*Statistical Science*
2022, Vol. 37, No. 3, 322–347
https://doi.org/10.1214/21-STS836

# Statistical Aspects of the Quantum Supremacy Demonstration

**Yosef Rinott, Tomer Shoham and Gil Kalai**

*Abstract.* In quantum computing, a demonstration of *quantum supremacy* (or quantum advantage) consists of presenting a task, possibly of no practical value, whose computation is feasible on a quantum device, but cannot be performed by classical computers in any feasible amount of time. The notable claim of quantum supremacy presented by Google's team in 2019 consists of demonstrating the ability of a quantum circuit to generate, albeit with considerable noise, bitstrings from a distribution that is considered hard to simulate on classical computers. Very recently, in 2020, a quantum supremacy claim was presented by a group from the University of Science and Technology of China, using a different technology and generating a different distribution, but sharing some statistical principles with Google's demonstration.

Verifying that the generated data is indeed from the claimed distribution and assessing the circuit's noise level and its fidelity is a statistical undertaking. The objective of this paper is to explain the relations between quantum computing and some of the statistical aspects involved in demonstrating quantum supremacy in terms that are accessible to statisticians, computer scientists, and mathematicians. Starting with the statistical modeling and analysis in Google's demonstration, which we explain, we study various estimators of the fidelity, and different approaches to testing the distributions generated by the quantum computer. We propose different noise models, and discuss their implications. A preliminary study of the Google data, focusing mostly on circuits of 12 and 14 qubits is given in different parts of the paper.

*Key words and phrases:* Google's quantum computer, random distributions, estimation of sampling weights, size bias.

## 1. INTRODUCTION

Google's announcement of quantum supremacy [3] was compared by various writers to landmark achievements such as the Wright brothers' invention of a motor-operated airplane, Fermi's demonstration of a nuclear chain reaction, and the discovery of the Higgs boson. It was also met with some skepticism, which is partly due to the fact that Google's quantum computer, and quantum computers in general, have not at this point performed any practical task (such as factoring large integers). Instead, Google's quantum computer performs a *sampling task*; that is, it generates random bitstrings, with considerable noise, from a discrete distribution supported on $M$ values, with probabilities whose computations are far beyond the reach of classical computers for large $M$. (Therefore, classical computers cannot carry out this sampling task.) A quantum computer, suitably adjusted, is capable of sampling from such a distribution without having to compute the probabilities explicitly, and thus could claim quantum supremacy. Google's team estimated that the task performed by their quantum circuit would take 10,000 years on classical computer. In a recent quantum advantage demonstration, see [23], a team from the University of Science and Technology of China (USTC) claimed that the sampling task computed by their quantum computer, which differs from Google's, would take 2.5 billion years to perform on a classical supercom-

*Yosef Rinott is Professor, Federmann Center for the Study of Rationality and Department of Statistics, The Hebrew University of Jerusalem, Israel (e-mail: yosef.rinott@mail.huji.ac.il). Tomer Shoham is a graduate student, Federmann Center for the Study of Rationality and Department of Statistics, The Hebrew University of Jerusalem, Israel (e-mail: tomer.shohamm@gmail.com). Gil Kalai is Professor, Einstein Institute of Mathematics and Federmann Center for the Study of Rationality, The Hebrew University of Jerusalem, Israel and Efi Arazy School of Computer Science, IDC, Herzliya, Israel (e-mail: gil.kalai@gmail.com).*

puter![1] Both Google's and USTC's quantum computers required about 200 seconds for their tasks.

Verifying that the generated data indeed have the claimed distribution in spite of much noise is a statistical problem, which we explain and generalize, offering some new statistical aspects. We focus on the statistical aspects of Google's experiment without aiming to pass judgment on its merits. Parts of the discussion below are quite general and could shed some light on other existing or future noisy intermediate-scale quantum computer experiments. Aiming for a general mathematical audience, we briefly review some basic relevant statistical ideas and some of the basics of quantum computing, and therefore parts of the paper are of an expository nature. Also, our notation is a compromise between Google's notation [3] and more standard statistical notation. We start with the main statistical models and problems; background on quantum computing is given in Section 2.

### 1.1 The Main Model and Statistical Problem

We provide a brief schematic view of the nature of quantum computers by describing Google's quantum computer. More details are given in Section 2.

A quantum computer (or circuit) consists of $n$ qubits, which are its basic memory units. The computer operates via *gates* operating on one and two qubits. In Google's case, qubits are realized as coupled (linked) chips that become superconducting at cold temperatures, resulting in reduced noise and dissipation. A qubit can exist at two excitation levels, that is, energy levels that depend on the orbitals of electrons in the chip, and thus when read/measured, each qubit takes the value 0 or 1 and the whole system yields an $n$-vector of 0's and 1's to which we refer as *bitstring*. Before being measured the system exists in a *superposition* of all $2^n$ $n$-vectors, which means that all $2^n$ $n$-vectors are represented potentially in the system, and only when *measured* (observed) it "collapses" to some $n$-vector, chosen with a certain probability. The system operates with a great deal of noise due to errors in the qubits and the gates, and to *readout errors* (of the qubits) at the end of the process, affecting the distribution of the observed vectors. When the circuit is set, by setting its gates, it determines a set of $M := 2^n$ probabilities $\{w_{\mathbf{x}^{(i)}} : \mathbf{x}^{(i)} \in \{0, 1\}^n\}$ such that when the system is measured, it will yield the vector $\mathbf{x}^{(i)}$ with probability $w_{\mathbf{x}^{(i)}}$ for $i = 1, \dots, M$, where $\sum_{i=1}^{M} w_{\mathbf{x}^{(i)}} = 1$, provided it runs without errors. (We may denote $w_{\mathbf{x}^{(i)}}$ by $w_i$, and $\mathbf{x}^{(i)}$ by $\mathbf{x}_{w_i}$.) If we run the system and measure it $N$ times, we get $N$ such $n$-vectors, and denoting the probability of no error, known as the *fidelity*, by $\phi$, our sample $\mathcal{S}_\mathbf{x}$ will consist of $N$ i.i.d. vectors $\mathbf{x}^{(i)}$, sampled with probability denoted by $\pi(\mathbf{x}^{(i)})$ or $\pi(w_i)$ satisfying

