Annual Review of Statistics and Its Application

Quantum Computing in a Statistical Context

Yazhen Wang and Hongzhi Liu

Department of Statistics, University of Wisconsin–Madison, Madison, Wisconsin 53706, USA;
email: yzwang@stat.wisc.edu

Annu. Rev. Stat. Appl. 2022. 9:479–504

The *Annual Review of Statistics and Its Application* is online at statistics.annualreviews.org

https://doi.org/10.1146/annurev-statistics-042720-024040

Keywords

quantum annealing, quantum Boltzmann machine, quantum computation, quantum computational supremacy, quantum information, quantum statistics

Abstract

Quantum computing is widely considered a frontier of interdisciplinary research and involves fields ranging from computer science to physics and from chemistry to engineering. On the one hand, the stochastic essence of quantum physics results in the random nature of quantum computing; thus, there is an important role for statistics to play in the development of quantum computing. On the other hand, quantum computing has great potential to revolutionize computational statistics and data science. This article provides an overview of the statistical aspect of quantum computing. We review the basic concepts of quantum computing and introduce quantum research topics such as quantum annealing and quantum machine learning, which require statistics to be understood.

1. INTRODUCTION

Quantum information science investigates the preparation and control of the quantum states of physical systems for the purposes of information transmission and manipulation. This field consists of quantum communication, quantum computation, and quantum information. There is a wide belief that quantum information science will likely lead to a new wave of technological innovations in communication, computation, and information (see Wang 2012, Wang et al. 2016, and Wang & Song 2020 for details). As the crown jewel of quantum information science, quantum computing is gaining growing interest and tremendous attention in fields ranging from computer science to physics and from chemistry to engineering. Theoretically, it has been shown that quantum computational algorithms can be much faster than the best or optimal classical algorithms for solving certain tough computational problems. Experimentally, the Google Quantum AI group (where AI stands for artificial intelligence) designed a hard sampling problem for its recently created quantum computer and successfully performed sampling computations in a computational space of dimension $2^{53} \approx 10^{16}$, which is practically beyond the reach of the fastest classical supercomputers available at present (see Section 4.1, Arute et al. 2019, and Zhong et al. 2020 for details). The media has frequently reported that a calculation that would take a quantum computer 3 minutes and 20 seconds would take the most powerful supercomputer in the world 10,000 years. This is an example of a concept often termed quantum (computational) supremacy—a demonstration that quantum computers can surpass classical ones—and requires a combination of hardware construction, software design, and problem creation and implementation.

As quantum computers of large scale are currently not available to implement faster quantum algorithms for accomplishing difficult computational tasks like breaking cryptosystems that are secure against any classical computer–based attack, it is important to demonstrate quantum supremacy and provide experimental evidence to support the (theoretical) claim that quantum computation has advantages over classical computation. Since quantum physics is essentially stochastic, quantum computation is random in nature. Consequently, statistics can play an important role in quantum computation, which offers, in turn, great potential for computational statistics and data science. As our goal in this article is to provide an overview of the statistical aspect of quantum computation, we introduce the basic concepts of quantum computation and present some selected relevant topics in quantum computing that encounter many statistical issues. Throughout the overview, we illustrate the interplay between statistics and quantum computation. In particular, our focus is on the application of new quantum resources to accomplish statistical computational tasks that are either very slow or infeasible by classical techniques, and the use of quantum approaches that may lead to new theories, methodologies, and computational techniques for statistics and machine learning. We refer readers to Wang (2012) and Wang & Song (2020) for quantum cryptography topics such as quantum code-breaking algorithms and quantum crypto devices.

The rest of the article proceeds as follows. Section 2 briefly introduces quantum mechanics and quantum probability and statistics. Section 3 reviews basic concepts in quantum computation and different architectures for quantum computation. Section 4 features two landmark projects on quantum computational supremacy that involve boson sampling and random quantum circuits. Section 5 illustrates quantum annealing and related statistical analysis. Section 6 presents quantum deep learning and describes both classical and quantum approaches with Boltzmann machines (BMs). Section 7 provides concluding remarks.

2. QUANTUM BACKGROUND REVIEW

2.1. Mathematical Concepts and Notations

Denote by $\mathbb{N}$ and $\mathbb{R}$ the sets of all integers and real numbers, respectively. We introduce finite-dimensional linear algebra and metric spaces. Denote by $\mathbb{C}$ the set of all complex numbers and by $\mathbb{C}^k$ the vector space consisting of the set of all k-tuples of complex numbers $(z_1, \ldots, z_k)$. In quantum mechanics and quantum computation, we utilize Dirac notations $|\cdot\rangle$ (which is called ket) and $\langle\cdot|$ (which is called bra) to specify that the objects are column vectors or row vectors in the vector space, respectively. We use superscripts *, $'$, and † to denote the conjugate of a complex number, the transpose of a vector or matrix, and the conjugate transpose operation, respectively. We denote by $\langle u|v\rangle$ the inner product of vectors $|u\rangle$ and $|v\rangle$ and adopt a natural inner product for $\mathbb{C}^k$: $\langle u|v\rangle = \sum_{j=1}^k u_j^* v_j = (u_1^*, \ldots, u_k^*)(v_1, \ldots, v_k)'$, where $\langle u| = (u_1, \ldots, u_k)$ and $|v\rangle = (v_1, \ldots, v_k)'$. The inner product induces a norm $\|u\| = \sqrt{\langle u|u\rangle}$ and a distance $\|u - v\|$ between $|u\rangle$ and $|v\rangle$. We say that $\mathcal{H}$ is a finite-dimensional Hilbert space if it is a vector space with an inner product.

We call a matrix $\mathbf{A}$ Hermitian (or self-adjoint) if $\mathbf{A} = \mathbf{A}^\dagger$, and a matrix $\mathbf{U}$ unitary if $\mathbf{U}\mathbf{U}^\dagger = \mathbf{U}^\dagger\mathbf{U} = \mathbf{I}$, where $\mathbf{I}$ denotes the identity matrix. A matrix $\mathbf{A}$ is called semipositive (or positive) definite if $\langle u|\mathbf{A}|u\rangle \geq 0$ for all $|u\rangle \in \mathcal{H}$ (or $\langle u|\mathbf{A}|u\rangle \geq 0$ for all $|u\rangle \in \mathcal{H}$ with equality only for $|u\rangle = 0$). For the matrix $\mathbf{A} = (a_{j\ell})$, denote its trace by $\mathrm{tr}(\mathbf{A}) = \sum_{j=1}^k a_{jj}$. We use $\otimes$ to denote the tensor product operation of vectors or matrices.

2.2. Quantum Physics

A quantum system is characterized by its state and the dynamic evolution of the state. We describe a quantum state by a unit complex vector and its dynamic time evolution by a unitary evolution, where the unitary evolution means that the quantum state changes over time, and these changes are linked by unitary matrices. Furthermore, the dynamic time evolution of the quantum state is governed by a differentiation equation called the Schrödinger equation. To be specific, let $|\psi(t)\rangle$ be the state of the quantum system at time t. The states $|\psi(t)\rangle$ and $|\psi(t+s)\rangle$ at times t and $t+s$, respectively, are linked by $|\psi(t+s)\rangle = \mathbf{U}(s)\,|\psi(t)\rangle$, where $\mathbf{U}(s) = \exp[-\sqrt{-1}\mathbf{H}s]$ is a unitary matrix, and $\mathbf{H}$ is a Hermitian matrix on $\mathbb{C}^d$, which is known as the Hamiltonian of the quantum system. The Schrödinger equation governs the continuous-time evolution of $|\psi(t)\rangle$ as follows: $\sqrt{-1}\frac{\partial|\psi(t)\rangle}{\partial t} = \mathbf{H}|\psi(t)\rangle$. Alternatively, we may depict a quantum system by a so-called density matrix. The quantum state of a d-dimensional quantum system can be described by a density matrix $\boldsymbol{\rho}$ on the d-dimensional complex space $\mathbb{C}^d$, where $\boldsymbol{\rho}$ is a positive semidefinite Hermitian matrix with unit trace. We often classify a quantum state as a pure state or an ensemble of pure states. A pure state is a unit vector $|\psi\rangle$ in $\mathbb{C}^d$ with a corresponding density matrix $\boldsymbol{\rho} = |\psi\rangle\langle\psi|$. An ensemble of pure states has a density matrix $\boldsymbol{\rho} = \sum_{j=1}^J p_j\,|\psi_j\rangle\langle\psi_j|$, which corresponds to the scenario that the quantum system is in one of the states $|\psi_j\rangle, j = 1, \ldots, J$, with probability p_j being in the state $|\psi_j\rangle$. We may describe the quantum evolution in the density matrix representation as follows: $\boldsymbol{\rho}_{t+s} = \mathbf{U}(t)\boldsymbol{\rho}_s\mathbf{U}^\dagger(t)$, where $\boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_{t+s}$ stand for the density matrix of the state of the quantum system at times s and $t+s$, respectively, and the unitary matrix $\mathbf{U}(s)$ is introduced above (for details, see Shankar 2012, Sakurai & Napolitano 2017).

2.3. Quantum Probability

Measurements on quantum systems are often through observables like position and momentum, where an observable $\mathbf{M}$ is defined as a Hermitian matrix on $\mathbb{C}^d$. Assume the following eigen-decomposition for $\mathbf{M}$:

$$\mathbf{M} = \sum_{a=1}^{d} x_a \mathbf{Q}_a, \tag{1}$$

where x_a are the real eigenvalues of $\mathbf{M}$, and $\mathbf{Q}_a$ are the corresponding projections onto the eigen-spaces of $\mathbf{M}$. Performing a measurement on $\mathbf{M}$ for a quantum system prepared in state $\boldsymbol{\rho}$, we adopt a measure space $(\Omega, \mathcal{F})$ to describe its possible measurement outcomes and treat the measurement result X as a random variable on $(\Omega, \mathcal{F})$ with probability distribution $\mathrm{P}_{\boldsymbol{\rho}}$ as follows. The random variable X takes values in $\{x_1, x_2, \ldots, x_d\}$, and the probability of obtaining measurement outcome x_a is given by

$$\mathrm{P}_{\boldsymbol{\rho}}(X = x_a) = \mathrm{tr}(\mathbf{Q}_a \boldsymbol{\rho}), \quad a = 1, 2, \ldots, d.$$

With the probability defined, we derive its expectation and variance,

$$\mathrm{E}_{\boldsymbol{\rho}}(X) = \mathrm{tr}(\mathbf{M}\boldsymbol{\rho}), \quad \mathrm{Var}(X) = \mathrm{tr}(\mathbf{M}^2 \boldsymbol{\rho}) - [\mathrm{tr}(\mathbf{M}\boldsymbol{\rho})]^2.$$

A quantum probability is a noncommutative analog of the Kolmogorov probability. We give a simple illustration using the finite case considered above. The quantum counterparts of sample space and sigma-field are $\mathcal{H} = \mathbb{C}^d$ and an algebra $\mathcal{A}$ formed by subspaces of $\mathcal{H}$, respectively. Quantum events are subspaces of $\mathcal{H}$ like eigen-spaces of $\mathbf{M}$, and observables are a quantum analog of random variables. We present a simple quantum probability $\mathbb{P}$ on $(\mathcal{H}, \mathcal{A})$ as follows. For a given subspace $\mathbf{Q}$ of $\mathbb{C}^d$, define $\mathbb{P}(\mathbf{Q}) = \mathrm{tr}(\mathbf{Q}\boldsymbol{\rho})$. Then there is a corresponding distributional relationship between $\mathbf{M}$ under $\mathbb{P}$ and random variable X under $\mathrm{P}_{\boldsymbol{\rho}}$. In fact, identifying the projection matrix $\mathbf{Q}_a$ with its corresponding eigen-space, we have $\mathbb{P}(\mathbf{Q}_a) = \mathrm{P}_{\boldsymbol{\rho}}(X = a)$ with the quantum expectation of observable $\mathbf{M}$,

$$\mathbb{E}(\mathbf{M}) = \sum_{a=1}^{d} a\, \mathbb{E}(\mathbf{Q}_a) = \sum_{a=1}^{d} a\, \mathbb{P}(\mathbf{Q}_a) = \sum_{a=1}^{d} a\, \mathrm{P}_{\boldsymbol{\rho}}(X = a) = \mathrm{E}_{\boldsymbol{\rho}}(X).$$

However, the defined quantum probability $(\mathcal{H}, \mathcal{A}, \mathbb{P})$ is noncommutative. For example, consider the case of $d = 2$ and two noncommutable observables $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ and $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Although each observable corresponds to a Bernoulli random variable, there is no quantum analog of the joint distribution for these two observables, which is related to Heisenberg's uncertainty principle and the fact that performing measurements on a quantum system changes its state and thus the resulting probability. The general definition of quantum probability takes $\mathcal{H}$ as a complex Hilbert space, $\mathcal{A}$ as a C^*-algebra on $\mathcal{H}$, and $\mathbb{P}$ as a noncommutative probability on $(\mathcal{H}, \mathcal{A})$. For details, readers are directed to Holevo (2001), Parthasarathy (2012), and Wang (2012).

