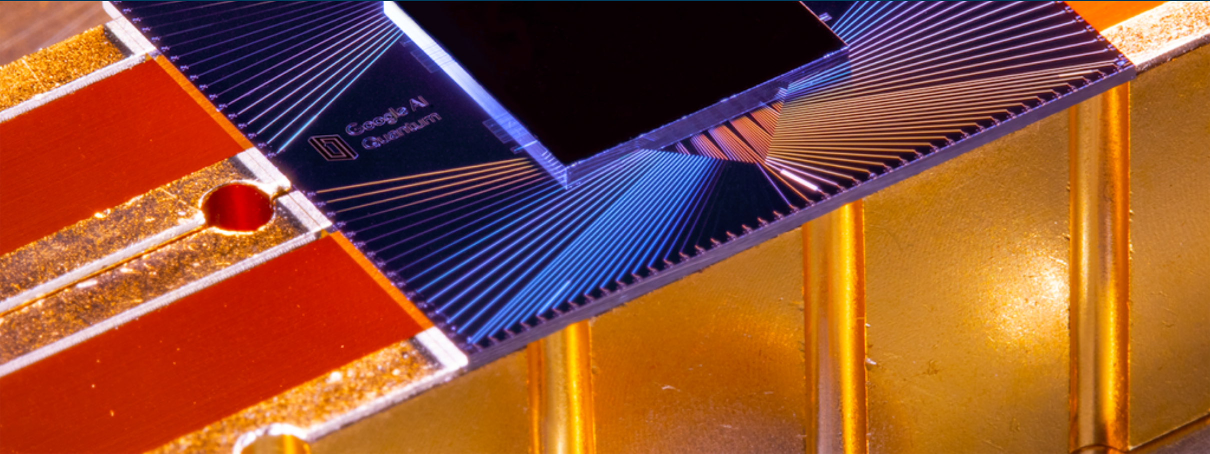
I. Kerenidis, A. Prakash (2016)

ÉCOLE POLYTECHNIQUE
IP PARIS

Alex Vigneron

March 16, 2021

1
INTRODUCTION



OVERVIEW ★

Definition of RS

A recommendation system helps users find compelling content in a large corpora - Google developpers

What?

Quantum-enhanced RS.

Input : a preference matrix

Output : one recommended item likely to be interesting

Why?

RS are everywhere :

media & entertainment

product & services

advertisement

...

Current classical approach is costly

How?

Using **HHL-derived** approach to LA problems, **phase estimation** and a **Grover-like** memory model

PREFERENCE MATRIX A ★

$$
M \text{ users} \;
\begin{array}{c|cccccc}
 & \multicolumn{6}{c}{N \text{ series}} \\
 & \textit{TheWitcher} & \textit{Rick\&Morty} & \textit{Ozark} & \textit{BreakingBad} & \textit{TheCrown} & \dots \\
\hline
\textit{Alice} & 0.5 & ? & 0.7 & 0.9 & 0.2 & \dots \\
\textit{Bao} & 0.5 & 0.6 & 0.5 & 0.2 & 0.5 & \dots \\
\textit{Charlie} & ? & 0.7 & 0.2 & 0.1 & 0.0 & \dots \\
\textit{Djamel} & 0.8 & 0.2 & ? & 0.8 & 0.1 & \dots \\
\textit{Esther} & ? & 0.5 & ? & ? & 0.4 & \dots \\
\textit{Francesca} & 0.7 & 0.2 & 0.6 & 0.7 & 0.1 & \dots \\
\dots & \dots & \dots & \dots & \dots & &
\end{array}
$$

A_{ij} whether user i likes product j

$A_{ij} = 1 \mapsto$ Likes

$A_{ij} = 0 \mapsto$ Dislikes

$$
k \text{ user types} \;
\begin{array}{c|cc}
 & \multicolumn{2}{c}{k' \text{ series types}} \\
 & Comedy & Fantasy \\
\hline
(< 20y) & 0 & 0 \\
(20 < y < 35) & 1 & 0 \\
(35+) & 1 & 0
\end{array}
$$

CLASSICAL RECOMMENDATION SYSTEMS

Matrix reconstruction

SVD
alternating minimization
$\mathcal{O}(k^l)$ **poly complexity!**

Matrix sampling

was as hard as matrix reconstruction
SVD + projection onto top-k

RL bandits

RL bandits : lin-UCB, KL-UCB, Thompson Sampling, ...