$$\pi(\mathbf{x}^{(i)}) = \pi(w_i) = \phi w_i + (1 - \phi)/2^n; \tag{1.1}$$

that is, the vector $\mathbf{x}^{(i)}$ is chosen with probability $w_i$ in the event that no error occurs (whose probability is $\phi$), and if an error occurs (with probability $1 - \phi$), the vector is chosen uniformly. This is the sampling model proposed in [3]. It describes the sampling task distribution as a mixture of the desired distribution $w_i$ and a uniform distribution on the space of $\{0, 1\}^n$. We shall discuss this model later, along with generalizations of the error model. We denote the $N$ sampled values by $\mathcal{S}_\mathbf{x} = \{\widetilde{\mathbf{x}}^{(j)}\} = \{\mathbf{x}_{\widetilde{w}_j}\}$, and by $\mathcal{S}_w$ the sample $\{\widetilde{w}_j\}$ of the probabilities associated with the sampled vectors in $\mathcal{S}_\mathbf{x}$. Note that $\mathcal{S}_\mathbf{x}$ and $\mathcal{S}_w$ are *multisets*, allowing multiple instances of elements as in i.i.d. sampling or sampling with replacement.

For Google's quantum computer, one can assume that the probabilities $w_i$ of a randomly chosen quantum circuit (to be discussed later) are random variables generated as follows. Let $z_i$ be i.i.d. Exponential(1) variables (with density $e^{-z}$ for $z > 0$), $i = 1, \dots, M = 2^n$, and $w_i = z_i / \sum_{j=1}^{M} z_j$. It is well known that $(w_1, \dots, w_M)$ has the Dirichlet($\boldsymbol{\alpha}$) distribution with parameter $\boldsymbol{\alpha} = \mathbf{1}$, an $M$-vector whose components are all equal to 1. This is a uniform distribution over the $(M - 1)$-dimensional standard simplex. For most purposes only a few moments of the distribution will be taken into account, and our study applies more generally. Clearly $Ew_i = 1/M$, and more generally, we shall need the following facts:

$$\begin{aligned} Ew_i^k &= k!/\big[M \cdots (M + k - 1)\big], \\ E(w_i w_j) &= 1/\big[M(M + 1)\big] \quad \text{for } i \neq j; \end{aligned} \tag{1.2}$$

see, for example, [13]. The distribution of $\{\widetilde{w}_j\}$ differs from that of $\{w_j\}$. Under the sampling model (1.1), with probability $\phi$, $w_j$ is sampled with a probability proportional to $w_j$. Such sampling is known as *size-biased sampling*; see Section 3.

In [3], the sample size is denoted by $N_s$, which we abbreviate to $N$; Google's notation for the fidelity parameter is $F$, which we denote by $\phi$, and their notation for the random probabilities $w_i$ is $\mathcal{P}(\mathbf{x}^{(i)})$, known as a Porter–Thomas distribution. We sometimes use the notation $\mathcal{P}_C(\mathbf{x}^{(i)})$ to emphasize the dependence of the probabilities on the (random) circuit $C$.

The statistical problems that arise in relation to the model of (1.1) and its extensions described below include estimating of the parameter $\phi$ and verifying that $\phi > 0$, as $\phi = 0$ indicates that the circuit produces pure noise, and testing the validity of the model (1.1) and its variations under different assumptions.

When the $w$'s are considered non-random, possibly by conditioning, this can be seen as sampling from a discrete (finite) population. Recalling that the $w$'s are generated

[1] However, the method of [11] may lead to an efficient classical sampling algorithm for this task.

randomly from a given distribution, we say that the sample is generated from a random discrete distribution in the sense of Kingman [12]. See Section 3 for further details. The estimation takes advantage of both the process generating the $w$'s, and the special nature of the sampling scheme (1.1) or variants of it.

We consider two kinds of analyses of estimators. First, we condition on $\{w_i\}_{i=1}^M$, which amounts to considering a particular quantum circuit, and sampling from the fixed set $\{w_i\}_1^M$. We study conditional properties of different estimators, such as their bias and variance. Second, assuming that the $w_i$'s are random and satisfy some moment conditions, we study properties of estimators when averaged over the randomness of $\{w_i\}_1^M$. We compare the two analyses and discuss them in Section 4. Google's estimator for the fidelity is quite simple and given by

$$U := 2^n \frac{1}{N} \sum_{j=1}^{N} \mathcal{P}_C(\tilde{\mathbf{x}}^{(j)}) - 1. \tag{1.3}$$

The estimator $U$ is nearly unbiased (see (4.7)) when both the sample $\{\tilde{\mathbf{x}}^{(j)}\}$ and the probabilities $w_i = \mathcal{P}(\mathbf{x}^{(i)})$ are random and expectation is taken over both. However, when considering fixed probabilities $w_i$, the estimator $U$ is not unbiased.

In Sections 4.2 and 4.4, respectively, we discuss an unbiased version $V$ of $U$, and the *maximum likelihood estimator* (MLE), which is nearly unbiased. Both turn out to be superior to the above $U$ in terms of variance and bias for both types of experiments: sampling repeatedly from a single circuit, and averaging over several circuits. This superiority decreases when $\phi$ is small and $M$ is large. In particular, the improvement achieved by the estimators $U$ and $V$ is insignificant when the number of qubits is above 30. Nevertheless, it does matter for many current relatively small-scale circuit experiments and to Google's demonstration where small circuits are used for extrapolation arguments to larger circuits; see [3] Figure 4, which starts with $n = 12$. Our statistical study involves theoretical arguments, demonstrated by simulated data and Google's experimental data.