2.4. Quantum Statistics

Statistics is heavily used in quantum theory and quantum experiments, particularly quantum computation. For a quantum system, we may make statistical inference about the measurement distribution $\mathrm{P}_{\boldsymbol{\rho}}$ and thus indirectly about the quantum state $\boldsymbol{\rho}$ based on measurements $X_1, \ldots, X_n$ obtained from measuring some observable for the quantum system. That is, $X_1, \ldots, X_n$ are independent and identically distributed observations with distribution $\mathrm{P}_{\boldsymbol{\rho}}$, and we can infer $\boldsymbol{\rho}$ based on $X_1, \ldots, X_n$. Assume that $\boldsymbol{\rho}$ is known up to some unknown parameter $\boldsymbol{\theta}$. Then we use $\mathrm{P}_{\boldsymbol{\rho}}$ to specify a quantum parametric model and draw statistical inference about $\boldsymbol{\theta}$. We may define quantum likelihood and quantum Fisher information and establish quantum statistical theory such as the quantum Cramér-Rao bound and asymptotic efficient estimation. Specifically, denote by $\boldsymbol{\rho}_\theta$ a parametric quantum density matrix family indexed by parameter $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)'$. Define its quantum score matrices $\ell_j(\boldsymbol{\theta}), j = 1, \ldots, p$, to be symmetrized logarithmic derivatives of $\boldsymbol{\rho}_{\boldsymbol{\theta}}$ with respect to $\boldsymbol{\theta}$—that is, $\ell_j(\boldsymbol{\theta})$ are Hermitian matrices satisfying

$$\frac{\partial \boldsymbol{\rho}_{\boldsymbol{\theta}}}{\partial \theta_j} = \frac{1}{2}\left[\ell_j(\boldsymbol{\theta})\boldsymbol{\rho}_{\boldsymbol{\theta}} + \boldsymbol{\rho}_{\boldsymbol{\theta}}\ell_j(\boldsymbol{\theta})\right].$$

The quantum Fisher information matrix is given by $\mathrm{J}(\boldsymbol{\theta}) = (J_{ij}(\boldsymbol{\theta}))_{1\le i,j\le p}$, where

$$J_{ij}(\boldsymbol{\theta}) = \frac{1}{2}\mathrm{tr}\left[\boldsymbol{\rho}_{\boldsymbol{\theta}}\left(\ell_i(\boldsymbol{\theta})\ell_j(\boldsymbol{\theta}) + \ell_j(\boldsymbol{\theta})\ell_i(\boldsymbol{\theta})\right)\right].$$

The quantum Cramér-Rao bound states that for an unbiased estimator of $\boldsymbol{\theta}$, its variance is bounded below by the inverse of $\mathrm{J}(\boldsymbol{\theta})$. Also, we may model $\boldsymbol{\rho}$ nonparametrically and employ nonparametric methods to infer $\boldsymbol{\rho}$. Quantum information science refers to the reconstruction of $\boldsymbol{\rho}$ as quantum state tomography. For example, consider testing the quantum hypothesis that a quantum state is in a given state $\boldsymbol{\rho}_1$ against an alternative state $\boldsymbol{\rho}_2$. We can establish the quantum analog of the classical theory for the trade-off behavior between two types of errors. As a case in point, denote by β_n the type II error of the optimal level α test for the quantum hypotheses based on n identical copies of the quantum system. Then we have the quantum Stein lemma: $\lim_{n\to\infty}[n^{-1}\log\beta_n] = -S(\boldsymbol{\rho}_1|\boldsymbol{\rho}_2)$, where $S(\boldsymbol{\rho}_1|\boldsymbol{\rho}_2) = \mathrm{tr}(\boldsymbol{\rho}_1[\log\boldsymbol{\rho}_1 - \log\boldsymbol{\rho}_2])$ is the quantum relative entropy (or quantum Kullback-Leibler divergence) of $\boldsymbol{\rho}_1$ and $\boldsymbol{\rho}_2$. That is, similar to the classical case, the quantum type II error exponentially decays to zero at a rate determined by the quantum relative entropy. For details, readers are directed to Artiles et al. (2005), Barndorff-Nielsen et al. (2003), Cai et al. (2016), Holevo (2001), Petz (2008), and Wang & Xu (2015).

3. QUANTUM COMPUTATION

3.1. Quantum Bit and Superposition

Bits are the most fundamental concept in classical information and computation science. We encode the information in a bit with two mutually exclusive states, 0 and 1, and may easily realize it by a mechanical switch. The quantum counterpart of the classical bit is the quantum bit, or qubit for short. Similar to the two state values 0 and 1 for the classical bit, a qubit has states $|0\rangle$ and $|1\rangle$, where the customary notation $|\cdot\rangle$ is used to denote the qubit state. However, there is a key difference between a classical bit and a qubit. Besides states $|0\rangle$ and $|1\rangle$, a qubit may be in superposition states,

$$|\psi\rangle = \alpha_0\,|0\rangle + \alpha_1\,|1\rangle,$$

where α_0 and α_1 are complex numbers called amplitudes satisfying $|\alpha_0|^2 + |\alpha_1|^2 = 1$. The states of a qubit consist of unit vectors in $\mathbb{C}^2$, with the states $|0\rangle$ and $|1\rangle$ forming an orthonormal basis, that are called computational basis states. Unlike the mutually exclusive states for classical bits, the superposition states allow qubits to be both 1 and 0 at the same time.

We may realize qubits in various physical systems. For example, a qubit can be represented by the states of an electron orbiting a single atom. This atom model may treat $|0\rangle$ and $|1\rangle$ as the so-called ground and excited states of the electron, respectively; if the atom is exposed to light with appropriate energy and for a suitable amount of time, we may move the electron from the $|0\rangle$ state to the $|1\rangle$ state, and vice versa. Moreover, by changing the length of exposure, we can move the electron initially in the state $|0\rangle$ to halfway between $|0\rangle$ and $|1\rangle$, say, into the state $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$, or the state $|-\rangle = (|0\rangle - |1\rangle)/\sqrt{2}$, where $|+\rangle$ and $|-\rangle$ constitute another qubit basis.

A classical bit can be checked to determine whether its state is 0 or 1; however, quantum physics indicates that we cannot examine a qubit to determine its state and find the values of α_0 and α_1. Qubits have stochastic behaviors that may be described by quantum probability, as defined in Section 2.3. We can measure a qubit to yield a measurement outcome that is either 0 with probability $|\alpha_0|^2$ or 1 with probability $|\alpha_1|^2$. Moreover, measuring a qubit will change its state.

Similarly to the classical case, we may define multiple qubits. A b-qubit can be described by $\mathbb{C}^{2^b}$, and its states are unit vectors in $\mathbb{C}^{2^b}$ with computational basis states $\langle x_1 \cdots x_b|$, $x_i = 0$ or 1, $i = 1, \ldots, b$. A given state can be expressed as a linear combination of the computational basis states with 2^b amplitudes. Quantum exponential complexity refers to the exponential growth in b of dimensionality of the vector space to describe the b-qubit and the number of amplitudes required to specify its superposition states (for details, see Nielsen & Chuang 2010, Wang 2012).

3.2. Quantum Entanglement

Quantum entanglement refers to the phenomenon of two particles acting in the same way, as twins that are linked by an unobserved wave and share each other's properties. Consider an entangled 2-qubit system. The quantum entanglement leads to an intriguing feature of the entangled state: Performing a measurement on one of the entangled qubits immediately casts the other one into the corresponding perfectly correlated state, which results in perfect correlation between the two measurement outcomes for the qubits. For example, take a 2-qubit system in a Bell state,

$$|\psi\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}. \qquad 2.$$

Performing a measurement on the first qubit of the Bell state $|\psi\rangle$, we obtain a random measurement outcome 0 or 1 with probability 1/2 and 1/2, respectively. However, after the measurement on the first qubit being either 0 or 1, the result of measuring the second qubit in the Bell state $|\psi\rangle$ is always 1 or 0, respectively. That is, there is a perfect correlation between the measurement results of the two qubits in $|\psi\rangle$. We refer to quantum states like the Bell state in Equation 2 as entangled states. The correlation phenomenon is called perfect anticorrelation in entanglement experiments (for more details, see Nielsen & Chuang 2010, Wang 2012, Wang & Song 2020).

3.3. Quantum Algorithms

The goal of quantum computation is to harness the enormous amount of information hidden in the quantum systems and utilize the exponential power of quantum particles for the purpose of computation. Classical computers are built by electrical circuits comprising wires for transferring information around the circuits and logic gates for accomplishing simple computational tasks. Similarly, quantum computers are created from quantum circuits with quantum gates to perform quantum computation and process quantum information. Despite the similarities, in contrast to classical computation, where transistors are used to crunch the ones and zeroes individually, quantum superposition can allow quantum computation to manage both one and zero at the same time and do the trick of carrying out simultaneous calculations. Moreover, the new quantum resources such as quantum superposition and quantum entanglement make it possible for quantum computers to outperform classical computers for certain tough tasks. Already it has been theoretically proven that many quantum algorithms can speed up the best-known classical algorithms, with examples including quadratic speedup for Grover's search algorithm and exponential speedup for Shor's factoring algorithm (see, e.g., Nielsen & Chuang 2010, Wang 2012).

3.4. Quantum Machine Learning

Quantum learning extends classical machine learning and statistical learning to the quantum realm. It studies how quantum resources can enhance classical learning in terms of computational complexity and statistical efficiency. Quantum computers can be faster than classical computers for solving certain machine learning problems, and it is possible for quantum learners to achieve higher statistical efficiency for some particular learning tasks, although there are caveats regarding quantum state preparation in some quantum machine learning algorithms. Examples of increased computational efficiency include support vector machines, principal component analysis, and BMs.

A case in point is quantum reinforcement learning. Classical reinforcement learning studies the problem of learning in and from interactive task environments, where a task environment is specified by a Markov decision process (MDP) through its states that are observed by an agent. The agent takes actions to produce transitions from states to states, and transitions are rated with rewards. The agent needs to learn which actions to perform in order to maximize the rewards. In reinforcement learning, the environments are unknown in the sense of unknown transition rules of MDPs, and the goal is to learn how to find optimal policies for achieving the maximum rewards, where a policy refers to a behavior rule to select actions based on states. Various procedures and algorithms are developed to estimate the so-called value functions and value-action functions and find optimal policies based on the estimated functions. Sutton & Barto (2018) provide more details.

We may consider a quantum approach to learning via interaction and establish a quantum framework for agents, environments, and their interactions—namely, a quantum agent-environment paradigm for quantum reinforcement learning. The quantum paradigm has the potential to lead to enhancements in both computational complexity and statistical efficiency of classical reinforcement learning. Moreover, we may mix the classical and quantum approaches for reinforcement learning frameworks. Depending on whether the agent and the environment are classical or quantum, we may obtain four agent-environment settings: classical agent and classical environment (CC), classical agent and quantum environment (CQ), quantum agent and classical environment (QC), and quantum agent and quantum environment (QQ) frameworks. The classification loosely corresponds to placing (classical) machine learning in CC; applications of machine learning to control quantum systems in CQ; quantum speedups in machine learning algorithms (like quantum annealers with nonquantum data) in QC; and quantum machine learning/reinforcement leaning with quantum agents, quantum environments, and quantum data in QQ.

Similar to quantum speedups in the case of supervised and unsupervised machine learning, quantum algorithms based on quantum walks and quantum Markov chains lead to provable quantum speedups in reinforcement learning. Like online learning, computational complexity and statistical efficiency may be closely connected in the context of reinforcement learning. Consider a reinforcement learning setting where an interaction is happening with respect to some external real time, and the environment alters with the passage of real time. For a learner with slower processing time relative to the environment alteration, the agent recognizes only some time average of the true environment, and the perceived blurred environment causes the learner to lose some statistical efficiency or even be unable to learn at all. However, a quantum learner can handle the environment change by facilitating the agent with enough time to learn before the environment changes and thus improve the statistical efficiency. Readers are directed to Biamonte et al. (2017), Ciliberto et al. (2018), Dunjko & Briegel (2018), Wang & Song (2020) and Wittek (2014) for more details.

4. QUANTUM SUPREMACY

Sections 3.3 and 3.4 mention quantum algorithms with theoretically proven quantum speedups, such as Shor's factoring algorithm, Grover's search algorithm, and other machine learning algorithms. However, from the algorithmic implementation point of view, we need to build quantum computers with a huge number of qubits in order to actually run the fast quantum algorithms on these large-scale quantum computers and practically demonstrate their theoretically proven quantum advantages, which is not possible at the present time due to the limitations of current technology. It is critical to acquire scalable architectures for constructing quantum computers with about 100 well-behaved qubits in the near-term future. We may employ such architectures to show so-called quantum computational supremacy, which refers to any practical major milestone achievement in the quest for outperforming classical computers on some tough computational tasks. Quantum computational supremacy is of great current interest in quantum computing and is being vigorously investigated by academic institutes, government labs, and private companies. We highlight below two landmark quantum supremacy projects (Arute et al. 2019, Zhong et al. 2020).