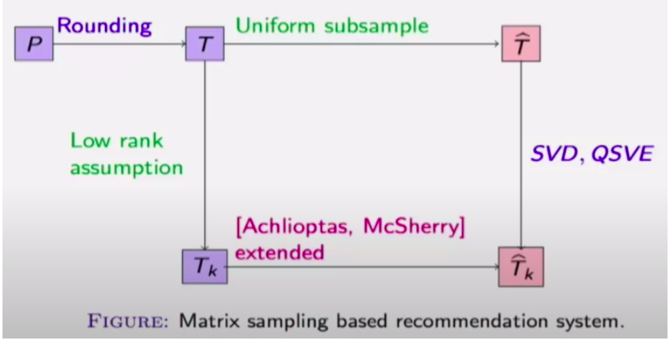
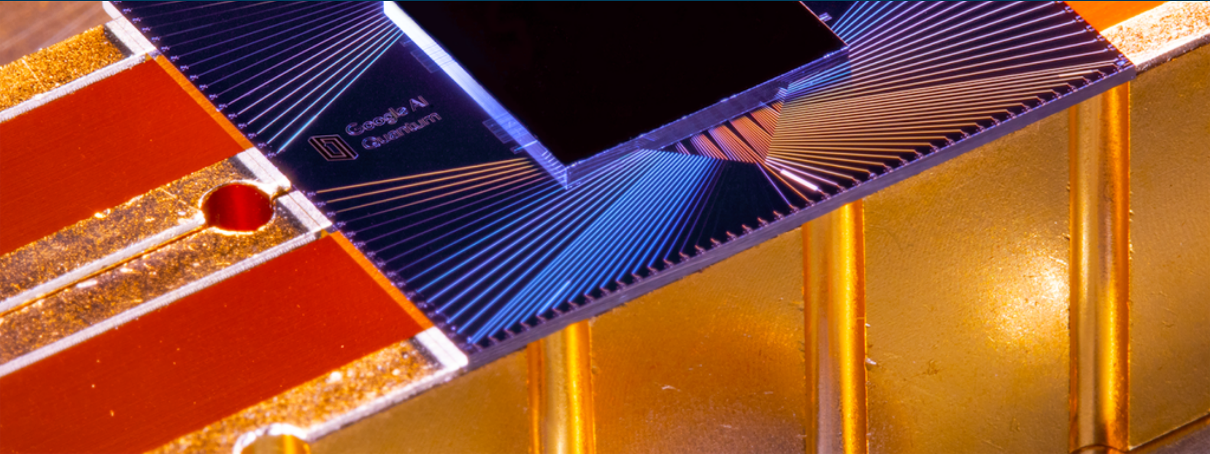


Figure: A. Prakash's talk at Microsoft research (2017)

2
ALGORITHMS



ALGORITHM: GENERALITIES

Time complexity

$\mathcal{O}(\frac{k^{\frac{1}{2}}}{\epsilon} log^l(mn))$

Polylogarithmic in dimensions of preference matrix

Polynomial in the rank k of approximation ($M \rightarrow k$ users)

Complexity novelty

Classical complexity $\mathcal{O}(mk)$

First algo polylog in matrix dim.

Disclaimer

Perf. depends on how good the LR approximation is

ASSUMPTIONS ★

Existence of a *good* LR approximation
With rank $k < 100$
small number of user/product categories