The present statistical setup brings to mind super-population models (see, e.g., [21] Ch. 14.5 or [15]) where a population $\mathscr{P} = \{w_i\}$ of size $M$ of some measurements is considered to be a realization from a continuous or discrete distribution known as a super-population model, and then a sample of size $N$ is taken from $\mathscr{P}$, using a known sampling scheme. A standard goal in this case is to make an inference on the parameters of the population $\mathscr{P}$ using the sample. However, in our case the sampling scheme (1.1) is unknown and our goal is different; instead of estimating the population parameters, we want to estimate the parameter $\phi$, which is part of the sampling scheme.

We add some more details on the connection of the statistical models described above to quantum physics and the Google experiment. (Section 2 gives a detailed explanation.) The (ideal) quantum state of a quantum computer with $n$ qubits is represented by a unit vector $\mathbf{u} = (u_1, u_2, \ldots, u_M)$ in an $M$-dimensional complex vector space, $M = 2^n$. The coordinates of $\mathbf{u}$ are referred to as amplitudes. We cannot probe these amplitudes directly (this follows from Heisenberg's uncertainty principle), but we can measure the state, and this yields a single sample from a discrete probability distribution with probabilities $w_1 = |u_1|^2, w_2 = |u_2|^2, \ldots, w_M = |u_M|^2$.

Next, come random circuits: when the collection of gates is chosen at random then the vector $\mathbf{u}$ behaves like a random unit vector and this implies, when $M$ is large, that the random probabilities $w_1, \ldots, w_M$ are modeled to arise from exponential $z_i$'s normalized by their sum as above. To see this note first that a uniformly distributed $M$-dimensional vector on a sphere of radius 1 can be generated by taking $M$ i.i.d. N(0,1) variables, and normalizing the length to 1, which is obtained approximately (for large $M$) by dividing by $\sqrt{M}$. Therefore, if we consider $\mathbf{u}$ as a unit vector in a real $2M$-dimensional space, the coordinates of $\mathbf{u}$ behave like i.i.d. Gaussian variables, and therefore each squared absolute value of a complex coordinate behaves like the sum of the squares of two i.i.d. Gaussians, which has an exponential distribution. (The sum of the squares of $k$ i.i.d. Gaussians is distributed as $\chi^2(k)$, that is, as a $\chi^2$-distribution with $k$ degrees of freedom. Thus the sum of the squares of two Gaussians has a $\chi^2(2)$ distribution, which coincides with a constant times Exp(1).)

The quantum computer samples $N$ of these $w_i$'s independently according to a model such as (1.1). This is the sampling task. Given a quantum circuit, the computation of the linked probabilities $w_i$ (even a single one of them) can only be done with exponentially increasing efforts by classical computers if $n < 40$ or so, and it becomes a practically impossible task if, say, $n > 50$ [2] (Google's ultimate experiment is with $n = 53$). As classical computers cannot compute these probabilities for large $n$, they cannot produce samples according to them. The quantum computer does not compute these probabilities, and the claimed supremacy is in its ability to perform the sampling task nevertheless, and produce a sample of bitstrings $\mathcal{S}_{\mathbf{x}} = \{\tilde{\mathbf{x}}^{(j)}\}$ according to these unknown probabilities (mixed with uniform probabilities); see [3]. It is important to realize that verifying that the quantum computer indeed performed its task of sampling from the right distribution requires to compute this distribution by a classical computer, which it cannot do for $n = 53$. Therefore,

[2] Here we refer to the most complicated circuits in the Google experiment. There are various simplified circuits for which computing the linked probabilities is feasible.

the proof of quantum supremacy for large $n$ requires various extrapolation arguments based on smaller values of $n$ or simplified circuits, in addition to statistical reasoning.

In [3] it is assumed that the fidelity $\phi$ is known approximately from an independent source, and part of the supremacy proof consists of showing that the sample is indeed generated as described, which requires to estimate $\phi$. We shall assume that for each sampled $\widetilde{\mathbf{x}}^{(j)}$ the associated probability $\widetilde{w}_j$ is known and so the sample $\mathcal{S}_w = \{\widetilde{w}_j\}$ is known. This is required for the computation of any estimator of $\phi$, both Google's and ours, apart from the estimator $T$ of Section 4.7 (which requires large samples). We emphasize that computing the probabilities requires a classical computation and is possible only when $n < 40$ or so.

We now generalize Google's model (1.1) to allow for more elaborate error models discussed later. Consider a realization of $M$ i.i.d. vectors with nonnegative components $\mathbf{z}_i = (z_{1i}, \dots, z_{pi})$, $i = 1, \dots, M$, distributed according to some $p$-dimensional distribution $D$ with marginal distributions $D_k$. For example, the vectors' components $z_{ki}$ can be independent, and $z_{ki} \sim D_k$, $k = 1, \dots, p$. For $i = 1, \dots, M = 2^n$, set

$$(1.4)\qquad \begin{aligned} &\mathbf{w}_i = (w_{1i}, \dots, w_{pi}), \quad \text{where} \\ &w_{ki} = z_{ki} / \sum_{j=1}^{M} z_{kj}, \quad \text{so} \\ &\sum_{i=1}^{M} w_{ki} = 1, \quad k = 1, \dots, p, \end{aligned}$$

and thus for each $k$, the vector $(w_{k1}, \dots, w_{kM})$ is a *random probability* vector, which has the Dirichlet($\mathbf{1}$) distribution if $z_{ki} \sim \mathrm{Exp}(1)$ i.i.d. for $i = 1, \dots, M$. Each vector $\mathbf{w}_i$ is associated with a vector $\mathbf{x}_{\mathbf{w}_i} \equiv \mathbf{x}^{(i)} \in \{0, 1\}^n$. The sampling task described next can be expressed in terms of both $\mathbf{w}_i$ and $\mathbf{x}_{\mathbf{w}_i}$.

