

Frontiers of Hamiltonian Learning

Ainesh Bakshi

NYU

Quantum Learning: Two Paradigms

Classical Input Quantum Algorithms



Using quantum algorithms
to analyze classical data

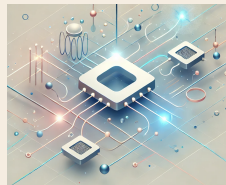
Quantum Learning: Two Paradigms

Classical Input Quantum Algorithms



Using quantum algorithms
to analyze classical data

Quantum Input Quantum Algorithms



Learning properties of
quantum systems

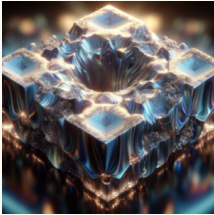
Schrödinger Equation

$$i\hbar \frac{d}{dt} |\psi\rangle = H |\psi\rangle$$

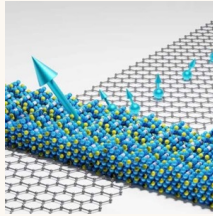
Learning Laws of Physics

Schrödinger Equation

$$i\hbar \frac{d}{dt} |\psi\rangle = H |\psi\rangle$$



Exotic Phase



Quantum Material



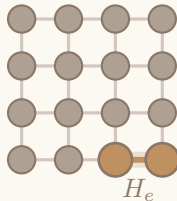
Quantum Device

Background and Setup: Local Hamiltonians

- n particles $\implies$ Hamiltonian $H \in \mathbb{C}^{2^n \times 2^n}$
- H is a sum over local Hamiltonians:

$$H = \sum_{e \in E} \lambda_e \cdot H_e \quad \text{where } H_e \in \mathbb{C}^{2^n \times 2^n} \text{ and } \|\lambda_e\| \leq 1$$

where H_e acts on the two particles at edge e



Background and Setup: Local Hamiltonians

- n particles $\implies$ Hamiltonian $H \in \mathbb{C}^{2^n \times 2^n}$

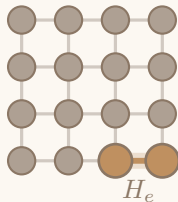
- H is a sum over local Hamiltonians:

$$H = \sum_{e \in E} \lambda_e \cdot H_e \quad \text{where } H_e \in \mathbb{C}^{2^n \times 2^n} \text{ and } \|\lambda_e\| \leq 1$$

where H_e acts on the two particles at edge e

- Ex. $H_{n-1,n} = I \otimes I \otimes \cdots \otimes X_{n-1} \otimes X_n$, where

$$X_i \in \left\{ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\}$$



Background and Setup: Local Hamiltonians

- n particles $\implies$ Hamiltonian $H \in \mathbb{C}^{2^n \times 2^n}$

- H is a sum over local Hamiltonians:

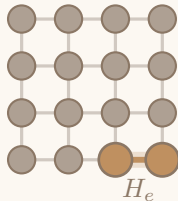
$$H = \sum_{e \in E} \lambda_e \cdot H_e \quad \text{where } H_e \in \mathbb{C}^{2^n \times 2^n} \text{ and } \|\lambda_e\| \leq 1$$

where H_e acts on the two particles at edge e

- Ex. $H_{n-1,n} = I \otimes I \otimes \cdots \otimes X_{n-1} \otimes X_n$, where

$$X_i \in \left\{ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\}$$

- X_i 's are Pauli Matrices



$$\begin{pmatrix} Z & & & \\ & Z & & \\ & & \ddots & \\ & & & Z \end{pmatrix}$$

$$Z = X_{n-1} \otimes X_n$$

Background and Setup: Gibbs States

- A quantum state is a matrix ρ such that $\rho \succeq 0$ and $\text{Tr}(\rho) = 1$

Background and Setup: Gibbs States

- A quantum state is a matrix ρ such that $\rho \succeq 0$ and $\text{Tr}(\rho) = 1$
- The state of a quantum system at thermal equilibrium at inverse temperature β is given by

$$\rho = \frac{\exp(-\beta H)}{\text{Tr}(\exp(-\beta H))}$$

Matrix exponential:

$$\begin{pmatrix} e^{-\beta\gamma_1} & & & \\ & e^{-\beta\gamma_2} & & \\ & & \ddots & \\ & & & e^{-\beta\gamma_{2^n}} \end{pmatrix}$$

Background and Setup: Gibbs States

- A quantum state is a matrix ρ such that $\rho \succeq 0$ and $\text{Tr}(\rho) = 1$
- The state of a quantum system at thermal equilibrium at inverse temperature β is given by

$$\rho = \frac{\exp(-\beta H)}{\text{Tr}(\exp(-\beta H))}$$

- This is called the Gibbs state and generalizes the notion of a Gibbs distribution

Matrix exponential:

$$\begin{pmatrix} e^{-\beta\gamma_1} & & & \\ & e^{-\beta\gamma_2} & & \\ & & \ddots & \\ & & & e^{-\beta\gamma_{2^n}} \end{pmatrix}$$

Background and Setup: Measurements

Single-copy measurement (POVM)

A measurement on one copy of a quantum state ρ is described by a collection of positive semidefinite operators

$$\{M_y\}_{y \in [N]}, \quad M_y \succeq 0, \quad \sum_{y \in [N]} M_y = I.$$

Background and Setup: Measurements

Single-copy measurement (POVM)

A measurement on one copy of a quantum state ρ is described by a collection of positive semidefinite operators

$$\{M_y\}_{y \in [N]}, \quad M_y \succeq 0, \quad \sum_{y \in [N]} M_y = I.$$

Outcome probabilities

When we measure ρ , the probability of seeing outcome y is

$$\Pr[Y = y] = \text{Tr}(M_y \cdot \rho).$$

Thus a POVM converts a quantum state into a classical random variable.