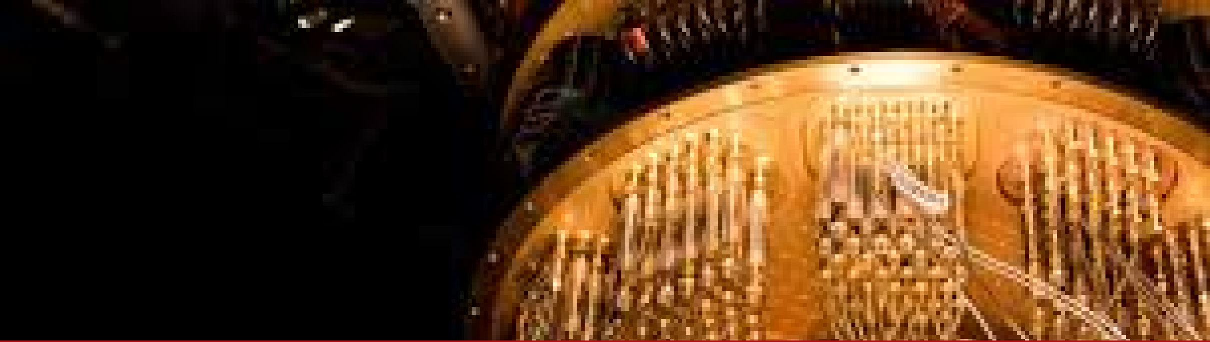
Knowledge of rank k
Or possibility to estimate it properly.

Knowledge of $||A||_F$
The Frobenius norm of matrix A is considered known.

Uniformity of subsampling
Each element is i.i.d.

Similar number of prefs.
Approx. uniformity of number of preferences per user

QRAM data structure D
Algorithm has oracle access to data structure.
Samples arrive online, update time $\mathcal{O}(log^2(mn))$



CLASSICAL ALGORITHMS

X

SINGULAR VALUE DECOMPOSITION

Matrix factorisation which generalises eigendecomposition of PCA to any dimension matrix

$$\underbrace{A}_{\text{m n}} = \underbrace{U}_{\text{m m}}\underbrace{S}_{\text{m n}}\underbrace{V^T}_{\text{n n}}$$

$$A = \sum_i s_i u_i v_i^T$$

Notation

U : left sing.vects.
S : diagonal matrix of sing. vals.
V^T : right sing. vects.

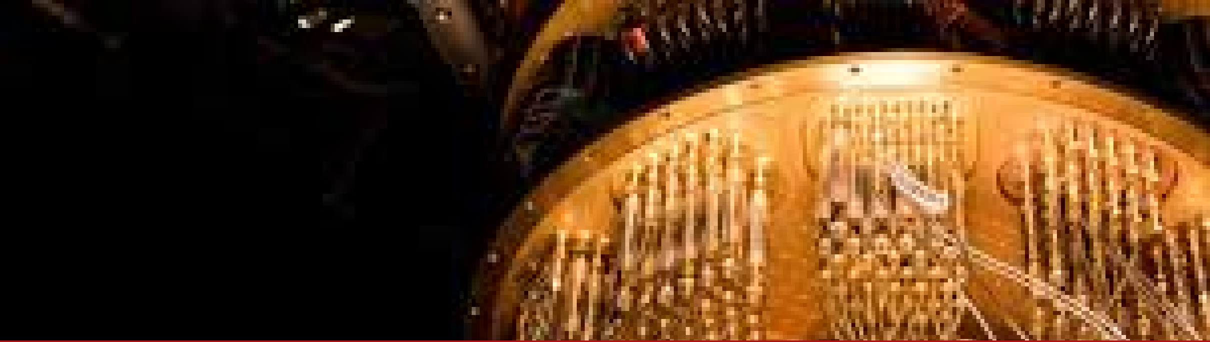
MATRIX RECONSTRUCTION

Algorithm 1: Matrix reconstruction

Input: A, $samplingprobability$p $\in [0,1]$
Output: $\hat{A}_k$
while *needed* **do**
  Sample an element $e_i \in A$ w.p. p
  $\tilde{e}_i \leftarrow$ rescaled e_i w.p. p ; 0 w.p. $1-p$
  Store $\tilde{e}_i$ into $\hat{A}$
end
$K \leftarrow$ top k sing. vects. from SVD
$\hat{A}_k \leftarrow$ projection of $\hat{A}$ onto K
return $\hat{A}_k$

Approximation error

LRA $\exists\epsilon \leq 1|\ ||A - A_k||_F \leq \epsilon||A||_F$
$\implies$ projecting onto top k sing.vects of $\hat{A}$ is sufficient
Sampling. Unif.
$||A - A_k|| \approx ||A - \hat{A}_k||$
$E[\hat{A}] = E[A]$



QUANTUM ALGORITHMS

X

GENERAL INTUITION ★

We want to...

Sample from the recommendation matrix

We can...