4.1. Random Quantum Circuits

This subsection presents the Google quantum supremacy study of Arute et al. (2019) that is based on random quantum circuits for solving a hard sampling problem. A quantum circuit is a quantum computation model in which a computation is a sequence of quantum gates, which are reversible transformations on a quantum mechanical analog of a classical n-bit register. Random quantum circuits are created as a quantum computation model with statistical sampling (in fact, a mixture of size-biased samplings) as its computation task. This model is proposed for the study of quantum computational supremacy. The computational task is to generate each random quantum circuit in a specific way so that we can sample from the output distribution corresponding to the generated quantum circuit. The goal is to construct random quantum circuits with enough complexity that even the most powerful classical supercomputer available at the time cannot directly simulate the constructed quantum circuits in practice.

4.1.1. Output distribution.

We now illustrate the way to generate random quantum circuits and describe their output distributions. A random quantum circuit refers to a sequence of clock cycles of 1-qubit and 2-qubit gates with gates applied to different qubits in the same cycle. The number of clock cycles, denoted by m, is called the depth of the circuit, and the number of qubits, denoted by n, is called the width of the circuit. When the gates to be put in use are randomly selected from the set of universal quantum gates, the unitary matrix $\mathbf{U}$ of the resulting quantum circuit is a random matrix. As the depth of the circuit goes to infinity, the distribution of the random unitary matrix $\mathbf{U}$ converges to the Haar measure on the unitary group of degree n.

Denote by $\mathcal{X} = \{|\mathbf{x}\rangle = |x_1 x_2 \ldots x_n\rangle : x_i \in \{0, 1\}\}$ the set of quantum computational basis states. Then $\mathcal{X}$ consists of $d = 2^n$ states. For a quantum circuit with a unitary matrix $\mathbf{U}$, let $|\psi_{\mathbf{U}}\rangle = \mathbf{U}|\psi_0\rangle$ be its output state, where ψ_0 is an input state. Given a computational basis state $|\mathbf{x}\rangle$, define measurement probability $p_{\mathbf{U}}(\mathbf{x}) = |\langle \mathbf{x}|\psi_{\mathbf{U}}\rangle|^2$—namely, the probability of obtaining measurement outcome $\mathbf{x}$. The quantum state of the random quantum circuit can be expressed as a linear combination of the computational basis with $d = 2^n$ amplitudes. As each amplitude has real and imaginary parts, there are a total of $2d = 2^{n+1}$ amplitude parameters. Because of the normalization constraint, the parameters lie in the unit sphere of a $2d$-dimensional Euclidean space. If the unitary matrix of the random quantum circuit follows the Haar distribution, the distribution of the amplitude parameters will be uniform on the unit sphere. Thus, as the depth m of the random quantum circuit goes to infinity, the measurement probability $p_{\mathbf{U}}(\mathbf{x})$ approaches the Porter–Thomas distribution (Rinott et al. 2020).

4.1.2. Challenges in output distribution sampling. Quantum supremacy can be demonstrated via random quantum circuits by checking quantum computers against state-of-the-art classical computers in the task of sampling the output distributions of random quantum circuits. A major part of the study of quantum supremacy is a purely statistical endeavor. It includes estimating the noise level in the quantum circuits, assessing their fidelity, and validating that the simulated bitstring data are actually generated from the claimed target distribution.

For a quantum computer, sampling the output distribution of a random quantum circuit means performing measurements on the qubits of the quantum circuit in the computational basis to generate a set of bitstrings, such as {0101101, 1001010, . . .}. Due to the noise in the circuit, the probability distribution of the observed bitstrings is different from the ideal output distribution $p_{\mathbf{U}}(\mathbf{x})$ as described in Sections 4.1.1, 4.1.3, and 4.1.4. However, because the complexity of a random quantum circuit grows exponentially in its size, which is defined by its width (the number of qubits) and depth (the number of cycles), classical algorithms for simulating its output distribution suffer from an exponential scaling of runtime with circuit size, and classical simulation of the output distribution is practically prohibitive. Indeed, at the time of the quantum supremacy study, classical sampling of the bitstring distribution is intractable in the quantum supremacy regime of random quantum circuits with 53 qubits and 20 cycles. Furthermore, the conventional tomographic estimation described in Section 2.4 scales exponentially in the circuit size, and an exponential number of bitstrings must be generated in order to statistically recover the circuit output distribution. These challenges motivate new statistical developments for the quantum supremacy study in the subsequent sections.

4.1.3. Cross-entropy benchmarking. Consider a sample $\mathcal{S} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_N\}$, where $\mathbf{x}_j$ are bitstrings obtained from measurements of every qubit in the computational basis. The joint distribution of $\mathcal{S}$ is given by $\Pr_{\mathbf{U}}(\mathcal{S}) = \prod_{i=1}^{N} p_{\mathbf{U}}(\mathbf{x}_i)$. An application of the central limit theorem leads to

$$\frac{1}{N} \log \Pr_{\mathbf{U}}(\mathcal{S}) = \frac{1}{N} \sum_{i=1}^{N} \log p_{\mathbf{U}}(\mathbf{x}_i) = -\mathcal{H}(p_{\mathbf{U}}) + O(N^{-1/2}),$$

where $\mathcal{H}(p_{\mathbf{U}})$ is the Shannon entropy of the output distribution $p_{\mathbf{U}}$.

For comparison, let us consider a sample $\mathcal{S}_\star = \{\mathbf{x}_1^\star, \mathbf{x}_2^\star, \ldots, \mathbf{x}_N^\star\}$ as outputs from a classical or quantum operation taking a specification of some random circuit $\mathbf{U}_\star$, where the distribution of $\mathbf{x}_j^\star$ depends on the unitary matrix $\mathbf{U}_\star$, and $\mathbf{x}_j^\star$ are uncorrelated with the output measurements $\mathbf{x}_j$ of the quantum circuit $\mathbf{U}$. For example, we may take $\mathbf{U}_\star$ as a noisy version of $\mathbf{U}$ in quantum computation, and $\mathbf{x}_j^\star$ are measured bitstrings obtained from a quantum circuit with noise (i.e., measurement outcomes of the quantum circuit $\mathbf{U}_\star$), while $\mathbf{x}_j$ are ideal bitstrings obtained from a quantum circuit without noise (i.e., measurement outcomes of the quantum circuit $\mathbf{U}$). Again applying the central limit theorem, we obtain

$$\frac{1}{N} \log \Pr_{\mathbf{U}}(\mathcal{S}_\star) = \frac{1}{N} \sum_{i=1}^{N} \log p_{\mathbf{U}}(\mathbf{x}_i^\star) = -\mathcal{H}(p_\star, p_{\mathbf{U}}) + O(N^{-1/2}),$$

where $p_\star(\mathbf{x}) = |\langle \mathbf{x} | \mathbf{U}_\star | \psi_0 \rangle|^2$ is the output distribution of $\mathbf{x}_j^\star \in \mathcal{S}_\star$ associated with $\mathbf{U}_\star$, and $\mathcal{H}(p_\star, p_{\mathbf{U}})$ is the cross-entropy between the two distributions. In the quantum supremacy study described below, we use the observed bitstrings $\mathbf{x}_i^\star$ to estimate the cross-entropy benchmarking by the average of the simulated probability distribution evaluated at the observed bitstrings $\mathbf{x}_i^\star$, where the simulated probability distribution refers to the ideal bitstring probability $p_{\mathbf{U}}(\cdot)$ for the quantum circuit $\mathbf{U}$ that is computed by classical simulations, and the average is over the bitstrings $\mathbf{x}_i^\star$ measured for the noisy quantum circuit $\mathbf{U}_\star$.

4.1.4. Quantum circuit model and fidelity estimation. The Google Quantum AI research group constructed a quantum processor (computer) named Sycamore, with 53 programmable superconducting qubits to implement random quantum circuits in a 2-dimensional lattice. The implementation creates quantum states on 53 qubits, corresponding to a computational state space of dimension 2^{53} ($\approx 10^{16}$). An approach based on the described cross-entropy benchmarking was adopted to handle noisy random quantum circuits and statistical sampling from bitstring probability distributions. Cross-entropy benchmarking (XEB), denoted by $\mathcal{F}_{\mathrm{XEB}}$, is defined as the expectation of a function of the ideal output distribution $p_{\mathbf{U}}(\cdot)$ with respect to the noisy output distribution.

To be specific, denote by $\mathcal{U}$ a set of r random quantum circuits $\mathbf{U}_1, \ldots, \mathbf{U}_r$ with n qubits and m cycles. Each circuit $\mathbf{U} \in \mathcal{U}$ is executed N times on the quantum processor, and every execution of the circuit $\mathbf{U}$ means that a quantum operation (corresponding to $\mathbf{U}_\star$, described in Section 4.1.3) as an imperfect realization of $\mathbf{U}$ (due to the noise) is applied to the input state $|\psi_0\rangle$ (with density matrix $|\psi_0\rangle\langle\psi_0|$). We model the imperfect realization of $\mathbf{U}$ by a noise model with the density matrix $\boldsymbol{\rho}_{\mathbf{U}}^\star$ of the noisy quantum operation as follows:

$$\boldsymbol{\rho}_{\mathbf{U}}^\star = \Upsilon|\psi_{\mathbf{U}}\rangle\langle\psi_{\mathbf{U}}| + (1-\Upsilon)\boldsymbol{\chi}_{\mathbf{U}}, \qquad 3.$$

where $|\psi_{\mathbf{U}}\rangle = \mathbf{U}|\psi_0\rangle$ is the ideal output state, $\Upsilon = \langle\psi_{\mathbf{U}}|\boldsymbol{\rho}_{\mathbf{U}}^\star|\psi_{\mathbf{U}}\rangle$ is the fidelity, and $\boldsymbol{\chi}_{\mathbf{U}}$ represents the density matrix of the noise that, along with fidelity Υ, describes the effect of the errors. The output probability distribution of $\boldsymbol{\rho}_{\mathbf{U}}^\star$ in Equation 3 is given by

$$p_{\mathbf{U},\Upsilon}(\mathbf{x}) = \Upsilon\langle\mathbf{x}|\psi_{\mathbf{U}}\rangle\langle\psi_{\mathbf{U}}|\mathbf{x}\rangle + (1-\Upsilon)\langle\mathbf{x}|\boldsymbol{\chi}_{\mathbf{U}}|\mathbf{x}\rangle = \Upsilon p_{\mathbf{U}}(\mathbf{x}) + (1-\Upsilon)\langle\mathbf{x}|\boldsymbol{\chi}_{\mathbf{U}}|\mathbf{x}\rangle. \qquad 4.$$

Then, an expression for the cross-entropy benchmarking $\mathcal{F}_{\mathrm{XEB}}$ is established as follows:

$$\mathcal{F}_{\mathrm{XEB}} = \Upsilon d \sum_{\mathbf{x}\in\{0,1\}^n} [p_{\mathbf{U}}(\mathbf{x})]^2 - \Upsilon = \Upsilon[d\,\langle p_{\mathbf{U}}(\cdot)\rangle_{p_{\mathbf{U}}} - 1], \qquad 5.$$

where $d = 2^n$, and $\langle p_{\mathbf{U}}(\cdot)\rangle_{p_{\mathbf{U}}}$ on the right-hand side denotes the expectation of the ideal output distribution $p_{\mathbf{U}}(\mathbf{x})$ with respect to itself that can be computed analytically or obtained numerically by simulations. Consider two special cases: (*a*) the bitstrings are sampled from the uniform distribution and (*b*) the bitstrings are sampled from the theoretical Porter–Thomas output distribution. For case *a*, we have $p_{\mathbf{U},\Upsilon}(\mathbf{x}) = 1/d$, and thus $\mathcal{F}_{\mathrm{XEB}} = \Upsilon = 0$. For case *b*, $p_{\mathbf{U},\Upsilon}(\mathbf{x}) = p_{\mathbf{U}}(\mathbf{x})$ is equal to the Porter–Thomas distribution, and hence $\mathcal{F}_{\mathrm{XEB}} = \Upsilon = 1$. Furthermore, for random quantum circuits with enough depth, their theoretical output distribution is essentially the Porter–Thomas distribution, and therefore from Equation 5 we conclude $\mathcal{F}_{\mathrm{XEB}} \doteq \Upsilon$—that is, $\mathcal{F}_{\mathrm{XEB}}$ is essentially the same as fidelity Υ even if bitstrings are sampled from noisy quantum circuits.

Equation 5 naturally leads to the following estimator of the cross-entropy benchmarking $\mathcal{F}_{\mathrm{XEB}}$ based on the observed bitstrings $\mathbf{x}_{ij}$ from random quantum circuit $\mathbf{U}_j \in \mathcal{U}$ with N bitstring samples; $i = 1, \ldots, N; j = 1, \ldots, r$:

$$\hat{\mathcal{F}}_{\mathrm{XEB}} = \frac{1}{Nr}\sum_{i=1}^{N}\sum_{j=1}^{r}[d\,p_{\mathbf{U}}(\mathbf{x}_{ij}) - 1], \qquad 6.$$

which has an asymptotic variance $(1 + 2\Upsilon - \Upsilon^2)/(Nr)$.