Background and Setup: Measurements

Estimating Local Observables (Blackbox)

For any k -local matrix $P = X_1 \otimes X_2 \otimes \cdots \otimes X_k \otimes I$, where each X_i is a Pauli matrix, we can estimate $\text{Tr}(P \cdot \rho)$ up to error ϵ with probability at least $1 - \delta$ using

$$O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$$

copies of ρ .

- Proved using POVM and Chernoff–Hoeffding.

Background and Setup: Measurements

Estimating Local Observables (Blackbox)

For any k -local matrix $P = X_1 \otimes X_2 \otimes \cdots \otimes X_k \otimes I$, where each X_i is a Pauli matrix, we can estimate $\text{Tr}(P \cdot \rho)$ up to error ϵ with probability at least $1 - \delta$ using

$$O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$$

copies of ρ .

- Proved using POVM and Chernoff–Hoeffding.
- We assume access to $\text{Tr}(P \cdot \rho)$ for any such P .

Background and Setup: Measurements

Estimating Local Observables (Blackbox)

For any k -local matrix $P = X_1 \otimes X_2 \otimes \cdots \otimes X_k \otimes I$, where each X_i is a Pauli matrix, we can estimate $\text{Tr}(P \cdot \rho)$ up to error ϵ with probability at least $1 - \delta$ using

$$O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$$

copies of ρ .

- Proved using POVM and Chernoff–Hoeffding.
- We assume access to $\text{Tr}(P \cdot \rho)$ for any such P .
- Can batch estimates for all k -local Paulis via *classical shadows* (details omitted).

Hamiltonian Learning

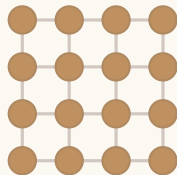
Input:

- Copies of the Gibbs state

$$\rho = \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})}$$

of an unknown local Hamiltonian $H = \sum_e \lambda_e H_e$.

- Accuracy parameter $0 < \epsilon < 1$.



Hamiltonian Learning

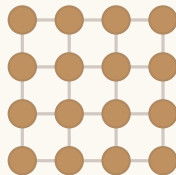
Input:

- Copies of the Gibbs state

$$\rho = \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})}$$

of an unknown local Hamiltonian $H = \sum_e \lambda_e H_e$.

- Accuracy parameter $0 < \epsilon < 1$.



Goal:

Output estimates $\hat{\lambda}_e$ such that $|\lambda_e - \hat{\lambda}_e| \leq \epsilon$ for all e .

Hamiltonian Learning: Motivation

- Local interactions can give rise to exotic macroscopic phenomena (e.g. superconductivity, superfluidity, topological order).
 - High-temperature Gibbs states are unentangled [**B**, Liu, Moitra, Tang '25]
 - Exotic phenomena arise at low temperatures

Hamiltonian Learning: Motivation

- Local interactions can give rise to exotic macroscopic phenomena (e.g. superconductivity, superfluidity, topological order).
 - High-temperature Gibbs states are unentangled [**B**, Liu, Moitra, Tang '25]
 - Exotic phenomena arise at low temperatures
- Hamiltonian learning can be used to:
 - reverse-engineer materials displaying exotic phenomena

Hamiltonian Learning: Motivation

- Local interactions can give rise to exotic macroscopic phenomena (e.g. superconductivity, superfluidity, topological order).
 - High-temperature Gibbs states are unentangled [**B**, Liu, Moitra, Tang '25]
 - Exotic phenomena arise at low temperatures
- Hamiltonian learning can be used to:
 - reverse-engineer materials displaying exotic phenomena
 - verify quantum devices (e.g. quantum simulators and sensors)

Hamiltonian Learning: Motivation

- Local interactions can give rise to exotic macroscopic phenomena (e.g. superconductivity, superfluidity, topological order).
 - High-temperature Gibbs states are unentangled [**B**, Liu, Moitra, Tang '25]
 - Exotic phenomena arise at low temperatures
- Hamiltonian learning can be used to:
 - reverse-engineer materials displaying exotic phenomena
 - verify quantum devices (e.g. quantum simulators and sensors)
 - characterize errors in controllable quantum systems

Hamiltonian Learning: Central Questions

Sample-Efficient Learning

Is there an algorithm that uses *polynomially many* copies of ρ to learn H ?

Hamiltonian Learning: Central Questions

Sample-Efficient Learning

Is there an algorithm that uses *polynomially many* copies of ρ to learn H ?

Computationally-Efficient Learning

Is there an algorithm that uses polynomially many copies of ρ *and runs in polynomial time* to learn H ?