- Approximate the true recommendation matrix with a **lower-rank**, **sub-sampled** matrix
- Sample from that matrix

$$A \to \hat{A}, A \to A_k$$

$$\hat{A}, A_k \to \hat{A}_k$$

So we...

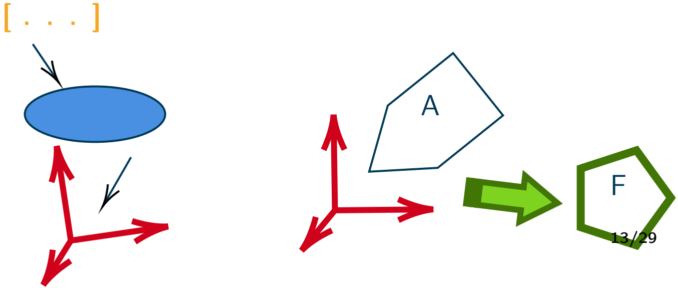
Select an interval of acceptable singular values $\tilde{\sigma}$

Select the singular vectors ξ corresponding to $\tilde{\sigma}$

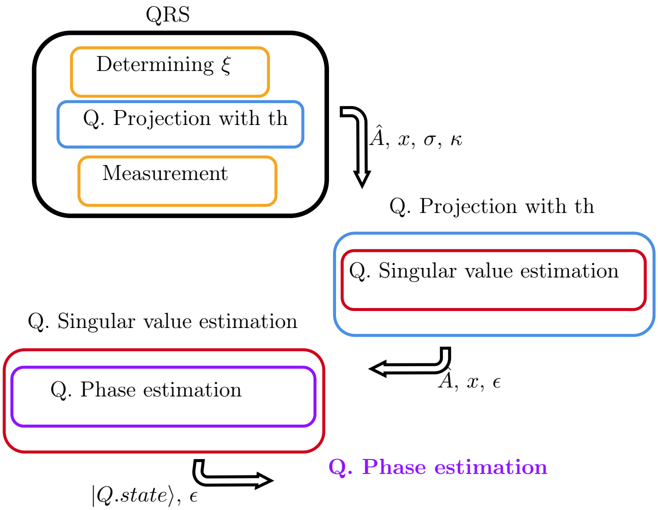
Obtain a space $\mathcal{S}$ spanned by ξ

Project matrix A onto $\mathcal{S}$

Consider the resulting family of matrices $\mathcal{F}$.



ALGORITHM'S FLOW ★



ALGORITHMS: QUANTUM RECOMMENDATION SYSTEM ★

Algorithm 2: Quantum recommendation algorithm

Input: subsample matrix $\hat{A}$ stored in data struct. $\mathcal{D}$, sampling probability $p \in [0,1]$ a vector $x \in \mathbb{R}^n$, user index $i \in \mathbb{N}$, $\epsilon \in \mathbb{R}$

Output: A recommended product ρ

Fix $\sigma \in \mathbb{R} \leftarrow f(\hat{A}, \epsilon, k, p)$

Fix $\kappa \in \mathbb{R}$

$|\phi\rangle \leftarrow$ QPROJ($\hat{A}$, x, σ, κ)

$\rho \leftarrow$ measure $|\phi\rangle$ in computational basis.

return ρ

Time complexity

Loglinear

$\mathcal{O}(\frac{k^{\frac{1}{2}}}{\epsilon} log^l(mn))$

QUANTUM PROJECTION

Algorithm 3: Q. Projection with threshold

Input: $\hat{A}$, $x \in \mathbb{R}^n$, $\sigma \in \mathbb{R}$, $\kappa \in \mathbb{R}$
Output: quantum state $|\phi\rangle$
$\epsilon \leftarrow f(\hat{A}, \kappa, \sigma)$, generate
V, done $\leftarrow$ False
while *not done* **do**
  Create $|x\rangle = \sum_i \alpha_i |v_i\rangle$
  Create $|x'\rangle = \sum_i \alpha_i |v_i\rangle |\bar{\sigma}_i\rangle$ with $\bar{\sigma} \leftarrow$ QSVE(ϵ, $\hat{A}$, x)
  Apply **V** $|reg.2\rangle$ to get :
   $|x''\rangle = \sum_g \alpha_i |v_i\rangle |\sigma_i\rangle + \sum_b \alpha_i |v_i\rangle |\sigma_i\rangle$
  Apply QSVE on $|x''\rangle$ to erase reg.2
  $m \leftarrow$ measure reg.2 in computational basis
  if $m = |0\rangle$ **then**
   $|\phi\rangle \leftarrow reg.1$, $done \leftarrow$ True
  end
end
return $|\phi\rangle$