Cross-entropy and fidelity measure the closeness of two quantum states. For a quantum circuit with enough depth, $\mathcal{F}_{\mathrm{XEB}} = 1$ when there is no noise in the quantum circuit and thus bitstrings are sampled from the ideal theoretical output distribution of the circuit; $\mathcal{F}_{\mathrm{XEB}} = 0$ when depolarizing errors are overwhelming in the quantum circuit and thus bitstrings are sampled from the uniform distribution. Intuitively $\mathcal{F}_{\mathrm{XEB}}$ calibrates how often high-probability bitstrings are sampled, and its value corresponds to the probability that bitstrings are sampled from the ideal quantum circuits (i.e., no error has occurred while running the circuit). From Equations 5 and 6, we must obtain the ideal output probability $p_{\mathbf{U}}(\mathbf{x})$ by classically simulating the quantum circuit $\mathbf{U}$ for evaluating $\mathcal{F}_{\mathrm{XEB}}$ and $\hat{\mathcal{F}}_{\mathrm{XEB}}$, which is exponentially hard. Hence, it is intractable to compute $\mathcal{F}_{\mathrm{XEB}}$ and its estimates in the regime of quantum supremacy, such as random quantum circuits with 53 qubits and 20 cycles. As we present in Section 4.1.5 below, using quantum techniques to manipulate quantum circuits and statistical methodologies to model and analyze experimental data, a statistical extrapolation approach has been developed to statistically secure a high enough $\mathcal{F}_{\mathrm{XEB}}$ for random quantum circuits that are practically prohibitive for classical computers to simulate at the present time. The approach utilizes classical numerical simulations to evaluate the likelihood of observed bitstrings but does not require the reconstruction of the bitstring output probability distribution, which needs an exponential number of bitstrings for the increasing number of qubits.

4.1.5. Statistical analysis for quantum supremacy. Classical computers were used to simulate random quantum circuits for confirming the quantum computer and calculating $\mathcal{F}_{\mathrm{XEB}}$ and $\hat{\mathcal{F}}_{\mathrm{XEB}}$ as well as estimating the cost of classical sampling of bitstrings from the random quantum circuits. They were employed to verify that the quantum computer was working correctly by checking how often bitstrings were observed experimentally against their corresponding probabilities evaluated via classical simulations. The classical computers used included a Google cloud cluster of 1,000 machines and the Jülich supercomputer (with 100,000 cores and 250 terabytes), as well as the Summit supercomputer (the most powerful supercomputer in the world at the time).

We cannot evaluate $\mathcal{F}_{\mathrm{XEB}}$ and $\hat{\mathcal{F}}_{\mathrm{XEB}}$ for random quantum circuits in the supremacy regime. However, we may modify the design of random quantum circuits to reduce their complexity so that they can be classically simulated to obtain output distributions and compute $\mathcal{F}_{\mathrm{XEB}}$ and/or $\hat{\mathcal{F}}_{\mathrm{XEB}}$ for verifying physical models and validating statistical analysis. All the modified random quantum circuits closely mimicked the full experiment, with random quantum circuits in the supremacy regime while still remaining classically simulatable, and their experimental data and associated statistical analysis provided models to track the cross-entropy benchmarking fidelity $\mathcal{F}_{\mathrm{XEB}}$ in the supremacy regime. Various experiments were conducted for r modified random quantum circuits with n qubits and m cycles to collect N bitstrings, where N ranges from half a million to 5 million, with r, n, and m up to 10, 53, and 20, respectively. Experiments were carried out to collect $N = 3$ million bitstrings on each of $r = 10$ modified quantum circuits with $n = 53$ qubits and $m = 20$ cycles. The data were used to estimate $\mathcal{F}_{\mathrm{XEB}} \doteq \Upsilon$ and assess the output distribution, and various statistical methods were employed to evaluate the cross-entropy benchmarking fidelity estimator $\hat{\mathcal{F}}_{\mathrm{XEB}}$ and its asymptotic variance and validate the theoretical output distribution. Also, the systematic uncertainty (Sinervo 2003) was quantified by a linear fit to model how the fidelity varies over time, as the performance of the quantum system may fluctuate and/or degrade with time. After the statistical checking and validation, with a total of 30 million bitstring data, an estimated value of 2.24×10^{-3} was found for the mean cross-entropy benchmarking fidelity of $r = 10$ modified random quantum circuits with $n = 53$ qubits and $m = 20$ cycles, where the square root of its mean square error is estimated to be 0.21×10^{-3}. Based on these statistical results, Arute et al. (2019) conclude that the average fidelity for running the random quantum circuits on the Sycamore quantum computer is about 0.002.

The computing experiments show that it took 200 seconds for the Sycamore quantum computer to sample a million bitstrings from random quantum circuits with $n = 53$ qubits and $m = 20$ cycles at target fidelity $\mathcal{F}_{\text{XEB}} = 0.002$, while an extrapolation based on statistical fitting of the computing data by the mentioned classical computers indicates that an equal-fidelity classical sampling would take 10,000 years on a million cores, with further millions of years to confirm the fidelity using classical methods. Furthermore, the quantum computer consumed several orders of magnitude less energy to perform the sampling task than the Summit supercomputer would have. Therefore, quantum supremacy is demonstrated by the performed sampling task on the Sycamore quantum computer that is practically beyond the reach of the fastest classical supercomputers available at the time. Readers are directed to Arute et al. (2019), Aaronson & Chen (2016), Boixo et al. (2018), Bouland et al. (2018), Neill et al. (2018), and Rinott et al. (2020) for more details.

We would like to point out again that the Google quantum supremacy study heavily relies on statistics. In spite of the extensive statistical analysis conducted by Arute et al. (2019), there are many statistical issues that deserve further investigation. For example, the authors of this article found that the noisy quantum circuit model does not fit to the generated bitstrings of Arute et al. (2019).

4.2. Boson Sampling

This subsection describes the boson sampling quantum supremacy study reported by Zhong et al. (2020). Boson sampling is a special quantum computation model based on linear optics where the required physical devices are single-photon sources, beam splitters, phase shifters and photon detectors. The quantum computation model for boson sampling arranges n identical bosons to pass through a network of passive optical elements (beam splitters and phase shifters) and then detects the locations of the bosons, and its purpose is to sample from the output distribution for demonstrating quantum supremacy. We introduce two equivalent ways to define the boson sampling model, where one naturally leads to quantum computation and the other directly shows the difficulty in classical computation.

4.2.1. Physical model definition. Consider the quantum system involving n identical photons and m modes, where mode can be loosely interpreted as the place that a photon can be in, and we are only interested in the case that m is greater than or equal to n. Note that we may write the computational basis states in the form of $|\mathbf{s}\rangle = |s_1, s_2\ , \ldots, s_m\rangle$, where s_i is the number of photons detected in the ith mode. Denote the set corresponding to all the computational basis states by

$$\Omega_{m,n} = \{\mathbf{s} = (s_1, s_2, \ldots, s_m) : \ s_i \in \mathbb{N}, \ s_1 + s_2 + \cdots + s_m = n\}.$$

The number of elements in the set $\Omega_{m,\,n}$ is $M = \binom{m+n-1}{n}$. Since a general state can be expressed as a linear combination of the computational basis states with complex coefficients whose squared norms sum up to 1, we may write a general computational state for the boson computer with n photons and m modes in the following form:

$$|\psi\rangle = \sum_{\mathbf{s}\in\Omega_{m,n}} \alpha_{\mathbf{s}}|\mathbf{s}\rangle, \quad \text{where} \sum_{\mathbf{s}\in\Omega_{m,n}} |\alpha_{\mathbf{s}}|^2 = 1.$$

With no loss of generality, we assume the initial state of a quantum computer to be $|\mathbb{1}_n\rangle \equiv |1, \ldots\ldots, 1, 0, \ldots\ldots, 0\rangle$, which means that each of the first n modes contains one photon and the remaining modes do not contain any photons. Consider how the linear optical elements work in a special case when there is only one photon in the quantum system. Each phase shifter and beam splitter takes action on at most two modes, without any action on the other $m - 2$ modes. Assume that a phase shifter takes action on the ith mode, and $\mathbf{s} \in \Omega_{m,n}$ indicates the photon in the ith mode. Then the phase shifter changes only one amplitude, $\alpha_{\mathbf{s}}$, by multiplying it with $e^{i\theta}$ for some specified θ, but it does not alter any other amplitudes. Suppose that a beam splitter takes action on the ith and jth modes, and denote the corresponding quantum states by $|\mathbf{s}\rangle$ and $|\mathbf{t}\rangle$, respectively. Then the action maps the two amplitudes $\alpha_{\mathbf{s}}$ and $\alpha_{\mathbf{t}}$ into $\breve{\alpha}_{\mathbf{s}}$ and $\breve{\alpha}_{\mathbf{t}}$, where their relationship can be specified by the following transformation:

$$\begin{bmatrix} \breve{\alpha}_{\mathbf{s}} \\ \breve{\alpha}_{\mathbf{t}} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \alpha_{\mathbf{s}} \\ \alpha_{\mathbf{t}} \end{bmatrix}.$$

As the quantum system in this case has only one photon, there are a total of m computational basis states. Consequently, the unitary matrices representing the transformations of a beam splitter and a phase shifter are equal to the m-dimensional identity matrices except for a 2×2 submatrix and a diagonal entry corresponding to the amplitude change, respectively. Then the product of such matrices corresponding to the optical elements in a linear optical network yields an $m \times m$ unitary matrix $\mathbf{U}$ to represent the unitary transformation of the linear optical network. Conversely, any $m \times m$ unitary matrix $\mathbf{U}$ can be mathematically decomposed as a product $\mathbf{U} = \mathbf{U}_T \ldots \mathbf{U}_1$, where each $\mathbf{U}_i$ corresponds to a unitary matrix of a beam splitter or a phase shifter, and $T = O(m^2)$. That is, we can use only linear optical elements to implement any $m \times m$ unitary transformation.

Given a quantum system with n photons, it can be shown that the unitary transformation corresponding to the optical network has a unitary matrix representation $\phi(\mathbf{U}) = \phi(\mathbf{U}_T) \ldots \phi(\mathbf{U}_1)$, where ϕ is a homomorphism map. We just need to identify the specification of each $\phi(\mathbf{U}_i)$ to obtain an explicit expression for $\phi(\mathbf{U})$.

Assume that the ith optical element is a phase shifter. We can mathematically express its action as the following diagonal unitary transformation:

$$|s_1, s_2, \ldots, s_m\rangle \to e^{i\theta s_i}|s_1, s_2, \ldots, s_m\rangle,$$

and the corresponding unitary matrix representation yields an expression for $\phi(\mathbf{U}_i)$.

Assume that the jth optical element is a beam splitter. The action of the beam splitter is hard to describe. It takes action on two modes only, without making any change for the other $m-2$ modes. Suppose that the two modes the beam splitter has acted on are the ith and jth modes. Then the corresponding unitary transformation can be expressed as follows:

$$\begin{aligned} &|s_1, s_2, \ldots, s_{i-1}, u, s_{i+1}, \ldots, s_{j-1}, v, s_{j+1}, \ldots, s_m\rangle \\ &\to \sum_{s+t=u+v} \beta_{u,v,s,t}|s_1, s_2, \ldots, s_{i-1}, s, s_{i+1}, \ldots, s_{j-1}, t, s_{j+1}, \ldots, s_m\rangle, \end{aligned}$$

where

$$\beta_{u,v,s,t} = \sqrt{\frac{u!v!}{s!t!}} \sum_{k+l=u, k\leq s, l\leq t} \binom{s}{k}\binom{t}{l}(-1)^{s-k}(\sin\theta)^{s+l-k}(\cos\theta)^{k+t-l}$$

for some specified angle θ. The unitary matrix corresponding to this transformation renders an expression for $\phi(\mathbf{U}_j)$.

Using the derived expression for each $\phi(\mathbf{U}_i)$ along with $\phi(\mathbf{U}) = \phi(\mathbf{U}_T) \ldots \phi(\mathbf{U}_1)$, we can obtain the unitary matrix for representing the action of the whole optical network. Initiating in the state $|\mathbb{1}_n\rangle$ and then passing through the optical network, the photons will be in the quantum state $\phi(\mathbf{U})|\mathbb{1}_n\rangle$. We measure the state to obtain a measurement outcome corresponding to a computational basis state. The measurement we obtain is random. Treating $|\langle \mathbb{1}_n|\phi(\mathbf{U})|\mathbf{s}\rangle|^2$ as a mapping from $\mathbf{s} \in \Omega_{m,n}$ to [0,1], we obtain a probability distribution on $\Omega_{m,n}$ that assigns probability $\Pr(\mathbf{s})$ to state $|\mathbf{s}\rangle$, where

$$\Pr(\mathbf{s}) = |\langle \mathbb{1}_n|\phi(U)|\mathbf{s}\rangle|^2, \quad \mathbf{s} \in \Omega_{m,n}. \qquad 7.$$

4.2.2. Permanent based model definition. Alternatively, we may define boson sampling by the permanents of the submatrices of the unitary matrix representing the optical network. Let $\mathbf{A} = (a_{ij})$ be a $n \times n$ matrix, and define its permanent as follows:

$$\text{Per}(\mathbf{A}) = \sum_{\pi \in \mathcal{S}_n} \prod_{i=1}^{n} a_{i\pi(i)},$$

where $\mathcal{S}_n$ is the set of all permutations of $1, \ldots, n$.