Sample-Efficient Hamiltonian Learning

Theorem [Anshu, Arunachalam, Kuwahara, Soleimanifar '20]

There exists a sample-efficient algorithm for Hamiltonian learning at *any* temperature.



Anshu



Arunachalam



Kuwahara



Soleimanifar

- **Sample complexity:** $\exp(O(\beta)) \cdot \text{poly}(n, 1/\epsilon)$

Sample-Efficient Hamiltonian Learning

Theorem [Anshu, Arunachalam, Kuwahara, Soleimanifar '20]

There exists a sample-efficient algorithm for Hamiltonian learning at *any* temperature.



Anshu



Arunachalam



Kuwahara



Soleimanifar

- **Sample complexity:** $\exp(O(\beta)) \cdot \text{poly}(n, 1/\epsilon)$
- **Runtime:** Computationally intractable in general

Sample-Efficient Hamiltonian Learning

Theorem [Anshu, Arunachalam, Kuwahara, Soleimanifar '20]

There exists a sample-efficient algorithm for Hamiltonian learning at *any* temperature.



Anshu



Arunachalam



Kuwahara



Soleimanifar

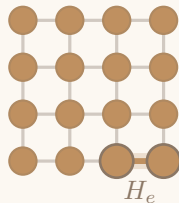
- **Sample complexity:** $\exp(O(\beta)) \cdot \text{poly}(n, 1/\epsilon)$
- **Runtime:** Computationally intractable in general
- **Key idea:** Match the *local marginals* of the Gibbs state

Sufficient Statistics

Key Identifiability Fact

Let $H = \sum_e \lambda_e H_e$ and $H' = \sum_e \lambda'_e H_e$ with Gibbs states ρ and ρ' .

$$\text{Tr}(H_e \cdot \rho) = \text{Tr}(H_e \cdot \rho') \text{ for all } e \implies \lambda_e = \lambda'_e \text{ for all } e.$$



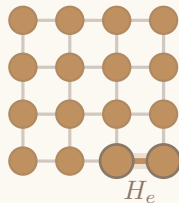
- The quantities $\{\text{Tr}(H_e \cdot \rho)\}_e$ are *sufficient statistics*: observable quantities that uniquely identify the Hamiltonian parameters.

Sufficient Statistics

Key Identifiability Fact

Let $H = \sum_e \lambda_e H_e$ and $H' = \sum_e \lambda'_e H_e$ with Gibbs states ρ and ρ' .

$$\mathrm{Tr}(H_e \cdot \rho) = \mathrm{Tr}(H_e \cdot \rho') \text{ for all } e \implies \lambda_e = \lambda'_e \text{ for all } e.$$



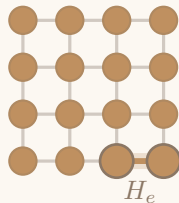
- The quantities $\{\mathrm{Tr}(H_e \cdot \rho)\}_e$ are *sufficient statistics*: observable quantities that uniquely identify the Hamiltonian parameters.
- Computational hardness arises from inverting the map $\{\lambda_e\} \mapsto \{\mathrm{Tr}(H_e \cdot \rho)\}$.

Sufficient Statistics

Key Identifiability Fact

Let $H = \sum_e \lambda_e H_e$ and $H' = \sum_e \lambda'_e H_e$ with Gibbs states ρ and ρ' .

$$\mathrm{Tr}(H_e \cdot \rho) = \mathrm{Tr}(H_e \cdot \rho') \text{ for all } e \implies \lambda_e = \lambda'_e \text{ for all } e.$$



- The quantities $\{\mathrm{Tr}(H_e \cdot \rho)\}_e$ are *sufficient statistics*: observable quantities that uniquely identify the Hamiltonian parameters.
- Computational hardness arises from inverting the map $\{\lambda_e\} \mapsto \{\mathrm{Tr}(H_e \cdot \rho)\}$.
- Corresponds to estimating the partition function $\mathrm{Tr}(e^{-\beta H})$, which is computationally hard at low temperature.

Sufficient Statistics: Proof Sketch

- Given $H = \sum_e \lambda_e H_e$ let $m(\lambda)$ be the vector of local expectations:

$$m(\lambda) = (\text{Tr}(H_{e_1}\rho), \text{Tr}(H_{e_2}\rho), \dots, \text{Tr}(H_{e_m}\rho))$$

Sufficient Statistics: Proof Sketch

- Given $H = \sum_e \lambda_e H_e$ let $m(\lambda)$ be the vector of local expectations:

$$m(\lambda) = (\text{Tr}(H_{e_1}\rho), \text{Tr}(H_{e_2}\rho), \dots, \text{Tr}(H_{e_m}\rho))$$

- Consider the log-partition function:

$$\Phi(\lambda) = \log Z(\lambda) = \log \text{Tr}(e^{-\beta H}).$$

Sufficient Statistics: Proof Sketch

- Given $H = \sum_e \lambda_e H_e$ let $m(\lambda)$ be the vector of local expectations:

$$m(\lambda) = (\text{Tr}(H_{e_1}\rho), \text{Tr}(H_{e_2}\rho), \dots, \text{Tr}(H_{e_m}\rho))$$

- Consider the log-partition function:

$$\Phi(\lambda) = \log Z(\lambda) = \log \text{Tr}(e^{-\beta H}).$$

Key Facts [AAKS '20]

- $\Phi(\lambda)$ is a *strongly convex* function of λ .