A random sample of size $N$, $\mathbf{x}_{\widetilde{\mathbf{w}}_1}, \dots, \mathbf{x}_{\widetilde{\mathbf{w}}_N}$, or equivalently $\widetilde{\mathbf{w}}_1, \dots, \widetilde{\mathbf{w}}_N$, is drawn (with replacement) from the above set of $M$ vectors $\mathbf{x}_{\mathbf{w}_i}$, or equivalently from $\mathbf{w}_i$, where draws are independent, and in each draw the probability $\pi(\mathbf{x}_{\mathbf{w}_i}) \equiv \pi(\mathbf{w}_i)$ of drawing $\mathbf{x}_{\mathbf{w}_i}$ (or $\mathbf{w}_i$) is

$$(1.5)\qquad \begin{aligned} \pi(\mathbf{x}_{\mathbf{w}_i}) &\equiv \pi(\mathbf{w}_i) \\ &= \phi_1 w_{1i} + \cdots + \phi_p w_{pi}, \quad i = 1, \dots, M, \end{aligned}$$

where $\phi_1, \dots, \phi_p$ are nonnegative parameters satisfying $\phi_1 + \cdots + \phi_p = 1$, and so $\sum_{i=1}^{M} \pi(\mathbf{w}_i) = 1$. Google's model (1.1) with the above assumptions corresponds to (1.5) with $p = 2$, $z_{ki} \sim \mathrm{Exp}(1)$ i.i.d. variables for $k = 1$, and for $k = 2$, $z_{ki}$ is identically equal to 1. Again, note that the sampled $\widetilde{\mathbf{w}}_j$'s do not have the same distribution as the original $\mathbf{w}_j$'s due to the sampling scheme that assigns higher probabilities to vectors with larger components. We denote the sample by

$$(1.6)\quad \mathcal{S}_w = \{\widetilde{\mathbf{w}}_j\} = \{(\widetilde{w}_{1j}, \dots, \widetilde{w}_{pj})\}, \quad j = 1, \dots, N,$$

and we use $\mathcal{S}_{\mathbf{x}} = \{\mathbf{x}_{\widetilde{\mathbf{w}}_j}\}_1^N = \{\widetilde{\mathbf{x}}^{(j)}\}_1^N$, to denote the sample, which is a multiset. Such notation is required because the random sampling is from a finite population that is itself random, and $w_{kj}$ and $\widetilde{w}_{kj}$ are both random, with different distributions. This sampling scheme is a mixture of size-biased sampling where in each of $N$ rounds, a coordinate of $\mathbf{w}_i$ is chosen, where $\phi_k$ is the probability of the $k$th coordinate, and then $\mathbf{w}_i$ is chosen with a probability proportional to the size of the chosen coordinate. Section 3 provides for more details on size bias.

Our model (1.5) is relevant for detailed modeling of the Google experiment. For a general noise model, we consider $p$ possible events describing a certain malfunction in the experiment leading to a noise distribution $w_{ki} \sim D_k$ with probability $\phi_k$. In the Google experiment based on random quantum circuits, it is reasonable to assume that these $w_{ki}$'s are approximately statistically independent.

In the general setup of (1.5)–(1.6), we study the following statistical problem: having observed the sampled vectors $\widetilde{\mathbf{w}}_1, \dots, \widetilde{\mathbf{w}}_N$, one goal is to estimate $\phi_1, \dots, \phi_p$, which together with $\mathbf{w}_i$ determine the sampling scheme (1.5). Furthermore, we want to test the hypothesis that the sample is indeed generated according to the model (1.5), and also that the $\mathbf{w}_i$'s are generated according to the given model, which is Dirichlet distribution in Google's model (1.1).

### 1.2 How Can Such Experiments Be Confirmed or Refuted?

A scientific (and technological) claim such as quantum supremacy, which has significant implications to understanding the nature of functions that can be calculated by an effective method (see [16], Sections 1.1.1 and 3.2.2) brings up the question of how can the evidence in the experiment be evaluated, confirmed or refuted. Since Google's quantum computer Sycamore is unique (and rather expensive), it is difficult to test it independently. This will probably hold for any similar experiments.

We propose the following protocol for evaluation of Google's experiment; we believe that understanding its rationale may elucidate the complexities involved in proving quantum supremacy. Independent scientists will prepare several programs (circuits) for Sycamore to be run with about $n = 40$ qubits. Computing the sampling probabilities $\{w_i\}$ of these circuits should be a task that takes several months (on a classical computer). These programs will be sent to Google for implementation. As the implementation may somewhat change the programs due to calibration (see Section 2.3), Google will send back the implemented programs, and large samples that they produce in a short time, which is assumed to preclude computation of the relevant $\{w_i\}$ of the implemented programs. Using classical computers the scientists will take their time and compute the set $\{w_i\}$ for each implemented programs.

They will then evaluate the relation between those $\{w_i\}$'s and the samples they received. Such a protocol is likely to be relevant to other quantum supremacy demonstrations which are being pursued very actively these days.

Of course, replications of the experiment for random quantum circuits in the regime of 10–60 qubits, via Sycamore and and via other quantum computers, and larger sample sizes than given so far in [3] are necessary and valuable even if they do not follow the strict protocol above, and will give a better opportunity to examine the noise modeling.

### 1.3 Paper Outline

Some background on quantum computing is given in Section 2. In Section 3, we discuss size-biased distributions and random discrete distributions, and the implications of testing goodness of fit to the size-biased sampling distribution. In Section 4, we concentrate on the case of $p = 2$ and compare various statistical methods for estimating the fidelity. More precisely, we analyze Google's estimator $U$, as well as two unbiased estimators: $V$, which is a variant of $U$ that is unbiased for any given realization $\{w_i\}_1^M$, and the maximum likelihood estimator, denoted by MLE. We study robustness properties of estimator $U$ and related estimators, and consider a new estimator, $T$, of a different nature. The results are demonstrated briefly by simulated data, and by Google's data. In Section 5, we briefly consider estimation in the case of general $p$. In Section 6, we propose more detailed noise models for the Google sample based on the analysis of readout errors, and analyze statistical estimators based on our readout noise models. Confidence intervals for estimated parameters are discussed in Section 7. In Section 8 we briefly address the question of testing goodness of fit of various empirical distributions to the theoretical ones. A preliminary study of Google's data on small circuits, supporting our findings on the fidelity estimators and the relevance of our readout noise models, is presented in Sections 4 and 6. Yet, neither Google's basic noise model nor our refined readout error model fit the observed data (Section 8). Section 9 concludes the paper, apart from two proofs given in the Appendix.