Given an $m \times m$ unitary matrix $\mathbf{U}$ and $\mathbf{s} = (s_1, \ldots, s_m) \in \Omega_{m,n}$, we construct an $n \times n$ matrix $\mathbf{U}_\mathbf{s}$ from $\mathbf{U}$ by retaining its first n columns and then copying s_i times its ith row. Define a discrete probability distribution on $\Omega_{m,n}$ as follows:

$$\text{Pr}(\mathbf{s}) = \frac{|\text{Per}(\mathbf{U}_\mathbf{s})|^2}{s_1! \ldots s_m!}. \qquad 8.$$

It can be proved that the probability distribution Equation 8 is equal to the probability distribution Equation 7 in the way described in Section 4.2.1, through the system with n photons, m modes, and an optical network whose action is represented by the unitary matrix $\mathbf{U}$.

4.2.3. Quantum supremacy with boson sampling. The boson sampling problem is defined as sampling from the distribution $\text{Pr}(\mathbf{s})$ defined in Equation 7 or Equation 8. Gaussian boson sampling makes use of Gaussian states as probability sources of photons, and the resulting output distribution can be further expressed as matrix functions called Hafnian and Torontonian. Since the permanent, Hafnian, and Torontonian matrix functions are in the #P-complete complexity class, it is intractable for classical computers to evaluate the matrix functions and thus handle boson sampling. However, the physical definition inherently shows that it is possible to carry out a successful quantum computing experiment on an optical network with appropriate size and thus render quantum supremacy.

The quantum supremacy study reported by Zhong et al. (2020) built a photonic quantum computer (processor) called Jiuzhang to perform Gaussian boson sampling. Jiuzhang can enable up to 76 qubits to successfully accomplish Gaussian boson sampling tasks that are beyond the capacity of the fastest classical supercomputers available at the time. Zhong et al. (2020) documented physical experiments performed and statistical analysis undertaken to verify quantum states and validate output distributions based on the generated samples in the easy regime where the full output distributions can be obtained. They provided circumstantial evidence to support the results in the quantum supremacy regime where a full verification is not possible due to the intractable nature of boson sampling. Also, time costs to run Gaussian boson sampling on supercomputers in the prohibitive regime were estimated based on statistical fitting of classical computational cost data. In a nutshell, Zhong et al. (2020) announced that a 200-second job of Gaussian boson sampling on Jiuzhang would require 0.6 billon years for the fastest supercomputer available at the time to finish. Hence, quantum supremacy is demonstrated by the performed Gaussian boson sampling on the photonic quantum computer that was practically beyond the reach of the fastest classical supercomputers available at the time. More details are provided by Aaronson & Arkhipov (2011), Hamilton et al. (2017), Harrow & Montanaro (2017), Lund et al. (2017), Markov et al. (2018), Quesada et al. (2018) and Zhong et al. (2020).

5. QUANTUM ANNEALING

Quantum annealing is the quantum analog of classical annealing, with thermodynamics replaced by quantum dynamics. Both classical annealing and quantum annealing are employed to solve optimization problems whose objective function can be represented by the energies of physical systems. Quantum annealing may be considered a special-purpose quantum computer that is designed to effectively solve specific optimization problems, and quantum annealers are physical hardware devices to implement quantum annealing (see McGeoch 2014, Wang et al. 2016).

5.1. Classical Ising Model

Consider a classical Ising model described by a graph $\mathcal{G} = (\mathcal{V}(\mathcal{G}), \mathcal{E}(\mathcal{G}))$, where $\mathcal{V}(\mathcal{G})$ and $\mathcal{E}(\mathcal{G})$ are the vertex and edge sets of $\mathcal{G}$, respectively. We assign each vertex a random variable with a value in $\{+1, -1\}$ and let each edge represent the interaction between two vertex variables connected by the edge. A configuration $\mathbf{s} = \{s_j, j \in \mathcal{V}(\mathcal{G})\}$ is a set of values assigned to all vertex variables s_j, $j \in \mathcal{V}(\mathcal{G})$. In physics, vertices and vertex variables are called sites and spins, respectively, and we refer to the spin values $+1$ and -1 as spin up and spin down, respectively. Define the following Hamiltonian for the classical Ising model:

$$\mathbf{H}_I^c \equiv \mathbf{H}_I^c(\mathbf{s}) = -\sum_{(i,j)\in\mathcal{E}(\mathcal{G})} \delta_{ij} s_i s_j - \sum_{j\in\mathcal{V}(\mathcal{G})} \gamma_j s_j, \qquad 9.$$

where (i,j) represents the edge between the sites i and j, the first summation is over all $(i, j) \in \mathcal{E}(\mathcal{G})$, δ_{ij} denotes the interaction between sites i and j associated with edge $(i, j) \in \mathcal{E}(\mathcal{G})$, and γ_j stands for an external magnetic field on vertex $j \in \mathcal{V}(\mathcal{G})$. We refer to a set of fixed values $\{\delta_{ij}, \gamma_j\}$ as an instance of the Ising model and $\mathbf{H}_I^c(\mathbf{s})$ as the energy of the Ising model at configuration $\mathbf{s}$. Then, the probability of a specific configuration $\mathbf{s}$ is described by the Boltzmann distribution of the Ising model as follows:

$$\mathrm{P}_T(\mathbf{s}) = \frac{e^{-\mathbf{H}_I^c(\mathbf{s})/T}}{Z_T}, \qquad Z_T = \sum_{\mathbf{s}} e^{-\mathbf{H}_I^c(\mathbf{s})/T}, \qquad 10.$$

where T is the fundamental temperature of the system in units of energy.

We illustrate the classical annealing by the Ising model as follows. The annealing is used to solve a combinatorial minimization problem whose objective function is represented by the Ising energy function $\mathbf{H}_I^c(\mathbf{s})$. Denote by b the total number of sites in the Ising model. As $\mathbf{H}_I^c(\mathbf{s})$ is defined over $\mathbf{s} \in \{-1, +1\}^b$, it is prohibitive to search over the exponential large configuration space for a minimizer of $\mathbf{H}_I^c(\mathbf{s})$ by deterministic exhaustive search algorithms. We resort to annealing methods to explore the huge search space probabilistically and search for a configuration with the minimal energy. For example, we implement simulated annealing by using Markov chain Monte Carlo (MCMC) to generate configurations from the Boltzmann distribution $\mathrm{P}_T(\mathbf{s})$ with slowly decreasing temperature T. The lowest energy state is often called a ground state in physics (see Bertsimas & Tsitsiklis 1993, Kirkpatrick et al. 1983, Wang et al. 2016).

5.2. Quantum Ising Model

The same graph $\mathcal{G}$ is used to describe the quantum Ising model, where the vertex set $\mathcal{V}(\mathcal{G})$ stands for the quantum spins, with the edge set $\mathcal{E}(\mathcal{G})$ for the interactions between two quantum spins. Each vertex has a qubit that is realized by its quantum spin. As $\mathcal{G}$ has b vertices, the vector space for the described quantum Ising system is $\mathbb{C}^d$ $(d = 2^b)$. We characterize its quantum state by a unit vector in $\mathbb{C}^d$ and its dynamic evolution by a Hermitian matrix of size d, which is called a quantum Hamiltonian for the quantum system. The energies of the quantum system are defined to be the eigenvalues of the quantum Hamiltonian, and ground states refer to the eigenvectors corresponding to the smallest eigenvalue. Set

$$\mathbf{I}_j = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{\sigma}_j^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \boldsymbol{\sigma}_j^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \qquad j = 1, \ldots, b,$$

where $\boldsymbol{\sigma}_j^x$ and $\boldsymbol{\sigma}_j^z$ are called Pauli matrices in x and z axes, respectively. We replace s_j in the classical Ising Hamiltonian $\mathbf{H}_I^c(\mathbf{s})$ by $\boldsymbol{\sigma}_j^z$ to define the quantum Hamiltonian as follows:

$$\mathbf{H}_I^q = -\sum_{(i,j)\in\mathcal{E}(\mathcal{G})} \delta_{ij}\boldsymbol{\sigma}_i^z\boldsymbol{\sigma}_j^z - \sum_{j\in\mathcal{V}(\mathcal{G})} \gamma_j\boldsymbol{\sigma}_j^z, \tag{11.}$$

where γ_j and δ_{ij} stand for the local field on the vertex $j \in \mathcal{V}(\mathcal{G})$ and the Ising interaction along the edge $(i, j) \in \mathcal{E}(\mathcal{G})$, respectively, and we use the quantum convention that $\boldsymbol{\sigma}_j^z$ and $\boldsymbol{\sigma}_i^z\boldsymbol{\sigma}_j^z$ in Equation 11 denote their tensor products with identity matrices

$$\boldsymbol{\sigma}_j^z \equiv \mathbf{I}_1 \otimes \cdots \otimes \mathbf{I}_{j-1} \underbrace{\otimes\, \boldsymbol{\sigma}_j^z \otimes}_{\text{vertex } j} \mathbf{I}_{j+1} \otimes \cdots \otimes \mathbf{I}_b \tag{12.}$$

and

$$\boldsymbol{\sigma}_i^z\boldsymbol{\sigma}_j^z \equiv \mathbf{I}_1 \otimes \cdots \otimes \mathbf{I}_{i-1} \otimes \underbrace{\boldsymbol{\sigma}_i^z \otimes \mathbf{I}_{i+1} \otimes \cdots \otimes \mathbf{I}_{j-1} \otimes \boldsymbol{\sigma}_j^z}_{\text{vertices } i \text{ and } j} \otimes \mathbf{I}_{j+1} \otimes \cdots \otimes \mathbf{I}_b. \tag{13.}$$

Observe that the quantum Hamiltonian $\mathbf{H}_I^q$ in Equation 11 is a diagonal matrix of size 2^b whose diagonal elements (eigenvalues) are the same as the classical Hamiltonian $\mathbf{H}_I^c(\mathbf{s})$ in Equation 9 corresponding to all 2^b binary states $\mathbf{s}$ ordered lexicographically. Therefore, finding the minimal energy of the classical Ising Hamiltonian $\mathbf{H}_I^c$ is equivalent to finding the minimal energy of the quantum Ising Hamiltonian $\mathbf{H}_I^q$. The quantum formulation of the original optimization problem is to facilitate the design of quantum annealing in the next subsection, but so far the computational task for solving the optimization problem is still the same as in the classical case.

5.3. Quantum Ising Model in the Transverse Field

Quantum annealing requires an introduction of a transverse magnetic field to yield a quantum Hamiltonian in the transverse field. Define a quantum Hamiltonian to govern the transverse magnetic field as follows:

$$\mathbf{H}_X = -\sum_{j\in\mathcal{V}(\mathcal{G})} \boldsymbol{\sigma}_j^x, \tag{14.}$$

where again we adopt the quantum convention to denote by $\boldsymbol{\sigma}_j^x$ the tensor products of b matrices of size 2,

$$\boldsymbol{\sigma}_j^x \equiv \mathbf{I}_1 \otimes \cdots \otimes \mathbf{I}_{j-1} \underbrace{\otimes\, \boldsymbol{\sigma}_j^x \otimes}_{\text{vertex } j} \mathbf{I}_{j+1} \otimes \cdots \otimes \mathbf{I}_b. \tag{15.}$$

Observe that the $2^b \times 2^b$ nondiagonal matrix $\mathbf{H}_X$ in Equation 14 does not commute with diagonal matrix $\mathbf{H}_I^q$ in Equation 11, and thus introducing $\mathbf{H}_X$ changes the system behavior from classical to quantum. Due to the simple symmetric structure of $\mathbf{H}_X$, we can derive explicit expressions for its eigenvalues and eigenvectors. In particular, $(1, \ldots, 1)'$ is its ground state, the eigenvector corresponding to the smallest eigenvalue of $\mathbf{H}_X$.

Quantum annealing proceeds as follows. The quantum annealing system is initially driven by the transverse magnetic field $\mathbf{H}_X$ prepared in its ground state $(1, \ldots, 1)'$, and then we slowly drive the system from the initial Hamiltonian $\mathbf{H}_X$ to its final target Hamiltonian $\mathbf{H}_I^q$. Specifically, we engineer the quantum annealing process through the instantaneous Hamiltonian for the Ising model in the transverse field as follows:

$$\mathbf{H}_D(t) = A(t)\mathbf{H}_X + B(t)\mathbf{H}_I^q, \qquad t \in [0, t_f], \tag{16.}$$

where t_f denotes the total annealing time; time-varying smooth functions $A(t)$ and $B(t)$ are called the annealing schedules, which satisfy $A(t_f) = B(0) = 0$; $A(t)$ is decreasing; and $B(t)$ is increasing. It is evident that at $t = 0$, $\mathbf{H}_D(0) = A(0)\mathbf{H}_X$, and at $t = t_f$, $\mathbf{H}_D(t_f) = B(t_f)\mathbf{H}_I^q$. As $A(0)$ and $B(t_f)$ are known constants, $\mathbf{H}_D(t)$ has identical eigenvectors as $\mathbf{H}_X$ at the initial time $t = 0$ and as $\mathbf{H}_I^q$ at the final time $t = t_f$, with the corresponding eigenvalues differing by factors $A(0)$ and $B(t_f)$, respectively. Consequently, through $\mathbf{H}_D(t)$, we can engineer the system to move from $\mathbf{H}_X$ initialized in its ground state to the final target $\mathbf{H}_I^q$.