Sufficient Statistics: Proof Sketch

- Given $H = \sum_e \lambda_e H_e$ let $m(\lambda)$ be the vector of local expectations:

$$m(\lambda) = (\text{Tr}(H_{e_1}\rho), \text{Tr}(H_{e_2}\rho), \dots, \text{Tr}(H_{e_m}\rho))$$

- Consider the log-partition function:

$$\Phi(\lambda) = \log Z(\lambda) = \log \text{Tr}(e^{-\beta H}).$$

Key Facts [AAKS '20]

- $\Phi(\lambda)$ is a *strongly convex* function of λ .
- Gradient is the vector of local expectations:

$$\nabla \Phi(\lambda) = -\beta \cdot m(\lambda).$$

Sufficient Statistics: Proof Sketch

Strong convexity means:

- Φ is a uniformly curved bowl

Sufficient Statistics: Proof Sketch

Strong convexity means:

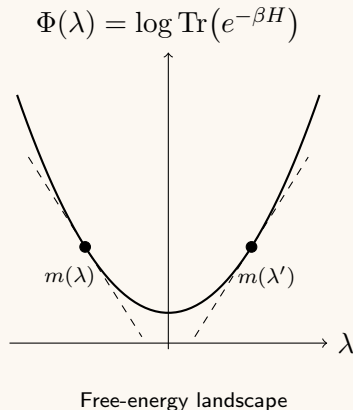
- Φ is a uniformly curved bowl
- Different points must have different slopes

Sufficient Statistics: Proof Sketch

Strong convexity means:

- Φ is a **uniformly curved bowl**
- Different points must have different slopes
- Equivalently, if $m(\lambda)$ is the vector of local expectations for H and $m(\lambda')$ is the vector of local expectations for H' ,

$$m(\lambda) = m(\lambda') \implies \lambda = \lambda'.$$

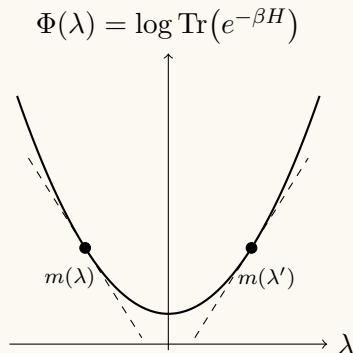


Sufficient Statistics: Proof Sketch

Strong convexity means:

- Φ is a **uniformly curved bowl**
- Different points must have different slopes
- Equivalently, if $m(\lambda)$ is the vector of local expectations for H and $m(\lambda')$ is the vector of local expectations for H' ,

$$m(\lambda) = m(\lambda') \implies \lambda = \lambda'.$$



Free-energy landscape

- [AAKS '20] prove a robust version of this statement to obtain identifiability

Computationally Efficient Algorithms

Efficient Algorithms at High Temperature

Theorem [Haah, Kothari, Tang '22]

When $\beta \leq \beta_c < 1$, there exists a polynomial-time algorithm for Hamiltonian learning.



Haah



Kothari



Tang

- The partition function $\Phi(\lambda) = \log \text{Tr}(e^{-\beta H})$ can be efficiently computed at high temperature via *cluster expansion*.

Efficient Algorithms at High Temperature

Theorem [Haah, Kothari, Tang '22]

When $\beta \leq \beta_c < 1$, there exists a polynomial-time algorithm for Hamiltonian learning.



Haah



Kothari



Tang

- The partition function $\Phi(\lambda) = \log \text{Tr}(e^{-\beta H})$ can be efficiently computed at high temperature via *cluster expansion*.
- The map $\lambda \mapsto m(\lambda)$ can be efficiently inverted via convex optimization.

Efficient Algorithms at High Temperature

Theorem [Haah, Kothari, Tang '22]

When $\beta \leq \beta_c < 1$, there exists a polynomial-time algorithm for Hamiltonian learning.



Haah



Kothari



Tang

- The partition function $\Phi(\lambda) = \log \text{Tr}(e^{-\beta H})$ can be efficiently computed at high temperature via *cluster expansion*.
- The map $\lambda \mapsto m(\lambda)$ can be efficiently inverted via convex optimization.
- At low temperature, cluster expansion breaks down and estimating the partition function is computationally hard.

Obstructions at Low Temperature

Theorem [Montanari '15]

There exist Hamiltonians for which no efficient algorithm can learn H from sufficient statistics alone at low temperature.

- Hardness inherited from the special case where H is diagonal.



Montanari

Obstructions at Low Temperature

Theorem [Montanari '15]

There exist Hamiltonians for which no efficient algorithm can learn H from sufficient statistics alone at low temperature.



Montanari

- Hardness inherited from the special case where H is diagonal.
- Matching sufficient statistics is not enough.

Obstructions at Low Temperature

Theorem [Montanari '15]

There exist Hamiltonians for which no efficient algorithm can learn H from sufficient statistics alone at low temperature.



Montanari

- Hardness inherited from the special case where H is diagonal.
- Matching sufficient statistics is not enough.
- A fundamentally different approach is needed.