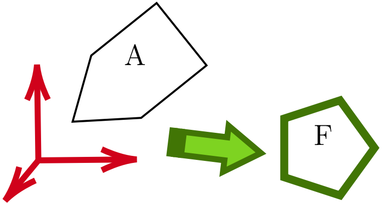
Time complexity

$\mathcal{O}(log^{l'}(mn))$

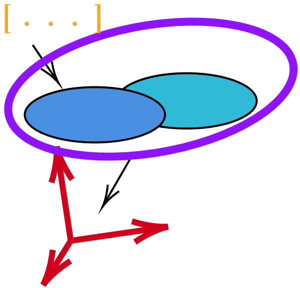
Notation

$g = i|\sigma_i \in \tilde{\sigma}$

$b = i|\sigma_i \notin \tilde{\sigma}$



DETERMINING $\tilde{\sigma}$ AND BUILDING V



Based on : σ of $\hat{A}$ and κ
$\kappa \in \mathbb{R}$, ex. $\frac{1}{3}$
σ the minimal threshold singular value
$\bar{\sigma} = cos(\frac{\bar{\theta}}{2})||A||_F$ the estimated singular values
$\tilde{\sigma} = \{\bar{\sigma_i} \in [(1-\kappa)\sigma, \sigma]\} \cup \{\bar{\sigma_i} \geq \sigma\}$
$\xi = \{v_i | \sigma_i \in \tilde{\sigma}\}$

$$V(\text{unitary}):$$

$$\begin{array}{ll} \text{if t} < (\sigma - \frac{\kappa}{2}\sigma) & |t\rangle|0\rangle \mapsto |t\rangle|1\rangle \\ \text{else} & |t\rangle|0\rangle \mapsto |t\rangle|0\rangle \end{array}$$

QUANTUM SINGULAR VALUE ESTIMATION

Algorithm 4: Q. Singular Value Estimation

Input: Subsample mat. $\hat{A}$, $x \in \mathbb{R}^n, \epsilon \in \mathbb{R}$
Output: $\emptyset$
Side Effect: Encodes $\bar{\sigma}_i$ into the quantum state
while *not done* **do**
    Create $|x\rangle = \sum_i a_i |v_i\rangle$
    Append $|0\rangle^{\otimes log(m)}$ to reg.1
    Create $|\mathbf{Q}x\rangle \leftarrow \sum_i a_i |\mathbf{Q}v_i\rangle$ (1)
    Step Γ **:** $\bar{\theta}_i \leftarrow$ **QPE**$(|Qx\rangle, W)$
    Obtain state $\sum_i \alpha_i |Qv_i, \bar{\theta}_i\rangle$
    Compute $\bar{\sigma}$ based on $\bar{\theta}$
    Uncompute output of QPE
    Apply inverse of (1), get back the resulting state $\sum_i \alpha_i |v_i\rangle |\bar{\sigma}_i\rangle$
end

Time complexity

$\mathcal{O}(\frac{log^l(mn)}{\epsilon})$

Correctness

$|\bar{\sigma}_i - \sigma| \leq \delta \; \forall i$ w.p. $1 - \frac{1}{\prime(n)}$

Link with QRAM

For efficiency, the QRAM should be designed with a particular factorisation in mind to allow for Γ easily
res.1 "Quantum GD for linear systems and least squares" (I.K., A.P.)