## 2. QUANTUM COMPUTERS AND GOOGLE'S QUANTUM SUPREMACY EXPERIMENT

### 2.1 Quantum Computers

In this section, we provide some background on quantum computing, sampling algorithms, and Google's experiment. The reader is referred to [16] and [22] for further information. The latter reference provides more math and physics details, and an updated summary of quantum information and computation, including various potential statistical applications.

Quantum computers are physical devices that are believed to have the potential for solving certain computational tasks that are well beyond the ability of classical computers. Shor's famous algorithm shows that if a suitable quantum computer could operate with a sufficiently low level of noise, it could factor $n$-digit integers efficiently in roughly $n^2$ computational steps! The best-known classical algorithms require an exponential number of steps in $n^{1/3}$. This ability for efficient factoring would allow quantum computers to break the majority of current cryptosystems.

A *sampling task* is one where the computer (either quantum or classical) produces samples from a certain probability distribution $\pi$. In the main example of Google's experiment paper [3] each sample is a 0–1 vector of length 53, where $\pi$ is a probability distribution on such vectors. It has been hypothesized that quantum algorithms allow sampling from probability distributions well beyond the capabilities of classical computers.

Quantum systems are inherently noisy; we cannot accurately control them, and any interaction with them introduces further noise. A noisy quantum computer has the property that at every computational step (applying a gate, measuring a qubit) the computer makes an error with a certain small probability. *Noisy intermediate-scale quantum (NISQ)* computers are quantum computers where the number of qubits is in the tens or at most in the hundreds. Over the past decade researchers conjectured that the huge computational advantage of sampling with quantum computers can be realized by NISQ computers that only approximate the target probability distribution, and predicted that this could lead to a demonstration of quantum computational supremacy. NISQ computers are also crucial to the task of creating good-quality quantum error-correction, which is a necessary ingredient for larger-scale quantum computers.

An important feature of NISQ systems – especially for the tasks of quantum supremacy – is the fact that a single error in the computation sequence has a devastating effect on the outcome. In the NISQ regime, the engineering challenge is to keep the computation error-free. The probability that not even a single error occurs is defined as the *fidelity*. Many companies and research groups worldwide are investing in attempts to implement quantum computations via NISQ computers (as well as by other means). Statistical tools from [3] (and this paper) are relevant to the study of fidelity and models for noise of NISQ circuits that go beyond Google's experiment.

### 2.2 A Little More on Quantum Computers

We give a brief overview of quantum computers. A qubit is a piece of quantum memory. The state of a qubit can be described by a unit vector in a two-dimensional complex Hilbert space $\mathcal{H}$. For example, a basis for $\mathcal{H}$ can correspond to two energy levels of the hydrogen atom, or to horizontal and vertical polarizations of a photon. Quantum mechanics allows the qubit to be in a *superposition* of the basis vectors, described by an arbitrary unit vector in $\mathcal{H}$, and the squares of the real and complex parts of this vector represent the probabilities of the qubit to be read as 0 or 1. The memory of a quantum computer (quantum circuit), denoted by $C$, consists of $n$ qubits. Let $\mathcal{H}_k$ be the two-dimensional Hilbert space associated with the $k$th qubit. The state of the entire memory of $n$ qubits is described by a unit vector in the tensor product $\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_n$. We can put one or two qubits through gates, acting on the corresponding two- or four-dimensional Hilbert spaces, and as for classical computers, a small list of gates is sufficient for universal quantum computing. Applying a gate amounts to applying a unitary operator on the two- or four-dimensional space that corresponds to the qubits involved in the gate, tensored with the identity transformation on all other qubits. (Applying a gate thus represents a unitary transformation on the large $2^n$-dimensional Hilbert space that represents all the qubits.) The system performs a given number of cycles, where each *cycle* in the Google experiment consists of applying in parallel single-qubit gates on all qubits followed by two-qubit gates on non-overlapping pairs of qubits. The final state of the quantum computer is a unit vector $\mathbf{u} = (u_1, \dots, u_M)$ in $\mathbb{C}^M$, where the $M = 2^n$ indexes correspond to all 0-1 vectors of length $n$. As already mentioned, at the end of the computation process, the state of the entire computer is measured, giving a sample from the probability distribution $\mathcal{P}_C$ on 0-1 vectors of length $n$.

A few words on the connection between the mathematical model of quantum circuits and quantum physics: in quantum physics, states and their evolutions (the way they change in time) are governed by the Schrödinger equation. A solution of the Schrödinger equation can be described as a unitary process on a Hilbert space and quantum computing processes like the ones we just described form a large class of such quantum evolutions. An interesting question is what precisely gives quantum computers their superior computing power. The state of the quantum computer can be a huge superposition that requires an exponential number of "amplitudes," and therefore, quantum computers allow for specific algorithmic tasks with massive parallelism. Another related fact is that quantum probabilities (unlike classical probabilities) can be both positive and negative, allowing massive cancellations.

### 2.3 The Google Supremacy Claim

The Google experiment is based on the building of a quantum computer (circuit) $C$ with $n$ qubits that perform $m$ cycles of computations, that is, rounds of parallel operation of a set of gates. At the end of the computation the qubits are measured, leading to a sample from a probability distribution $\mathcal{P}_C$ on 0-1 vectors $\mathbf{x}$ of length $n$. This process is repeated $N$ times (for the same circuit $C$) to produce a sample of size $N$. For the ultimate experiment ($n = 53$, with 1113 1-qubit gates, and 430 2-qubit gates, and $m = 20$ cycles) the quantum computer produced a sample of several million 0-1 vectors of length 53.

The specific circuit $C$ is itself a random circuit. For every experiment, specific gates are chosen at random (by a classical computer) and are fixed, thus determining (programming) the circuit. In parts of Google's experiments (which produced most of the data), an additional *calibration* was necessary after fixing the circuit, which resulted in modifying some gates and the associated circuit's probabilities. In ideal situations, namely, without noise, the quantum computer would produce samples from a certain probability distribution $\mathcal{P}_C$ that depends on the specific circuit $C$. When the circuit $C$ is chosen at random, the probability distribution $\mathcal{P}_C$ looks like an instance of the random distribution (a Porter–Thomas distribution) described in the introduction.