Obstructions at Low Temperature

Theorem [Montanari '15]

There exist Hamiltonians for which no efficient algorithm can learn H from sufficient statistics alone at low temperature.



Montanari

- Hardness inherited from the special case where H is diagonal.
- Matching sufficient statistics is not enough.
- A fundamentally different approach is needed.

Hardness Conjecture [Anshu, Arunachalam '23]

Low-temperature Gibbs states may be *computationally indistinguishable* from random states, explaining the difficulty.



Anshu



Arunachalam

Story So Far

- Sample-efficient algorithms for Hamiltonian learning at any temperature

Story So Far

- Sample-efficient algorithms for Hamiltonian learning at any temperature
- Computationally efficient algorithms at high temperature

Story So Far

- Sample-efficient algorithms for Hamiltonian learning at any temperature
- Computationally efficient algorithms at high temperature
- Interesting macroscopic phenomena arises only at low temperature

Story So Far

- Sample-efficient algorithms for Hamiltonian learning at any temperature
- Computationally efficient algorithms at high temperature
- Interesting macroscopic phenomena arises only at low temperature
- Sufficient statistics are not sufficient for efficient algorithms at low temperature

Story So Far

- Sample-efficient algorithms for Hamiltonian learning at any temperature
- Computationally efficient algorithms at high temperature
- Interesting macroscopic phenomena arises only at low temperature
- Sufficient statistics are not sufficient for efficient algorithms at low temperature
- Low-temperature Gibbs states may be computationally indistinguishable from random states

Hamiltonian Learning at any temperature

Theorem [B., Liu, Moitra, Tang '24]

Given copies of the Gibbs state of any local Hamiltonian at any β and $0 < \epsilon < 1$, there exists an algorithm that outputs $\hat{\lambda}_e$ such that

- **Correctness:** $|\lambda_e - \hat{\lambda}_e| \leq \epsilon$ for all e .



Liu



Moitra



Tang

Hamiltonian Learning at any temperature

Theorem [B., Liu, Moitra, Tang '24]

Given copies of the Gibbs state of any local Hamiltonian at any β and $0 < \epsilon < 1$, there exists an algorithm that outputs $\hat{\lambda}_e$ such that

- **Correctness:** $|\lambda_e - \hat{\lambda}_e| \leq \epsilon$ for all e .
- **Sample Complexity:** $s = \text{poly}(m)(1/\epsilon)^{2^{O(\beta)}}$



Liu



Moitra



Tang

Hamiltonian Learning at any temperature

Theorem [B., Liu, Moitra, Tang '24]

Given copies of the Gibbs state of any local Hamiltonian at any β and $0 < \epsilon < 1$, there exists an algorithm that outputs $\hat{\lambda}_e$ such that

- **Correctness:** $|\lambda_e - \hat{\lambda}_e| \leq \epsilon$ for all e .
- **Sample Complexity:** $s = \text{poly}(m)(1/\epsilon)^{2^{O(\beta)}}$
- **Running Time:** $s^{O(1)}$



Liu



Moitra



Tang

Hamiltonian Learning at any temperature

Theorem [B., Liu, Moitra, Tang '24]

Given copies of the Gibbs state of any local Hamiltonian at any β and $0 < \epsilon < 1$, there exists an algorithm that outputs $\hat{\lambda}_e$ such that

- **Correctness:** $|\lambda_e - \hat{\lambda}_e| \leq \epsilon$ for all e .
- **Sample Complexity:** $s = \text{poly}(m)(1/\epsilon)^{2^{O(\beta)}}$
- **Running Time:** $s^{O(1)}$



Liu



Moitra



Tang

Upshot: Hamiltonian Learning is tractable at any temperature!

Recasting Hamiltonian Learning as a constrained optimization problem

Learning via Constrained Optimization

Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot \frac{e^{-\beta(\sum_e \lambda'_e H_e)}}{\text{Tr}(e^{-\beta(\sum_e \lambda'_e H_e)})} \right) = \text{Tr}(H_e \cdot \rho) \end{array} \right\},$$

- Non-convex optimization problem with exponential equalities as constraints

Learning via Constrained Optimization

Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot \frac{e^{-\beta(\sum_e \lambda'_e H_e)}}{\text{Tr}(e^{-\beta(\sum_e \lambda'_e H_e)})} \right) = \text{Tr}(H_e \cdot \rho) \end{array} \right\},$$

- Non-convex optimization problem with exponential equalities as constraints
- The RHS can be estimated from measurements

Learning via Constrained Optimization

Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot \frac{e^{-\beta(\sum_e \lambda'_e H_e)}}{\text{Tr}(e^{-\beta(\sum_e \lambda'_e H_e)})} \right) = \text{Tr}(H_e \cdot \rho) \end{array} \right\},$$

- Non-convex optimization problem with exponential equalities as constraints
- The RHS can be estimated from measurements
- Remains computationally intractable

Learning via Constrained Optimization

Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot \frac{e^{-\beta(\sum_e \lambda'_e H_e)}}{\text{Tr}(e^{-\beta(\sum_e \lambda'_e H_e)})} \right) = \text{Tr}(H_e \cdot \rho) \end{array} \right\},$$