PHASE ESTIMATION

Algorithm 5: Q. Phase Estimation

Input: Eigenvector as quantum state $|u\rangle$ of unitary U s.t. $\lambda = e^{2\pi i\theta}$

Output: θ, phase of λ

Setup reg.1 to $|0\rangle^{\otimes t}$

Reg.2 $\leftarrow |u\rangle$

Apply the Walsh-Hadamard transform to reg.1

Apply n ctrl-U^{2_j} operations on reg.2, controlled by reg.1

Apply $QFT^\dagger$ on reg.1

$\theta \leftarrow$ measure reg.1 in computational basis

return θ

Time complexity

$\mathcal{O}(t^2)$

STEP Γ (LEMMA 5.3)

Perform QPE on the state $|Qx\rangle$ with the unitary W

Step Γ : $\bar{\theta}_i \leftarrow$ QPE($|Qx\rangle, W$)

Q

$A \in \mathbb{R}^{mn}$ has a svd $\iff \exists P, Q$ factorisation matrices

$P, Q | \frac{A}{||A||_F} = P^T Q$

$P^T P = \mathbb{I}_m$

$Q^T Q = \mathbb{I}_n$

Isometry $Q : \mathbb{R}^n \to \mathbb{R}^{mn}$

maps a row sing. vect. $v_i \in A$ with sing. val σ_i to an eigenvector Qv_i of W with eigenvalue

$e^{i\theta_i} | cos(\frac{\theta_i}{2}) = \frac{\sigma_i}{||A||_F}$

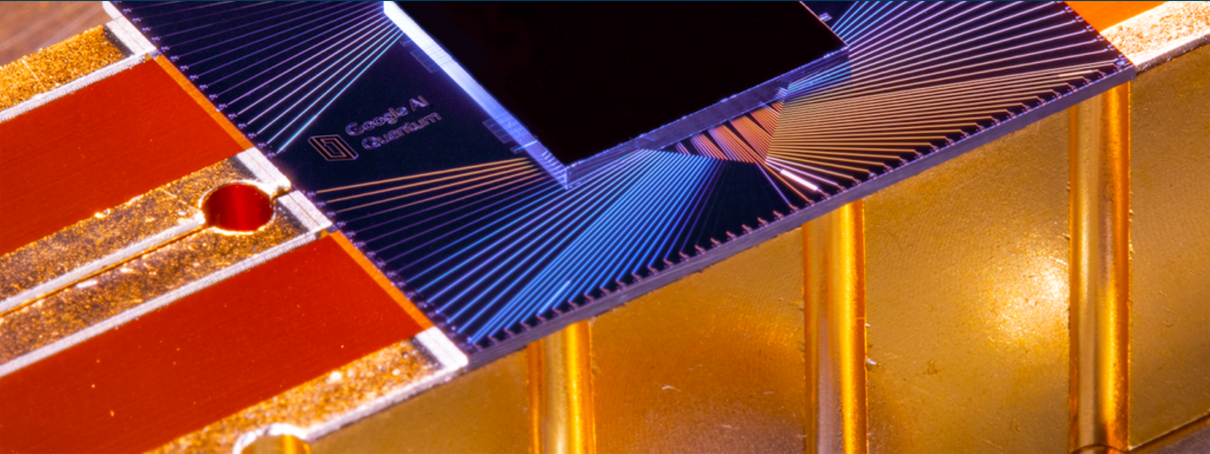
W

$W = UV$

$U = 2PP^T - \mathbb{I}_{mn}$

$V = 2QQ^T - \mathbb{I}_{mn}$ W is a reflection on the column space of Q followed by a reflection on the column space of P

3
EXTRAS



COMPLEXITY ANALYSIS

QSVE

$\mathcal{O}(\frac{log^l(mn)}{\epsilon})$
or
$\mathcal{O}(\frac{log^l(mn)}{\epsilon^3})$ (Lloyd *et. al.*)

QPE

$\mathcal{O}(t^2)$
where t is the number of qubits of reg.1

Q.Proj with th

$\mathcal{O}(\frac{log^l(mn)||A||_F||x||^2}{\sigma||T||^2})$
where T is the state outputed by Q.Proj with th

Q. Recommendation System

$\mathcal{O}(k^l log^{l'}(mn))$
Polylogarithmic in dimensions of preference matrix
Polynomial in the rank k of approximation
Loglinear

Classical LRA

$\mathcal{O}(n)$, $o(n)$

QRAM DATA STRUCTURE $\mathcal{D}$

Quantum operations

A Q.algorithm with acces to $\mathcal{D}$ can perform Unitary mappings & create state $x \mapsto |x\rangle$ in $\mathcal{O}(log^l(mn))$