Google's quantum computer is "noisy"; the size $N$ sample it produces is modeled as follows: a fraction $\phi$ of the samples are from $\mathcal{P}_C$ and a fraction $(1 - \phi)$ of the samples are from a uniform distribution (or uniform noise), and $\phi$ is referred to as the *fidelity*. This corresponds to (1.1). We shall generalize this model in the sequel. Google's paper made two crucial claims regarding the ultimate 53-qubit samples.

(A) The fidelity $\phi$ of their sample is above $1/1000$.

(B) Producing a sample with a similar fidelity would require 10,000 years on a supercomputer.

As it was only possible to give indirect evidence for both these claims, we shall now describe the logic of Google's quantum supremacy argument.

For claim (A) regarding the value of $\phi$, the paper describes a statistical estimator of $\phi$ and the argument relies on a bold extrapolation argument that has two ingredients. One ingredient is a few hundred experiments in the classically tractable regime: the regime where the probability distribution $\mathcal{P}_C$ can be computed by a classical computer and the performance of the quantum computer can be tested directly. The other ingredient is a theoretical formula for computing the fidelity. According to the Supplement to Google's paper [3], the fidelity of entire circuits closely agrees with the prediction of the simple mathematical formula (Formula (77) in the Supplement to [3], Equation (2.1) below). There are around 200 reported experiments in the classically tractable regime. These experiments support the claim that the prediction given by Formula (77) for the fidelity is indeed very robust and therefore may apply also to the 53-qubit circuit in the supremacy regime. To test whether the quantum computer produced a sample with the hoped-for properties we need to be able to simulate the quantum computer on a classical supercomputer. This is beyond reach for 53 qubits and therefore the samples for the 53-qubit experiments demonstrating "supremacy" are archived, but it is not possible to test them in any direct way.

For claim (B) regarding the difficulty for classical computers to compute the sampling distribution of a given quantum circuit, the Google team relies on extrapolation from the running time of a specific algorithm they use and on results and conjectures from computational complexity that support the assertion that the sampling task at hand is hard, requiring an exponential number (in $n$) of computational steps on classical computers; see Section XI in the Supplement to [3] and [1].

A team from IBM [18] found a classical algorithm that would require only several days of computation on a classical supercomputer (which is less than 10,000 years and more than the running time of the quantum computer of 200 seconds). IBM's and Google's algorithms computes the entire $M$ probabilities $\mathcal{P}_C(\mathbf{x})$ described by the circuit. Further classical algorithms for performing tasks similar to Google's that might be in conflict with the supremacy claims are given in [9] and [17]. Specifically, the recent paper [17] announces a classical method for the task of producing a sample with fidelity $> 1/1000$ for $n = 53$.

## 2.4 On Fidelity

The Google argument relies crucially on the following simple formula (Equation (77) in the Supplement to [3]) for estimating the fidelity $\phi$ of their experiments:

$$(2.1)\qquad \phi \approx \prod_{g\in\mathcal{G}_1}(1-e_g)\prod_{g\in\mathcal{G}_2}(1-e_g)\prod_{q\in\mathcal{Q}}(1-e_q).$$

Here $\mathcal{G}_1$ is the set of 1-gates (gates operating on a single qubit), $\mathcal{G}_2$ is the set of 2-gates (gates operating on two qubits), and $\mathcal{Q}$ is the set of qubits; the term $e_g$ refers to the probability of error of the individual gate $g$, and $e_q$ is the probability of a readout error when we measure the qubit $q$.

An important aspect of (2.1) is that it is assumed that the fidelity $\phi$ is common to circuits with common numbers of qubits and gates (and the same individual gate and qubit error rates) even if the realizations $\mathcal{P}_C$ vary. This allows estimation of the common $\phi$ by averaging over different circuits with the same parameters.

The rationale for Equation (2.1) is simple: as long as there are no errors in the performance of all the gates and all the measurements of the qubits, the circuit produces a sample from the correct distribution. A single error in one of these components leads to an irrelevant sample. The Google paper reports that for a large number of experiments the actual fidelity estimated by Equation (2.1) agrees with the statistical estimator of the fidelity up to 10%–20%.

A simpler form of (2.1) is obtained by replacing the detailed individual values of the fidelities by their average value, leading to

$$(2.2)\quad \phi' \approx (1-0.0016)^{|\mathcal{G}_1|}(1-0.0062)^{|\mathcal{G}_2|}(1-0.038)^n.$$

The Google team reports that $\phi'$ (more precisely, a slight variant, based on a combined estimate for 2-gate and 1-gate errors) differs from $\phi$ by a few percent (in most cases).

We remark that the excellent predictive power of Equation (2.1) is, on its own, a major scientific discovery as well as engineering achievement of the Google experiment. Concerns regarding the claimed predictive power of Equation (2.1) were raised by Kalai [10].[3] In particular, Formula (2.1) is based on the assumption of the independence of the errors, which is often considered unrealistic in the field of system reliability theory (see, e.g., [20] for a general discussion, and [6] for a convenient specific discussion of potential causes for the dependence of errors in quantum systems).

## 2.5 Google's Porter–Thomas Probability Distributions and Fidelity Estimation

In the introduction we described the Porter–Thomas distributions $\mathcal{P}_C(\mathbf{x})$, a random probability distribution on a finite space $\mathcal{X}$, where $C$ is a random circuit. In fact, a random circuit does not have enough randomness to generate a fully random vector on the $(M-1)$-dimensional real simplex (which is the Dirichlet($\mathbf{1}$) distribution). The vectors $\mathcal{P}_C$ are concentrated on a tiny subset of the simplex and are thus pseudo-random. However, $\mathcal{P}_C$ behaves statistically like a realization of a Porter–Thomas distribution, and for our purposes we can consider it as random.