The adiabatic quantum theorem shows that during the evolution of quantum annealing, the system tends to stay in the ground states of the instantaneous Hamiltonian via quantum tunneling (Farhi et al. 2000, 2001, 2002; McGeoch 2014). At the end of the annealing procedure, we measure the quantum system. With some probability, the quantum system stays in a ground state of the final Hamiltonian $\mathbf{H}_I^q$, and thus the measurement outcome renders a solution to the optimization problem. That is, we can utilize the quantum annealing procedure driven by Equation 16 to find the global minimum of $\mathbf{H}_I^c(\mathbf{s})$ and solve the original minimization problem. For details, readers may refer to Brooke et al. (1999), Hu & Wang (2021), Isakov et al. (2016), Jörg et al. (2010), Wang et al. (2016), and Wang & Song (2020).

5.4. Quantum Annealer

Quantum annealers, quantum devices to implement quantum annealing, are currently being investigated by a number of academic labs and companies, with uncertain quantum speedup. In particular, the D-Wave machine is a commercially available hardware device that is designed and built to physically implement quantum annealing. It is an analog computing device based on superconducting qubits to process quantum annealing and solve certain combinatorial optimization problems. Many experiments have been conducted to test D-Wave machines, and computational studies, such as Monte Carlo simulations, have been carried out to assess the performance of D-Wave machines and compare it with classical and quantum models through sophisticated statistical analysis. It has been demonstrated that D-Wave machines are useful in designing quantum algorithms and solving application problems and can be faster than classical algorithms like classical annealing, yet no quantum supremacy or quantum speedup over classical computation has been found in D-Wave machines. Readers are directed to Albash et al. (2015), Boixo et al. (2014, 2016, 2018), Brady & van Dam (2016), Rønnow et al. (2014), and Wang et al. (2016) for more discussion.

6. QUANTUM LEARNING WITH BOLTZMANN MACHINES

BMs have been introduced as probabilistic generative models that contain bidirectionally connected networks of stochastic binary units and can be interpreted as neural network models. They can be regarded as particular graphical models—more precisely, undirected graphical models known as Markov random fields. BMs provide a model for deep learning architectures such as deep belief networks. They have the potential to learn internal representations for complex unsupervised learning tasks such as object and speech recognition problems. A BM model is mathematically equivalent to the Ising model in physics. This provides a new way to sample from a BM: (*a*) map the BM to the corresponding Ising model, (*b*) engineer a physical system to realize the target problem to be solved, (*c*) run the physics until the Ising system establishes some equilibrium with a state corresponding to a possible solution, (*d*) measure the physical system to obtain a realization of states of the Ising model, and (*e*) map the Ising measurement outputs into these corresponding to the BM to render a possible solution to the original problem. The procedure requires relevant physical implementations or computer simulations, which have been employed

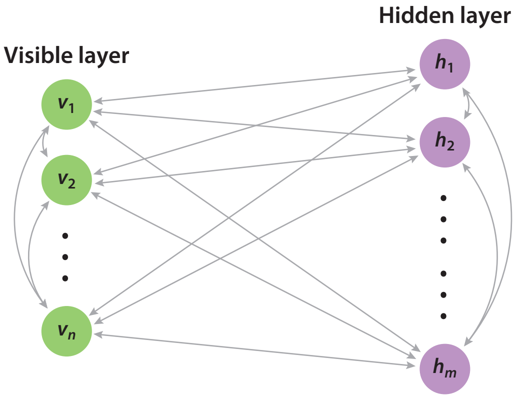

Figure 1

General Boltzmann machine. The diagram shows that every two nodes are connected to create a fully connected undirected graph.

in both classical and quantum domains with examples including classical and quantum annealing. We review classical BMs and describe their training by quantum means; we also introduce quantum BMs for deep learning, where quantum deep learning here refers to both quantum BMs and the training of classical BMs by quantum resources. The scenarios may bear some resemblance to quantum reinforcement learning considered in Section 3.4, where mixtures of classical and quantum approaches are used in the quantum reinforcement learning framework.

6.1. Boltzmann Machines

A BM is a network of symmetrically coupled stochastic binary units, which consists of a set of visible units $\mathbf{v} \in \{0,1\}^n$ associated with observations and a set of hidden units $\mathbf{h} \in \{0,1\}^m$ used to capture dependencies between observed variables. Every two nodes are connected, so the model creates a fully connected undirected graph as illustrated in **Figure 1**. The model has a joint distribution

$$p(\mathbf{v},\mathbf{h};\theta) = \frac{1}{Z(\theta)} e^{-E(\mathbf{v},\mathbf{h};\theta)} \text{ and } Z(\theta) = \sum_{\mathbf{v}}\sum_{\mathbf{h}} \exp(-E(\mathbf{v},\mathbf{h});\theta),$$

where $Z(\theta)$ is called the partition function; the energy function $E(\mathbf{v},\mathbf{h};\theta)$ is defined as

$$E(\mathbf{v},\mathbf{h};\theta) = -\frac{1}{2}\mathbf{v}^T\mathbf{L}\mathbf{v} - \frac{1}{2}\mathbf{h}^T\mathrm{J}\mathbf{h} - \mathbf{v}^T\mathbf{W}\mathbf{h};$$

$\theta = \{\mathbf{W}, \mathbf{L}, \mathrm{J}\}$ is the model parameter; and matrices $\mathbf{W} = (W_{ij})$, $\mathbf{L} = (L_{ij})$, and $\mathrm{J} = (J_{ij})$ represent visible-to-hidden, visible-to-visible, and hidden-to-hidden symmetric interaction terms. The diagonal elements of $\mathbf{L}$ and J are set to 0. The conditional distributions of hidden and visible units are given by

$$p(h_j = 1|\mathbf{v},\mathbf{h}_{-j}) = \sigma\left(\sum_i W_{ij}v_i + \sum_{l\neq j} J_{jl}h_j\right) \text{ and}$$

$$p(v_i = 1|\mathbf{h},\mathbf{v}_{-i}) = \sigma\left(\sum_j W_{ij}h_j + \sum_{k\neq i} L_{ik}v_j\right),$$

where $\sigma(x) = 1/(1 + e^{-x})$ is the sigmoid function. The marginal distribution of the visible units is

$$p(\mathbf{v}; \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}; \theta)).$$

Given the observed data, the training of a BM consists of finding the parameter θ that maximizes the log-likelihood function $\log p(\mathbf{v}; \theta)$. Because for a general BM it is not possible to analytically find the maximizer of the likelihood, the usual approach is to apply gradient ascent, iteratively updating $\theta^{(t)}$ to $\theta^{(t+1)}$ by the gradient of the log-likelihood,

$$\theta^{(t+1)} = \theta^{(t)} + \eta \frac{\partial}{\partial \theta^{(t)}} \left(\log p(\mathbf{v}; \theta^{(t)})\right),$$

where η is the learning rate. It can be shown that the parameter increment during the iteration has the following expressions:

$$\Delta \mathbf{W} = \eta(\mathrm{E}_{\mathrm{P}_{\mathrm{data}}}[\mathbf{v}\mathbf{h}'] - \mathrm{E}_{\mathrm{P}_{\mathrm{model}}}[\mathbf{v}\mathbf{h}']), \quad \Delta \mathbf{L} = \eta(\mathrm{E}_{\mathrm{P}_{\mathrm{data}}}[\mathbf{v}\mathbf{v}'] - \mathrm{E}_{\mathrm{P}_{\mathrm{model}}}[\mathbf{v}\mathbf{v}']), \text{ and}$$

$$\Delta \mathrm{J} = \eta(\mathrm{E}_{\mathrm{P}_{\mathrm{data}}}[\mathbf{h}\mathbf{h}'] - \mathrm{E}_{\mathrm{P}_{\mathrm{model}}}[\mathbf{h}\mathbf{h}']),$$

where $\mathrm{E}_{\mathrm{P}_{\mathrm{model}}}$ and $\mathrm{E}_{\mathrm{P}_{\mathrm{data}}}$ denote, respectively, the expectations with respect to the model distribution $p(\mathbf{v}, \mathbf{h}; \theta)$ and the data distribution $\mathrm{P}_{\mathrm{data}}(\mathbf{h}, \mathbf{v}; \theta) = p(\mathbf{h}|\mathbf{v}; \theta)\mathrm{P}_{\mathrm{data}}(\mathbf{v})$, and $\mathrm{P}_{\mathrm{data}}(\mathbf{v})$ is the empirical distribution of the observed data. We call $\mathrm{E}_{\mathrm{P}_{\mathrm{data}}}$ the data-dependent expectation, and since $\mathrm{E}_{\mathrm{P}_{\mathrm{model}}}$ represents an expectation taken with respect to the joint distribution $p(\mathbf{v}, \mathbf{h}; \theta)$ of the visible and hidden variables, we call it the data-independent expectation or model expectation. For general BMs, it is usually very difficult to directly compute the model expectation due to its exponential growth in the units.

We employ MCMC approaches to estimate the data-dependent expectation and the data-independent expectation. In particular, for the computation of model expectation, the conditional distribution of every node given the other nodes is known, so standard MCMC simulation methods are often employed to compute the model expectation, although MCMC can be very costly or even impossible for large BMs (for more details, see Hinton & Salakhutdinov 2012; Salakhutdinov 2015; Salakhutdinov & Hinton 2009, 2012).

6.2. Restricted Boltzmann Machines

Because the learning process for general BMs is time consuming, we may impose some restrictions on the network topology to simplify the learning problem. A restricted Boltzmann machine (RBM) model is a special variant of BMs where every visible node is connected to every hidden node, but there is no connection between two variables of the same layer, as shown in **Figure 2**. One major advantage of the RBM is that its model expectations are easy to calculate. Again it contains n visible units $\mathbf{v} = (v_1, \ldots, v_n)$ associated with observations and m hidden units $\mathbf{h} = (h_1, \ldots, h_m)$ to capture dependencies between observed variables. The joint distribution of the RBM model is given by $p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}$ with the energy function

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} h_i v_j - \sum_{j=1}^{n} b_j v_j - \sum_{i=1}^{m} c_i h_i.$$

Thus, we have

$$p(\mathbf{h}|\mathbf{v}) = \prod_{i=1}^{m} p(h_i|\mathbf{v}) \text{ and } p(\mathbf{v}|\mathbf{h}) = \prod_{i=1}^{n} p(v_i|\mathbf{h}).$$

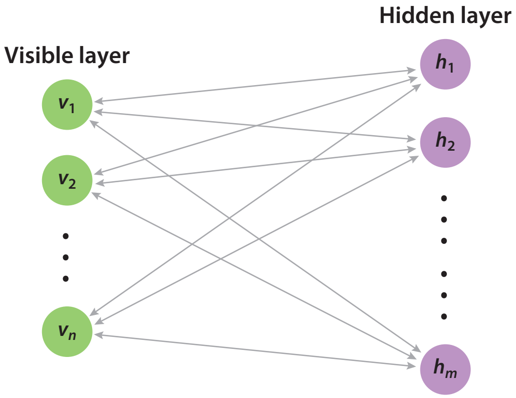

Figure 2

Restricted Boltzmann machine. The diagram shows that every visible node is connected to every hidden node, but there is no connection between two variables of the same layer.