Format

Array of m binary trees (one per row of A)

Level : ordered list

Leaves : squared amplitude

Internal nodes : sum of leaves

Extra node : max row norm of A

Binary tree

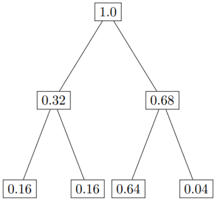


Figure: arxiv.org/pdf/1603.08675.pdf

Time complexity

element insertion $\mathcal{O}(log^2(mn))$ classically worst $\mathcal{O}(h)$, average $\mathcal{O}(d)$ where d is the number of entries

creation $\mathcal{O}(w\, log^2(mn))$ where w denotes pre-existing entries

FROM CLASSICAL TO QUANTUM... AND BACK!

Classical algorithm running in $\mathcal{O}(k^j log^l(mn))$.
Ewin Tang, U. Washington (2018)

Algorithm 4: Low-rank approximation sampling

Input: Matrix $A \in \mathbb{R}^{m \times n}$ supporting the operations in 4.2, user $i \in [m]$, threshold σ, $\varepsilon > 0$, $\kappa > 0$

Output: Sample $s \in [n]$

Run ModFKV (3) with $(\sigma, \varepsilon, \kappa)$ parameters as $(\sigma(1-\kappa/2), \varepsilon, 2\kappa/(2-\kappa))$ to get a description of $D = A\hat{V}\hat{V}^T$;

For the following, simulate $\hat{V} \in \mathbb{R}^{n \times k}$ from the description as described in Proposition 6.14;

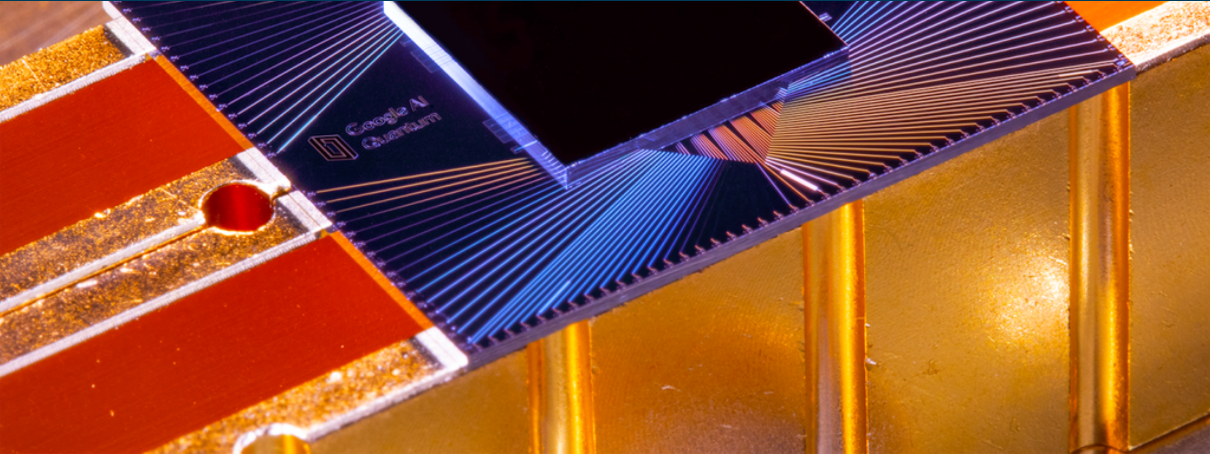
Use Proposition 6.2 with parameters $\frac{\varepsilon}{\sqrt{k}}$ to estimate $\langle A_i, \hat{V}^{(t)} \rangle$ for all $t \in [k]$;

Let est be the $1 \times k$ vector of estimates: est $= \{\langle A_i, \hat{V}^{(t)} \rangle\}_{t \in [k]}$;

Use Lemma 6.8 to get a sample s from est$\hat{V}^T$;

Output s;