Once the quantum computer produces $N$ samples $\tilde{\mathbf{x}}^{(1)},\dots,\tilde{\mathbf{x}}^{(N)}$, the Google estimator of $\phi$,

$$(2.3)\qquad U = 2^n\frac{1}{N}\sum_{j=1}^{N}\mathcal{P}_C(\tilde{\mathbf{x}}^{(j)}) - 1,$$

proposed in [3], can be computed provided that the probabilities $\mathcal{P}_C(\tilde{\mathbf{x}}^{(j)})$ are known. This estimator is referred to in [3] as "XEB," an abbreviation for "cross-entropy benchmarking." A crucial aspect of this estimator and of Google's statistical approach as a whole is that relatively small samples (of size $N$ of order $10^6$) give powerful confirmation of the fidelity being significantly nonzero for samples from a huge probability space (of size $M = 2^{53} \approx 10^{16}$ in Google's experiment).

We reiterate that computing the Google estimator (2.3) requires the computation of the probabilities $\mathcal{P}_C(\tilde{\mathbf{x}}^{(j)})$,

[3] The same paper, like several earlier papers, explains Kalai's skeptical views regarding the entire endeavor of quantum computers. Those views, however, are not related to the present work.

$j = 1, \dots, N$. Sampling according to the probability distribution $\mathcal{P}_C$ is easy for the quantum circuit $C$, but not computing or estimating individual probabilities. Computing $\mathcal{P}_C(\mathbf{x})$ is assumed to be infeasible for the ultimate experiments involving 53 qubits, and requires heavy computations on (classical) supercomputers for $n$ approaching 40. It is known (see [3] and [1]) that computing $\mathcal{P}_C(\mathbf{x})$, even for a single value $\mathbf{x}$, is NP-hard.

## 3. SIZE BIAS, RANDOM DISTRIBUTIONS, AND A CHOICE MODEL EXAMPLE

### 3.1 Size-Biased Sampling from a Random Distribution

Some background on *size-biased* distributions will be useful in explaining the nature of the sampling schemes described above. For a recent broad survey with applications see [2]. Let $p(x)$ be a probability function on a finite space $\mathcal{X} \subseteq \mathbb{R}_+ := [0, \infty]$ or a continuous density on $\mathcal{X}$, and let $x$ be a nonnegative random variable distributed according to $p(x)$. We say that the random variable $x^*$ has the $p(x)$-size-biased distribution if $P(x^* = x) = xp(x)/a$ in the discrete case, and $x^*$ has density $xp(x)/a$ in the continuous case, where $a = Ex$ is the normalizing constant. If $p(x) = e^{-x}$ for $x > 0$, the density of the Exp(1) distribution, then the $p$-size-biased density is $xe^{-x}$, corresponding to the Gamma(2, 1) distribution.

Consider a distribution $D$ on $\mathbb{R}_+$ with expectation $=1$, and a vector $(z_1, \dots, z_M)$ whose components are i.i.d. distributed by $D$, and so $Ez_i = 1$. We normalize the vector by setting $w_i = z_i / \sum_{j=1}^M z_j$, and denote the normalized vector by $\mathbf{w} = (w_1, \dots, w_M)$. Since $\sum_{j=1}^M z_j/M \to 1$ with probability 1, we have $w_i \approx z_i/M$ for large $M$. The vector $\mathbf{w}$ defines a probability distribution on a finite set of size $M$. When $D$ is the Gamma distribution, the vector $\mathbf{w}$ has the Dirichlet distribution; see, for example, [12]. The special case where the $D = \text{Exp}(1) = \text{Gamma}(1, 1)$ with density $e^{-x}$ for $x > 0$ arose in quantum physics; see Porter and Thomas [19]. This case plays an important role in the Google experiment.

More generally, one can assume that $\mathbf{w}$ is a realization of some distribution on the $(M-1)$-simplex, that is, a random vector of nonnegative components summing to one; this structure is known as a random discrete distribution; see Kingman [12].

Proposition 3.1. *Let $z_1, z_2, \dots$ be nonnegative, i.i.d. distributed by $D$, with expectation $=1$, and for each $M$ define $w_i^{(M)} = z_i / \sum_{j=1}^M z_j$. Let $x_M^*$ denote a value drawn at random from $\{z_1, \dots, z_M\}$, where $P(x_M^* = z_i) = w_i^{(M)}$, $i = 1, \dots, M$. Then with probability $=1$ (over sequences $z_1, z_2, \dots$) we have that $x_M^*$ converges in distribution to the $D$-size-biased distribution as $M \to \infty$.*

Proof. Let $F_n$ be the empirical distribution function of $z_1, \dots, z_M$ (which assigns probability $1/M$ to each $z_i$). If $x^*$ takes one of the values $z_1, \dots, z_M$ with probability $P(x^* = z_i) = z_i / \sum_{j=1}^M z_j$, then clearly the distribution of $x^*$ is the $F_n$-size-biased distribution. Since $F_n$ converges in distribution to $D$, the $F_n$-size-biased distribution converges in distribution to the $D$-size-biased distribution by Theorem 2.3 in [2]. □

In the sampling scheme of (1.1), $N$ values of $w_i$ are sampled from the set $\{w_i\}_1^M$, where with probability $\phi$ the value $w_i$ is drawn with the probability $w_i^{(M)}$ above. The corresponding values $z_i$ are asymptotically distributed according to a mixture of $D$-size-biased values with weight $\phi$, and the original $D$ with weight $1 - \phi$ (since then a value of $z$ is chosen from iid $z_i \sim D$ with equal probabilities). Note that since $\frac{1}{M} \sum_{j=1}^M z_j \approx 1$ by the law of large numbers, $w_i^{(M)} \approx z_i/M$. Obviously, relative to the distribution $D$, large values of $z_i$ or $w_i$ are overrepresented under size-biased sampling, and the distribution of the sample is tilted to the right.

### 3.2 Testing the Size-Biased Distribution and Model (1.1)

The quantum computer produces a sample of bitstrings $\mathcal{S}_{\mathbf{x}} = \{\widetilde{\mathbf{x}}_j\}_1^N$. The $\{w_i\}$'s associated with the bitstrings $\mathbf{x}_i$ must be computed by a classical computer, and with the known association one can focus on the sample $\mathcal{S}_w = \{\widetilde{w}_j\}$, and $\mathcal{S}_{\mathbf{x}}$ is now of no further use.