Since the connections between hidden variables are absent, the marginal distribution of the visible variables has a simplified expression,

$$p(\mathbf{v}) = \frac{1}{Z}\sum_{\mathbf{h}} p(\mathbf{v},\mathbf{h}) = \frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})} = \frac{1}{Z}\prod_{j=1}^{n}\left(e^{b_j v_j}\right)\prod_{i=1}^{m}\left(1+e^{c_i+\sum\limits_{j=1}^{n} w_{ij}v_j}\right).$$

The conditional distribution of h_i given $\mathbf{v}$ and the conditional distribution of v_j given $\mathbf{h}$ are, respectively, given by

$$p(h_i = 1|\mathbf{v}) = \sigma\left(\sum_{j=1}^{n} w_{ij}v_j + c_i\right) \quad \text{and} \quad p(v_j = 1|\mathbf{h}) = \sigma\left(\sum_{i=1}^{m} w_{ij}h_i + b_j\right),$$

where $\sigma(x)$ is the sigmoid function. For the RMB model with $\theta = (w_{ij}, b_j, c_i)$, we can obtain an explicit form for the gradient of the log-likelihood function, giving

$$\theta^{(t+1)} = \theta^{(t)} + \eta\frac{\partial}{\partial\theta^{(t)}}\left[\sum_{i=1}^{n}\ln L(\theta^{(t)}|v_i)\right] - \lambda\theta^{(t)} + \epsilon\Delta\theta^{(t-1)},$$

where η, λ, and ϵ are positive constants, representing the learning rate, moment weight, and momentum coefficient, respectively, and $L(\theta|\mathbf{v}) = \prod_{i=1}^{n} L(\theta|v_i)$ is the likelihood function. Given one single training example $\bar{\mathbf{v}}$, the log-likelihood of the RBM model with the parameter θ is given by

$$\ln L(\theta|\bar{\mathbf{v}}) = \ln p(\bar{\mathbf{v}}|\theta) = \ln\left[\frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\bar{\mathbf{v}},\mathbf{h})}\right] = \ln\sum_{\mathbf{h}} e^{-E(\bar{\mathbf{v}},\mathbf{h})} - \ln\sum_{\mathbf{v},\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})},$$

and the gradient is

$$\frac{\partial \ln L(\theta|\bar{\mathbf{v}})}{\partial\theta} = -\sum_{\mathbf{h}} p(\mathbf{h}|\bar{\mathbf{v}})\frac{\partial E(\bar{\mathbf{v}},\mathbf{h})}{\partial\theta} + \sum_{\mathbf{v},\mathbf{h}} p(\mathbf{v},\mathbf{h})\frac{\partial E(\mathbf{v},\mathbf{h})}{\partial\theta}.$$

That is,

$$\frac{\partial \ln L(\theta|\bar{\mathbf{v}})}{\partial w_{ij}} = p(h_i = 1|\bar{\mathbf{v}})\bar{v}_j - \sum_{\mathbf{v}} p(\mathbf{v})p(h_i = 1|\mathbf{v})v_j,$$

$$\frac{\partial \ln L(\theta|\bar{\mathbf{v}})}{\partial b_j} = \bar{v}_j - \sum_{\mathbf{v}} p(\mathbf{v})v_j, \quad \frac{\partial \ln L(\theta|\bar{\mathbf{v}})}{\partial c_i} = p(h_i = 1|\bar{\mathbf{v}}) - \sum_{\mathbf{v}} p(\mathbf{v})p(h_i = 1|\mathbf{v}).$$

In the above gradient expressions, the second terms correspond to the expectations of the gradient of the energy function under the model distribution, and their exact computation calls for a summation over all the visible variables, which is computationally intractable. One way to get around this difficulty is to estimate the second term in the log-likelihood gradient by MCMC simulations, though the MCMC approach can be computationally very expensive. Fischer & Igel (2012, 2014) provide more details.

We may utilize RBMs to build deep BMs (DBMs), where a DBM model consists of multi-hidden layers and only between-layer connections exist. For example, a DBM with two hidden layers can be built by a stack of two RBMs with tied weights. The learning procedure for BMs and RBMs described in this and previous sections can be applied to DBMs (for more details, see Hinton & Salakhutdinov 2012; Salakhutdinov 2015; Salakhutdinov & Hinton 2009, 2012).

6.3. Quantum Training of Boltzmann Machines

As the typical training of BMs relies on MCMC and thus is computationally very expensive or even impossible, we may find that training with quantum resources can be very helpful in reducing the training cost. In fact, the quantum approach in learning with BMs can be more feasible than or preferable to the classical approach. Quantum training techniques have been developed to train classical BMs. Examples include special-purpose quantum computers such as quantum annealers and programmable photonic circuits. In particular, quantum annealing is very suitable for training BMs, and the D-Wave machine, a quantum annealer with thousands of qubits, has been explored for training BMs with deep quantum learning protocols. More details are provided by Adachi & Henderson (2015), Benedetti et al. (2016), and Wiebe et al. (2014).

6.4. Quantum Boltzmann Machines

As in the classical BM case, we adopt a probabilistic graphical model that consists of a set of visible units $\mathbf{v} \in \{-1, 1\}^n$ associated with observations and a set of hidden units $\mathbf{h} \in \{-1, 1\}^m$ used to capture dependencies between observed variables. We use the notation $\mathbf{z} = (\mathbf{v}, \mathbf{h})$ to denote the combined units. A quantum Ising model is defined through its quantum Hamiltonian given by

$$\mathbf{H}_q = \sum_i b_i \boldsymbol{\sigma}_i^z + \sum_{i,j} w_{ij} \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z,$$

where, as in Section 5, we write

$$\boldsymbol{\sigma}_i^z \equiv \underbrace{\mathbf{I} \otimes \dots \otimes \mathbf{I}}_{i-1} \otimes \boldsymbol{\sigma}_z \otimes \underbrace{I \otimes \dots \otimes \mathbf{I}}_{b-i}, \quad \mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{\sigma}_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \text{ and } \ b = m + n.$$

Define the density matrix as $\boldsymbol{\rho} = Z^{-1} e^{-\mathbf{H}_q}$, where $Z = \mathrm{tr}[e^{-\mathbf{H}_q}]$ is the partition function. The ith diagonal element of the density matrix gives the probability associated with the ith state of the quantum Boltzmann machine (QBM). We can derive the marginal distribution of the visible units from the density matrix. For a given visible vector $\mathbf{v} = (v_1, \dots, v_n) \in \{1, -1\}^n$, define a matrix $\Lambda_\mathbf{v} = |\mathbf{v}\rangle\langle\mathbf{v}| \otimes \mathbf{I_h}$, where $\mathbf{I_h}$ is the identical matrix of dimension equal to the number of hidden units. Then, the marginal distribution corresponding to $\mathbf{v}$ of the QBM model is given by $\mathrm{P_v} = \mathrm{tr}[\Lambda_\mathbf{v} \boldsymbol{\rho}]$.

Now we consider adding a transverse field to the Ising Hamiltonian to introduce a transverse field quantum Ising model. Define nondiagonal matrices

$$\boldsymbol{\sigma}_i^x \equiv \underbrace{\mathbf{I} \otimes \dots \otimes \mathbf{I}}_{i-1} \otimes \boldsymbol{\sigma}_x \otimes \underbrace{\mathbf{I} \otimes \dots \otimes \mathbf{I}}_{b-i} \text{ and } \ \boldsymbol{\sigma}_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

The transverse Ising Hamiltonian is then given by

$$\mathbf{H} = \sum_i \Gamma_i \boldsymbol{\sigma}_i^x + \sum_i b_i \boldsymbol{\sigma}_i^z + \sum_{i,j} w_{ij} \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z,$$

where Γ_i, b_i, and w_{ij} are model parameters. Again we define the partition function $Z = \mathrm{tr}[e^{-\mathbf{H}}]$, the density matrix $\boldsymbol{\rho} = Z^{-1}e^{-\mathbf{H}}$, and the marginal distribution of the visible units $\mathrm{P}_{\mathbf{v}} = \mathrm{tr}[\Lambda_{\mathbf{v}}\boldsymbol{\rho}]$. To train the QBM, we employ gradient descent to update the parameters $\theta = (\boldsymbol{\Gamma}, \mathbf{b}, \mathbf{w})$ and minimize the negative log-likelihood function

$$\mathcal{L} = -\sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \log(\mathrm{P}_{\mathbf{v}}),$$

where $\mathrm{P}_{\mathbf{v}}^{\mathrm{data}}$ denotes the empirical distribution of the training set. The gradient of $\mathcal{L}$ is given by

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \left(\frac{\mathrm{tr}[\Lambda_{\mathbf{v}} \frac{\partial e^{-\mathbf{H}}}{\partial \theta}]}{\mathrm{tr}[\Lambda_{\mathbf{v}} e^{-\mathbf{H}}]} - \frac{\mathrm{tr}[\frac{\partial e^{-\mathbf{H}}}{\partial \theta}]}{\mathrm{tr}[e^{-\mathbf{H}}]} \right).$$

As $\mathrm{tr}[\frac{\partial e^{-\mathbf{H}}}{\partial \theta}] = -\mathrm{tr}[\frac{\partial \mathbf{H}}{\partial \theta} e^{-\mathbf{H}}]$, we have

$$\frac{\mathrm{tr}[\frac{\partial e^{-\mathbf{H}}}{\partial \theta}]}{\mathrm{tr}[e^{-\mathbf{H}}]} = -\left\langle \frac{\partial \mathbf{H}}{\partial \theta} \right\rangle,$$

where $\langle \mathbf{A} \rangle \equiv \mathrm{tr}[\boldsymbol{\rho}\mathbf{A}]$ denotes the Boltzmann average of a given matrix $\mathbf{A}$. We may estimate $\langle \frac{\partial \mathbf{H}}{\partial \theta} \rangle$ by sampling from the model. However, the term $\frac{\mathrm{tr}[\Lambda_{\mathbf{v}} \frac{\partial e^{-\mathbf{H}}}{\partial \theta}]}{\mathrm{tr}[\Lambda_{\mathbf{v}} e^{-\mathbf{H}}]}$ cannot be estimated using sampling. Thus, it is almost impossible to find the model parameters by minimizing the negative log-likelihood. We use the variational approach to solve this problem by minimizing an upper bound of the negative log-likelihood.

Let $\mathbf{H}_{\mathbf{v}} \equiv \langle \mathbf{v} | \mathbf{H} | \mathbf{v} \rangle$. We call $\mathbf{H}_{\mathbf{v}}$ the clamped Hamiltonian because every visible qubit $\boldsymbol{\sigma}_i^z$ is clamped to its corresponding classical data value v_i. An application of the Golden–Thompson inequality leads us to $\mathrm{P}_{\mathbf{v}} \geq \frac{\mathrm{tr}[e^{-\mathbf{H}_{\mathbf{v}}}]}{\mathrm{tr}[e^{-\mathbf{H}}]}$, and we then can conclude that

$$\widetilde{\mathcal{L}} = -\sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \log \left(\frac{\mathrm{tr}[e^{-\mathbf{H}_{\mathbf{v}}}]}{\mathrm{tr}[e^{-\mathbf{H}}]} \right)$$

is an upper bound of the negative log-likelihood function $\mathcal{L}$. Instead of minimizing $\mathcal{L}$, we now minimize the upper bound $\widetilde{\mathcal{L}}$ using its gradient,

$$\frac{\partial \widetilde{\mathcal{L}}}{\partial \theta} = \sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \left(\frac{\mathrm{tr}[\Lambda_{\mathbf{v}} \frac{\partial e^{-\mathbf{H}_{\mathbf{v}}}}{\partial \theta}]}{\mathrm{tr}[\Lambda_{\mathbf{v}} e^{-\mathbf{H}_{\mathbf{v}}}]} - \frac{\mathrm{tr}[\frac{\partial e^{-\mathbf{H}}}{\partial \theta}]}{\mathrm{tr}[e^{-\mathbf{H}}]} \right) = \overline{\left\langle \frac{\partial \mathbf{H}_{\mathbf{v}}}{\partial \theta} \right\rangle_{\mathbf{v}}} - \left\langle \frac{\partial \mathbf{H}}{\partial \theta} \right\rangle,$$

where, for a matrix $\mathbf{A}$, we define

$$\overline{\langle \mathbf{A} \rangle_{\mathbf{v}}} = \sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \langle \mathbf{A} \rangle_{\mathbf{v}} = \sum_{\mathbf{v}} \mathrm{P}_{\mathbf{v}}^{\mathrm{data}} \frac{\mathrm{tr}\left[e^{-\mathbf{H}_{\mathbf{v}}\mathbf{A}}\right]}{\mathrm{tr}\left[e^{\mathbf{H}_{\mathbf{v}}}\right]}.$$

Therefore, the updating rules for b_i, w_{ij}, and Γ_i are given as follows:

$$\Delta b_i = \eta \left(\overline{\langle \boldsymbol{\sigma}_i^z \rangle_{\mathbf{v}}} - \langle \boldsymbol{\sigma}_i^z \rangle \right), \ \ \Delta w_{ij} = \eta \left(\overline{\langle \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z \rangle_{\mathbf{v}}} - \langle \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z \rangle \right), \text{ and } \ \Delta \Gamma_i = \eta \left(\overline{\langle \boldsymbol{\sigma}_i^x \rangle_{\mathbf{v}}} - \langle \boldsymbol{\sigma}_i^x \rangle \right),$$

where η is the learning rate of the gradient descent algorithm (Wang & Wu 2020). We can estimate the unclamped terms $\langle \boldsymbol{\sigma}_i^z \rangle$ and $\langle \boldsymbol{\sigma}_i^z \boldsymbol{\sigma}_j^z \rangle$ by sampling from a Boltzmann distribution with Hamiltonian H, and the clamped terms $\overline{\langle \boldsymbol{\sigma}_i^z \rangle_{\mathbf{v}}}$ and $\overline{\langle \boldsymbol{\sigma}_i^z \rangle_{\mathbf{v}}}$ by sampling from a Boltzmann distribution with Hamiltonian $H_{\mathbf{v}}$. However, there is a serious issue regarding estimating $\overline{\langle \boldsymbol{\sigma}_i^x \rangle_{\mathbf{v}}}$ and $\langle \boldsymbol{\sigma}_i^x \rangle$. In fact, we need measurements in the $\boldsymbol{\sigma}_i^x$ basis to estimate $\langle \boldsymbol{\sigma}_i^x \rangle$, but we cannot estimate $\langle \boldsymbol{\sigma}_i^x \rangle$ by sampling in the $\boldsymbol{\sigma}_i^z$ basis, as nondiagonal matrix $\boldsymbol{\sigma}_i^x$ does not commute with $\boldsymbol{\sigma}_i^z$. As a matter of fact, for all visible variables $\mathbf{v}$, we have $\overline{\langle \boldsymbol{\sigma}_i^x \rangle_{\mathbf{v}}} = 0$. Note that $\langle \boldsymbol{\sigma}_i^x \rangle > 0$ for positive Γ_i and $\langle \boldsymbol{\sigma}_i^x \rangle < 0$ for negative Γ_i; thus, we conclude that $\Delta\Gamma_i < 0$ if $\Gamma_i > 0$ and $\Delta\Gamma_i > 0$ if $\Gamma_i < 0$. This renders an invalid training of Γ_i using the updating rule. One possible ad hoc fix is to treat the b_is and w_{ij}s as trainable parameters and regard the Γ_is as superparameters, or even set all the Γ_is to be equal to a fixed value.