- Non-convex optimization problem with exponential equalities as constraints
- The RHS can be estimated from measurements
- Remains computationally intractable
- [AAKS '20] prove that this constraint system is *feasible* and has a unique solution

Learning via Constrained Optimization

Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot \frac{e^{-\beta(\sum_e \lambda'_e H_e)}}{\text{Tr}(e^{-\beta(\sum_e \lambda'_e H_e)})} \right) = \text{Tr}(H_e \cdot \rho) \end{array} \right\},$$

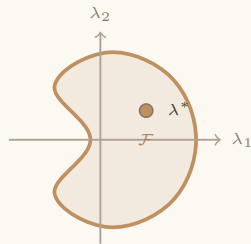
- Non-convex optimization problem with exponential equalities as constraints
- The RHS can be estimated from measurements
- Remains computationally intractable
- [AAKS '20] prove that this constraint system is *feasible* and has a unique solution
- How do we modify this optimization problem to make it tractable?

Proofs-to-Algorithms Paradigm

Proofs-to-Algorithms Paradigm

- Consider a structured system of *polynomial inequalities*

$$\left\{ \begin{array}{l} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 \geq 0 \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 \geq 0 \\ \vdots \end{array} \right.$$

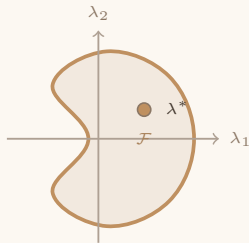


Proofs-to-Algorithms Paradigm

- Consider a structured system of *polynomial inequalities*

$$\left\{ \begin{array}{l} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 \geq 0 \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 \geq 0 \\ \vdots \end{array} \right\}$$

- Show that the system is *feasible* i.e. the true parameters satisfy the constraints

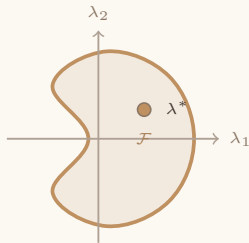


Proofs-to-Algorithms Paradigm

- Consider a structured system of *polynomial inequalities*

$$\left\{ \begin{array}{l} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 \geq 0 \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 \geq 0 \\ \vdots \end{array} \right.$$

- Show that the system is *feasible* i.e. the true parameters satisfy the constraints
- Show that the system of polynomial inequalities *identifies* the true parameters



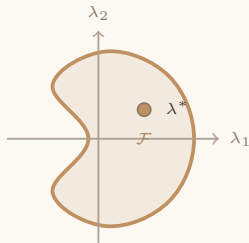
Proofs-to-Algorithms Paradigm

- Consider a structured system of *polynomial inequalities*

$$\left\{ \begin{array}{l} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 \geq 0 \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 \geq 0 \\ \vdots \end{array} \right.$$

- Show that the system is *feasible* i.e. the true parameters satisfy the constraints
- Show that the system of polynomial inequalities *identifies* the true parameters

Surprising Consequence: There is an efficient algorithm that finds the true parameters



Proofs-to-Algorithms Paradigm

- Consider a system of *sum-of-squares inequalities* of degree at most d :

$$\mathcal{A} = \left\{ \begin{array}{lcl} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 & = & \sum_j p_j^2(\lambda') \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 & = & \sum_j q_j^2(\lambda') \\ \vdots & & \end{array} \right\}$$

Proofs-to-Algorithms Paradigm

- Consider a system of *sum-of-squares inequalities* of degree at most d :

$$\mathcal{A} = \left\{ \begin{array}{lcl} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 & = & \sum_j p_j^2(\lambda') \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 & = & \sum_j q_j^2(\lambda') \\ \vdots & & \end{array} \right\}$$

- Feasibility:** Show that λ satisfies the constraint system $\mathcal{A}$

Proofs-to-Algorithms Paradigm

- Consider a system of *sum-of-squares inequalities* of degree at most d :

$$\mathcal{A} = \left\{ \begin{array}{lcl} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 & = & \sum_j p_j^2(\lambda') \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 & = & \sum_j q_j^2(\lambda') \\ \vdots & & \end{array} \right\}$$

- Feasibility:** Show that λ satisfies the constraint system $\mathcal{A}$
- Identifiability:** Show that any λ' that satisfies the constraint system $\mathcal{A}$ must satisfy $\lambda' = \lambda$ (this proof needs to only use low-degree SoS inequalities)

Proofs-to-Algorithms Paradigm

- Consider a system of *sum-of-squares inequalities* of degree at most d :

$$\mathcal{A} = \left\{ \begin{array}{lcl} \lambda_1'^2 + \lambda_2'^4 \lambda_3'^4 + \lambda_4'^2 & = & \sum_j p_j^2(\lambda') \\ \lambda_3'^4 \lambda_2'^4 + \lambda_1'^4 + \lambda_1'^2 \lambda_4'^2 & = & \sum_j q_j^2(\lambda') \\ \vdots & & \end{array} \right\}$$

- Feasibility:** Show that λ satisfies the constraint system $\mathcal{A}$
- Identifiability:** Show that any λ' that satisfies the constraint system $\mathcal{A}$ must satisfy $\lambda' = \lambda$ (this proof needs to only use low-degree SoS inequalities)

Surprising Consequence: There is algorithm that runs in time $n^{O(d)}$ and finds the true parameters λ .