4
CONCLUSION



TAKE HOME MESSAGE ★

Repo
https://github.com/doctor-who-42/quantum_reco_systems_pres

General subroutine for LA
SVE $\mathcal{O}(\frac{log^l(mn)}{\epsilon})$

Complexity
$\mathcal{O}(\frac{k^{\frac{1}{2}}}{\epsilon}log^l(mn))$

Class. $\rightarrow$ Q. $\rightarrow$ Class.
Tang's algorithm

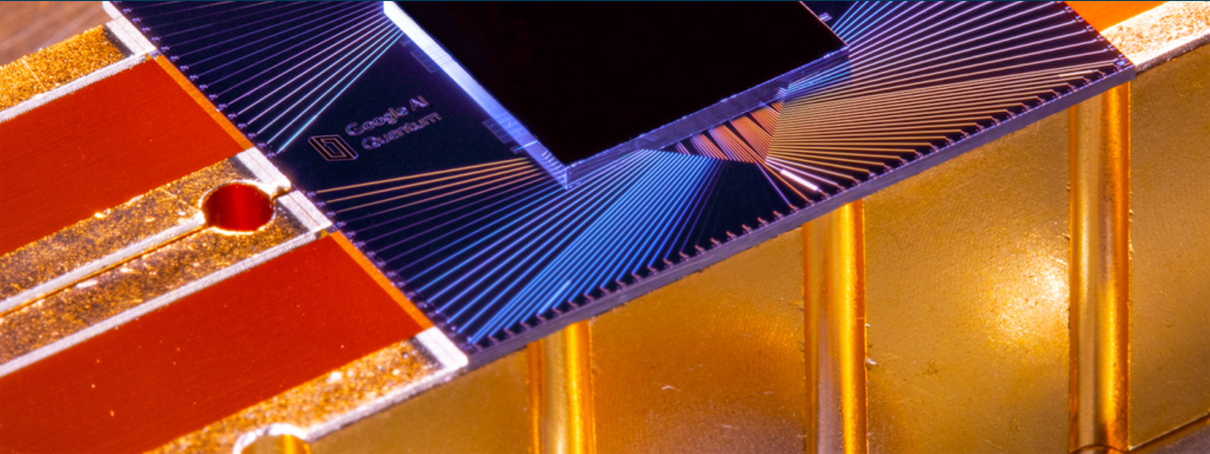
Knowledge of $||A||_F$
The Frobenius norm of matrix A is considered known.

Semantic of 0s
Disliked **and** unknown entries $= 0$

QRAM
Necessity to implement it with specific SVE in mind

5

MORE!



DETAILED QUANTUM PHASE ESTIMATION

More!

Algorithm 6: Q. Phase Estimation

Input: Eigenvector as quantum state $|u\rangle$ of unitary U s.t. $\lambda = e^{2\pi i\theta}$
Output: θ, phase of λ
Prepare 2 quantum registers:
Reg.1 set to $|0\rangle^{\otimes t}$
Reg.2 $\leftarrow |u\rangle$
if *eigVect is not known* **then**
 decompose in eigenbasis as $|u\rangle = \sum_i c_i |u_i\rangle$
Apply the Walsh-Hadamard transform to reg.1 to get $\frac{1}{2^n}\sum_x |x\rangle \otimes |u\rangle$
Apply to reg.2 controlled unitaries, controlled by reg.1 to get $\frac{1}{2^n}\sum_x e^{ix\theta}|x\rangle \otimes |u\rangle$
Apply $QFT^\dagger$ to reg.1 to get $\frac{1}{2^n}\sum_y \sum_x e^{\frac{-2\pi ixy}{2^n}} e^{ix\theta}|y\rangle \otimes |\phi\rangle$
Obtaining through continuous fraction from reg.1 $\theta = \frac{a}{2^n} + \delta$ where $\delta \in [0, \frac{1}{2^{n+1}}]$,
 rewrite as $\frac{1}{2^n}\sum_y e^{-2\pi i\delta y}|a\rangle \otimes |\phi\rangle = \frac{1}{2^n}\frac{1-e^{2\pi i\delta 2^n}}{1-e^{2\pi i\delta}}|a\rangle \otimes |\phi\rangle$
if $\delta = 0$ **then**
 the phase is obtained with $p = 1$
else
 the phase is approximated with $p = \frac{1}{2^n}\frac{1-e^{2\pi i\delta 2^n}}{1-e^{2\pi i\delta}}^2$
return θ

QUANTUM FOURRIER TRANSFORM CIRCUIT

More!