As we described in Section 5, quantum annealing driven by the same quantum Hamiltonian $\mathbf{H}$ can be implemented by quantum annealers like the D-Wave machine. It turns out that a quantum annealer can provide a sample from the Boltzmann distribution of the Hamiltonian and train a QBM to tune the model parameters $\theta = (\Gamma_i, b_i, w_{ij})$. More details can be found in the articles by Amin et al. (2018) and Kieferova & Wiebe (2016).

7. CONCLUDING REMARKS

Quantum computation has attracted enormous attention at the frontiers of science. While the main goals of quantum computation are the invention of faster quantum algorithms and the creation of quantum computers to demonstrate quantum advantages and implement quantum algorithms for accomplishing hard computational or communication tasks, this article provides an overview on the statistical aspect of quantum computation to illustrate the interaction between statistics and quantum computation. This stands in contrast to classical computation, where there is little role for statistics to play in its deterministic platform. We introduce important quantum concepts and key quantum properties for quantum computation. We review quantum annealing, quantum machine learning with BMs, and quantum supremacy via boson sampling and random quantum circuits. Our discussion of the selected topics focuses on the use of quantum computation in statistical machine learning and applications of statistical analysis to resolve issues encountered in quantum computation, as well as the interplay between quantum computation and statistics, which may demonstrate quantum advantage and/or lead to new theories, methodologies, and computational techniques for statistics and machine learning. In particular, we present quantum computation and illustrate its interface with statistics and data science, and we highlight the advantages of quantum computation and quantum learning for statistics and machine learning in terms of computational complexity and learning efficiency. There is a great demand for the certification of quantum devices, such as testing and assessing their quantum performance, and such certification needs sound and scalable statistical methods for calibrating and validating quantum properties. In fact, a quantum computation endeavor such as quantum supremacy calls for an integration of new experimental techniques, better mathematical and statistical modeling, and improved computational tools where statistics and data science can play a major role (Hu & Wang 2021; Wang 2012, 2022; Wang & Song 2020; Wang et al. 2016). For example, for the study of quantum supremacy, we need to repeat computing experiments, reanalyze observed data, and address or close potential loopholes. Classical algorithms and computer power continue to be improved, and what is impractical for classical computers today may become tractable in the future. At the same time, the computational power of quantum computers will keep growing. Hence, the benchmark of the classical computational cost is a moving target, and the quantum supremacy frontier will be moving toward larger and larger computational problems to herald a much-anticipated computing paradigm that will ultimately offer a large-scale computational platform to run well-known quantum algorithms, such as the Shor and Grover algorithms. As shown in Section 4, we expect a high demand for statistics in the continuing study of quantum supremacy.

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

The research of Y.W. was supported in part by NSF grants DMS-1528735, DMS-1707605, and DMS-1913149. The authors thank the Editorial Committee for the comments and suggestions that led to improvements to the article.

LITERATURE CITED

Aaronson S, Arkhipov A. 2011. The computational complexity of linear optics. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, pp. 333–42. New York: ACM

Aaronson S, Chen L. 2016. Complexity-theoretic foundations of quantum supremacy experiments. arXiv:1612.05903 [quant-ph]

Adachi SH, Henderson MP. 2015. Application of quantum annealing to training of deep neural networks. arXiv:1510.06356 [quant-ph]

Albash T, Rønnow TF, Troyer M, Lidar DA. 2015. Reexamining classical and quantum models for the D-Wave One processor. *Eur. Phys. J. Spec. Top.* 224(1):111–29

Amin MH, Andriyash E, Rolfe J, Kulchytskyy B, Melko R. 2018. Quantum Boltzmann machine. *Phys. Rev. X* 8(2):021050

Artiles LM, Gill RD, Guţă MI. 2005. An invitation to quantum tomography. *J. R. Stat. Soc. Ser. B* 67(1):109–34

Arute F, Arya K, Babbush R, Bacon D, Bardin JC, et al. 2019. Quantum supremacy using a programmable superconducting processor. *Nature* 574(7779):505–10

Barndorff-Nielsen OE, Gill RD, Jupp PE. 2003. On quantum statistical inference. *J. R. Stat. Soc. Ser. B* 65(4):775–804

Benedetti M, Realpe-Gómez J, Biswas R, Perdomo-Ortiz A. 2016. Estimation of effective temperatures in quantum annealers for sampling applications: a case study with possible applications in deep learning. *Phys. Rev. A* 94(2):022308

Bertsimas D, Tsitsiklis J. 1993. Simulated annealing. *Stat. Sci.* 8(1):10–15

Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N, Lloyd S. 2017. Quantum machine learning. *Nature* 549(7671):195

Boixo S, Isakov SV, Smelyanskiy VN, Babbush R, Ding N, et al. 2018. Characterizing quantum supremacy in near-term devices. *Nat. Phys.* 14(6):595

Boixo S, Rønnow TF, Isakov SV, Wang Z, Wecker D, et al. 2014. Evidence for quantum annealing with more than one hundred qubits. *Nat. Phys.* 10(3):218

Boixo S, Smelyanskiy VN, Shabani A, Isakov SV, Dykman M, et al. 2016. Computational multiqubit tunnelling in programmable quantum annealers. *Nat. Commun.* 7:10327

Bouland A, Fefferman B, Nirkhe C, Vazirani U. 2018. Quantum supremacy and the complexity of random circuit sampling. arXiv:1803.04402 [quant-ph]

Brady LT, van Dam W. 2016. Quantum Monte Carlo simulations of tunneling in quantum adiabatic optimization. *Phys. Rev. A* 93(3):032304

Brooke J, Bitko D, Aeppli G. 1999. Quantum annealing of a disordered magnet. *Science* 284(5415):779–81

Cai T, Kim D, Wang Y, Yuan M, Zhou HH. 2016. Optimal large-scale quantum state tomography with Pauli measurements. *Ann. Stat.* 44(2):682–712

Ciliberto C, Herbster M, Ialongo AD, Pontil M, Rocchetto A, et al. 2018. Quantum machine learning: a classical perspective. *Proc. R. Soc. A* 474(2209):20170551

Dunjko V, Briegel HJ. 2018. Machine learning & artificial intelligence in the quantum domain: a review of recent progress. *Rep. Prog. Phys.* 81(7):074001

Farhi E, Goldstone J, Gutmann S. 2002. Quantum adiabatic evolution algorithms versus simulated annealing. arXiv:quant-ph/0201031

Farhi E, Goldstone J, Gutmann S, Lapan J, Lundgren A, Preda D. 2001. A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. *Science* 292(5516):472–75

Farhi E, Goldstone J, Gutmann S, Sipser M. 2000. Quantum computation by adiabatic evolution. arXiv:quant-ph/0001106

Fischer A, Igel C. 2012. An introduction to restricted Boltzmann machines. In *CIARP 2012: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, ed. L Alvarez, M Mejail, L Gomez, J Jacobo, pp. 14–36. New York: Springer

Fischer A, Igel C. 2014. Training restricted Boltzmann machines: an introduction. *Pattern Recognit.* 47:25–39

Hamilton CS, Kruse R, Sansoni L, Barkhofen S, Silberhorn C, Jex I. 2017. Gaussian boson sampling. *Phys. Rev. Lett.* 119(17):170501

Harrow AW, Montanaro A. 2017. Quantum computational supremacy. *Nature* 549(7671):203

Hinton G, Salakhutdinov R. 2012. A better way to pretrain deep Boltzmann machines. In *Advances in Neural Information Processing Systems 25*, ed. F Pereira, CJC Burges, L Bottou, KQ Weinberger, pp. 2447–55. Red Hook, NY: Curran

Holevo AS. 2001. *Statistical Structure of Quantum Theory*. New York: Springer

Hu J, Wang Y. 2021. Quantum annealing via path-integral Monte Carlo with data augmentation. *J. Comput. Graph. Stat.* 30:284–96

Isakov SV, Mazzola G, Smelyanskiy VN, Jiang Z, Boixo S, et al. 2016. Understanding quantum tunneling through quantum Monte Carlo simulations. *Phys. Rev. Lett.* 117(18):180402

Jörg T, Krzakala F, Kurchan J, Maggs AC. 2010. Quantum annealing of hard problems. *Prog. Theor. Phys. Suppl.* 184:290–303

Kieferova M, Wiebe N. 2016. Tomography and generative data modeling via quantum Boltzmann training. arXiv:1612.05204 [quant-ph]

Kirkpatrick S, Gelatt CD, Vecchi MP. 1983. Optimization by simulated annealing. *Science* 220(4598):671–80

Lund A, Bremner MJ, Ralph T. 2017. Quantum sampling problems, BosonSampling and quantum supremacy. *NPJ Quantum Inform.* 3(1):15

Markov IL, Fatima A, Isakov SV, Boixo S. 2018. Quantum supremacy is both closer and farther than it appears. arXiv:1807.10749 [quant-ph]

McGeoch CC. 2014. Adiabatic quantum computation and quantum annealing: theory and practice. *Synth. Lect. Quantum Comput.* 5(2):1–93

Neill C, Roushan P, Kechedzhi K, Boixo S, Isakov SV, et al. 2018. A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science* 360(6385):195–99

Nielsen MA, Chuang IL. 2010. *Quantum Computation and Quantum Information*. Cambridge, UK: Cambridge Univ. Press. 10th ed.

Parthasarathy KR. 2012. *An Introduction to Quantum Stochastic Calculus*. Basel, Switz.: Birkhäuser

Petz D. 2008. *Quantum Information Theory and Quantum Statistics*. New York: Springer

Quesada N, Arrazola JM, Killoran N. 2018. Gaussian boson sampling using threshold detectors. *Phys. Rev. A* 98(6):062322

Rinott Y, Shoham T, Kalai G. 2020. Statistical aspects of the quantum supremacy demonstration. arXiv:2008.05177 [quant-ph]

Rønnow TF, Wang Z, Job J, Boixo S, Isakov SV, et al. 2014. Defining and detecting quantum speedup. *Science* 345(6195):420–24

Sakurai J, Napolitano J. 2017. *Modern Quantum Mechanics*. Cambridge, UK: Cambridge Univ. Press

Salakhutdinov R. 2015. Learning deep generative models. *Annu. Rev. Stat. Appl.* 2:361–85

Salakhutdinov R, Hinton G. 2009. Deep Boltzmann machines. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, ed. D van Dyk, M Welling, pp. 448–55. Brookline, MA: Microtome

Salakhutdinov R, Hinton G. 2012. An efficient learning procedure for deep Boltzmann machines. *Neural Comput.* 24(8):1967–2006

Shankar R. 2012. *Principles of Quantum Mechanics*. New York: Springer

Sinervo PK. 2003. Definition and treatment of systematic uncertainties in high energy physics and astrophysics. In *Proceedings of the Conference on Statistical Problems in Particle Physics, Astrophysics and Cosmology (PHYSTAT2003)*, ed. L Lyons, R Mount, R Reitmeyer, pp. 122–29. Stanford, CA: SLAC

Sutton RS, Barto AG. 2018. *Reinforcement Learning: An Introduction*. Cambridge, MA: Bradford. 2nd ed.

Wang Y. 2012. Quantum computation and quantum information. *Stat. Sci.* 27(3):373–94

Wang Y. 2022. When quantum computation meets data science: making data science quantum. *Harv. Data Sci. Rev.* In press

Wang Y, Song X. 2020. Quantum science and quantum technology. *Stat. Sci.* 35(1):51–74

Wang Y, Wu S. 2020. Asymptotic analysis via stochastic differential equations of gradient descent algorithms in statistical and computational paradigms. *J. Mach. Learn. Res.* 21(199):1–103

Wang Y, Wu S, Zou J. 2016. Quantum annealing with Markov chain Monte Carlo simulations and D-Wave quantum computers. *Stat. Sci.* 31(3):362–98

Wang Y, Xu C. 2015. Density matrix estimation in quantum homodyne tomography. *Stat. Sin.* 3:953–73

Wiebe N, Kapoor A, Svore KM. 2014. Quantum deep learning. arXiv:1412.3489 [quant-ph]

Wittek P. 2014. *Quantum Machine Learning: What Quantum Computing Means to Data Mining*. New York: Academic

Zhong HS, Wang H, Deng YH, Chen MC, Peng LC, et al. 2020. Quantum computational advantage using photons. *Science* 370(6523):1460–63