Learning via Optimization

Vanilla Optimization Problem

Find $\lambda'_1, \dots, \lambda'_m$ such that

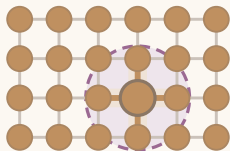
$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall e \in [m], \quad \text{Tr} \left(H_e \cdot e^{-\beta H'} / \text{Tr} \left(e^{-\beta H'} \right) \right) = \text{Tr} (H_e \cdot \rho) \end{array} \right\},$$

Learning via Optimization

Modified Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) = \text{Tr} (P Q \rho) \end{array} \right\},$$



Learning via Optimization

Modified Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) = \text{Tr} (P Q \rho) \end{array} \right\},$$

- Avoids the sufficient statistics hardness by estimating a larger set of local observables.
- True parameters remain feasible since

$$\text{Tr}(Q e^{-\beta H} P e^{\beta H} \rho) = \text{Tr} \left(Q e^{-\beta H} P e^{\beta H} \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} \right) = \text{Tr}(P Q \rho)$$

Learning via Optimization

Modified Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) = \text{Tr} (P Q \rho) \end{array} \right\},$$

- Avoids the sufficient statistics hardness by estimating a larger set of local observables.
- True parameters remain feasible.
- Constraints are still exponential equalities — remain computationally intractable.

Low-Degree Polynomial Approximation

Key Lemma

There exists a degree- $(2^{O(\beta)} + \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$

- The polynomial $p(H', P) = I + c_1[H', P] + c_2[H'[H', P]] + \dots + c_d[H' \dots [H', P]]$

Low-Degree Polynomial Approximation

Key Lemma

There exists a degree- $(2^{O(\beta)} + \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$

- The polynomial $p(H', P) = I + c_1[H', P] + c_2[H'[H', P]] + \dots + c_d[H' \dots [H', P]]$
- The approximation guarantee admits a sum-of-squares proof

Low-Degree Polynomial Approximation

Key Lemma

There exists a degree- $(2^{O(\beta)} + \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$

- The polynomial $p(H', P) = I + c_1[H', P] + c_2[H'[H', P]] + \dots + c_d[H' \dots [H', P]]$
- The approximation guarantee admits a sum-of-squares proof
- We can write down a system of sum-of-squares inequalities that capture Hamiltonian learning

Tractable Optimization via Sum-of-Squares

Tractable Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr}(Q p(H', P) \rho) = \text{Tr}(P Q \rho) \pm \epsilon \end{array} \right\},$$

- **Feasibility:** Inherited feasibility just from the fact that the polynomial approximation is good

Tractable Optimization via Sum-of-Squares

Tractable Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr}(Q p(H', P) \rho) = \text{Tr}(P Q \rho) \pm \epsilon \end{array} \right\},$$

- **Feasibility:** Inherited feasibility just from the fact that the polynomial approximation is good
- **Identifiability:** Major technical heavy-lifting in the paper

Tractable Optimization via Sum-of-Squares

Tractable Optimization Problem

Let $\mathcal{P}_k$ be the set of k -local Pauli operators ($k = O(1)$). Find $\lambda'_1, \dots, \lambda'_m$ such that

$$\left\{ \begin{array}{l} \forall e \in [m] \quad -1 \leq \lambda'_e \leq 1 \\ H' = \sum_e \lambda'_e H_e \\ \forall P, Q \in \mathcal{P}_k, \quad \text{Tr}(Q p(H', P) \rho) = \text{Tr}(P Q \rho) \pm \epsilon \end{array} \right\},$$

- **Feasibility:** Inherited feasibility just from the fact that the polynomial approximation is good
- **Identifiability:** Major technical heavy-lifting in the paper
- **Running time:** System can be solved in $\text{poly}(n) \cdot (1/\epsilon)^{2^{2^{O(\beta)}}}$ time

Theorem [Narayanan '25]

There exists a degree- $(\beta^2 \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$

- Improves running time to $\text{poly}(n) \cdot (1/\epsilon)^{\beta^2}$



Narayanan

Recent Progress

Theorem [Narayanan '25]

There exists a degree- $(\beta^2 \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$



Narayanan

- Improves running time to $\text{poly}(n) \cdot (1/\epsilon)^{\beta^2}$
- Exponential dependence on β is necessary: 2^β sample complexity lower bound by [Bresler'15]

Theorem [Narayanan '25]

There exists a degree- $(\beta^2 \log(1/\epsilon))$ degree polynomial p such that

$$\left| \text{Tr} \left(Q e^{-\beta H'} P e^{\beta H'} \rho \right) - \text{Tr} \left(Q p(H', P) \rho \right) \right| \leq \epsilon$$



Narayanan

- Improves running time to $\text{poly}(n) \cdot (1/\epsilon)^{\beta^2}$
- Exponential dependence on β is necessary: 2^β sample complexity lower bound by [Bresler'15]

Open Question: Does there exist a $\text{poly}(n, 1/\epsilon) \cdot 2^{O(\beta)}$ time algorithm for Hamiltonian learning?

Recent Progress

Theorem [Chen, Anshu, Nguyen '25]

For Hamiltonians on a lattice, there exists a Hamiltonian learning algorithm such that

- Sample complexity: $\log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$
- Running Time: $n \log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$
- Sample complexity is optimal in n and ϵ



Chen



Anshu



Nguyen

Recent Progress

Theorem [Chen, Anshu, Nguyen '25]

For Hamiltonians on a lattice, there exists a Hamiltonian learning algorithm such that

- Sample complexity: $\log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$
 - Running Time: $n \log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$
-
- Sample complexity is optimal in n and ϵ
 - For graphs with bounded degree, running time degrades



Chen



Anshu



Nguyen

Recent Progress

Theorem [Chen, Anshu, Nguyen '25]

For Hamiltonians on a lattice, there exists a Hamiltonian learning algorithm such that

- Sample complexity: $\log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$
- Running Time: $n \log(n)/\epsilon^2 \cdot \exp(\text{poly}(\beta))$

- Sample complexity is optimal in n and ϵ
- For graphs with bounded degree, running time degrades

Open Question: Is there an efficient learning algorithm that requires only $2^{O(\beta)} \cdot \log(n)/\epsilon^2$ samples?



Chen



Anshu



Nguyen

Recent Progress

Consider the “Proper Learning” variant: Output some state $\hat{\rho}$ such that

$$\left\| \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} - \hat{\rho} \right\|_{\text{tr}} \leq \epsilon$$

Recent Progress

Consider the “Proper Learning” variant: Output some state $\hat{\rho}$ such that

$$\left\| \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} - \hat{\rho} \right\|_{\text{tr}} \leq \epsilon$$

Theorem [Bluhm, Caro, Gutierrez, Lee, Oufkir, Rouze, Shin '26]

Proper learning at any temperature is possible with:

- **Sample Complexity:** $O\left(\text{poly}(n), \frac{\beta^2}{\epsilon^4}\right)$
- **Running Time:** Exponential in n



Bluhm



Caro



Gutierrez



Lee



Rouze



Shin

Recent Progress

Consider the “Proper Learning” variant: Output some state $\hat{\rho}$ such that

$$\left\| \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} - \hat{\rho} \right\|_{\text{tr}} \leq \epsilon$$

Theorem [Bluhm, Caro, Gutierrez, Lee, Oufkir, Rouze, Shin '26]

Proper learning at any temperature is possible with:

- **Sample Complexity:** $O\left(\text{poly}(n), \frac{\beta^2}{\epsilon^4}\right)$
- **Running Time:** Exponential in n



Bluhm



Caro



Gutierrez



Lee



Rouze



Shin

Open Question: Is there an efficient proper learning algorithm which requires only $\text{poly}(n, \beta, 1/\epsilon)$ samples?

Open Directions: Structure Learning

- All the results discussed so far assume that the structure of the Hamiltonian is known

Open Directions: Structure Learning

- All the results discussed so far assume that the structure of the Hamiltonian is known
- The algorithms only search for the coefficients

Open Directions: Structure Learning

- All the results discussed so far assume that the structure of the Hamiltonian is known
- The algorithms only search for the coefficients
- Promised that the Hamiltonian is k -local, can we learn the structure as well?

Open Directions: Structure Learning

- All the results discussed so far assume that the structure of the Hamiltonian is known
- The algorithms only search for the coefficients
- Promised that the Hamiltonian is k -local, can we learn the structure as well?
- For classical Ising models, [Klivans, Meka'17] give an efficient structure learning algorithm

Open Directions: Structure Learning

- All the results discussed so far assume that the structure of the Hamiltonian is known
- The algorithms only search for the coefficients
- Promised that the Hamiltonian is k -local, can we learn the structure as well?
- For classical Ising models, [Klivans, Meka'17] give an efficient structure learning algorithm
- If there is an obstruction it must be something entirely quantum

Open Directions: Agnostic Tomography

Agnostic Tomography for Gibbs states

Given copies of an unknown state ψ , find the closest Gibbs state

$$\rho = \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} \text{ where } H \text{ is } k\text{-local}$$

- Minimize trace distance
- Maximize fidelity

Open Directions: Agnostic Tomography

Agnostic Tomography for Gibbs states

Given copies of an unknown state ψ , find the closest Gibbs state

$$\rho = \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} \text{ where } H \text{ is } k\text{-local}$$

- Minimize trace distance
- Maximize fidelity

Agnostic tomography is possible for simple ensembles of states such as

Open Directions: Agnostic Tomography

Agnostic Tomography for Gibbs states

Given copies of an unknown state ψ , find the closest Gibbs state

$$\rho = \frac{e^{-\beta H}}{\text{Tr}(e^{-\beta H})} \text{ where } H \text{ is } k\text{-local}$$

- Minimize trace distance
- Maximize fidelity

Agnostic tomography is possible for simple ensembles of states such as

- Stabilizer Product States [Grewal, Iyer, Kretschmer, Liang '24]
- Stabilizer States [Chen, Gong, Ye, Zhang '25]
- Product States and Matrix Product States
[B., Bostanci, Kretschmer, Landau, Li, Liu, O'Donnell, Tang]