We can test whether $\{\widetilde{w}_j\}$ arise from a mixture of Dirichlet and size-biased Dirichlet variables according to (1.1) (call the distribution of this mixture $\mathcal{D}$), by, say, looking at a histogram of the sample $\mathcal{S}_w$ and comparing it to the theoretical mixture density of $\mathcal{D}$, which we do in Figure 9 of Section 8. If the model fits, it could be because Google's sampling model (1.1) indeed holds with the given $\{w_i\}$'s. In this case, the quantum computer produces a sample which is correlated with the $w_i$'s, that is, the"signal" can be identified from the noisy sample, indicating supremacy if $n$ is large. If (1.1) is applied with a different (independent) set of probabilities $\{w_i'\}$, the histogram of the sample $\mathcal{S}_w$ will not exhibit size bias. Now, consider a statistic like $U = \frac{1}{N} M \sum_{i=1}^N \widetilde{w}_i - 1$ of (1.3). Under the model (1.1), we expect to observe large $\widetilde{w}_i$'s due to the size bias, and therefore $EU > 0$; thus a positive significant $U$ demonstrates the presence of the the $w_i$'s, that is, supremacy (if $n$ is large).

However, if the sample $\mathcal{S}_w$ exhibits size bias, it does not prove that the model (1.1) holds as stated. For example, if some bitstrings are rejected from the sample with probabilities that are independent of the $w_i$'s, and hence some of the sampled $\widetilde{w}_j$'s are rejected, then the fit to $\mathcal{D}$ will still hold. This will be explained in Section 4.3. Moreover, a fit to $\mathcal{D}$ will be exhibited also if (1.1) is replaced by $\pi(\mathbf{x}^{(i)}) = \pi(w_i) = \phi w_i + (1-\phi)v_i$ with a set of probabilities $\{v_i\}$ that are not uniform, provided that they are independent of the $w_i$'s, that is, the nature of the noise is unimportant as long as the noise is independent of the probabilities $w_i$. Robustness of $U$ under the above rejection model and to different kinds of noise instead of uniform noise are discussed in Section 4.3.

Thus the fit of histograms to $\mathcal{D}$ and a positive significant $U$ can indicate supremacy, without confirming the model (1.1). To test the model (1.1), we need much data; we can then compute a chi-square goodness of fit statistic, for example; see Section 8.

Contrary to Google's $U$, the estimator $T$ of $\phi$, considered in Section 4.7, is nearly unbiased if model (1.1) holds, but is not robust to sampling with rejection, for example. If one knows or claims a certain value of $\phi$, and $T$ suggests a different value, it could be because of certain deviations from (1.1), which may not affect $U$. Computing the estimator $T$ does not require knowledge of $w_i$'s (unlike $U$ and chi-square), but does require large samples relative to $M$.

### 3.3 An Individual Choice Model

We briefly describe a simple application, unrelated to quantum computing, that is a variation on problems in individual choice models in economics; see, for example, [8] for a simple introduction. It provides another perspective on the models we discuss, and suggests other potential applications. Consider $M$ items, say refrigerators (fridges), in a store (they need not be all different). Each fridge is characterized by $p$ attributes such as size, price per given size, energy consumption per size, etc., which for fridge $i$ are $w_{1i}, \dots, w_{pi}$. By considering one attribute relative to another (a further example could be price per size and energy consumption) we attempt to make the attributes $(w_{1i}, \dots, w_{pi})$ independent. Each of $N$ customers chooses one fridge and it is assumed that the probability of choosing fridge $i$ is given by (1.5), that is, a convex combination of the $w$'s that characterize this fridge with the $\phi$'s as coefficients. Thus, the vector of $\phi$'s characterizes the population of customers from which we have a sample of $N$. The parameter $\phi_k$ quantifies the weight the population assigns to attribute $k$ when choosing a fridge, and this is a quantity the seller or the producer would like to know. This paper proposes various ways of estimating the $\phi$'s from the sample. The information consists of the $N$ chosen fridges and their attributes, but the customers do not reveal their selection process, and they may not be aware of it.

A useful interpretation of the linear combination in (1.5) is the following: each customer chooses an attribute at random with probabilities according to the $\phi$'s, and then if attribute $k$ was chosen by customer $i$, she chooses fridge $i$ with a probability proportional to $w_{ki}$, and hence it is a size-biased selection. (See, for example, [4], where size-biased observations with respect to one variable out of several are considered. See also Section 3.) Equivalently, a proportion $\phi_k$ of customers choose in proportion to the value of the $k$th attribute. In either interpretation, $\phi_k$ represents the importance of attribute $k$, thus quantifying the relative importance of attributes like size, price, energy consumption, etc., in the item selection.

## 4. ESTIMATION OF THE FIDELITY $\phi$ FOR $p = 2$

We first discuss the estimation of $\phi$ given a sample $\widetilde{w}_1, \dots, \widetilde{w}_N$ according to Google's model (1.1). In Section 5, we generalize to estimation of $\phi_1, \dots, \phi_p$ in the case of observing $\widetilde{\mathbf{w}}_1, \dots, \widetilde{\mathbf{w}}_N$, sampled according to (1.5). We study two types of analyses and consider biases, variances, etc.: the first type is conditioned on $w_1, \dots, w_M$, which depend on the quantum circuit, and the second is with expectation taken over the random $w_1, \dots, w_M$ when considering random circuits. We compare these analyses and their implications for different estimators, and for different parameters of the problem. In particular, we compare Google's estimator with the maximum likelihood estimator (MLE).

Google's estimator $U$, defined in (1.3) and again in (4.3) below, and the MLE, defined in (4.14), depend only on the sampled $\widetilde{\mathbf{w}}_1, \dots, \widetilde{\mathbf{w}}_N$. Therefore, they are statistics (i.e., functions of observed data) if we assume that the sample $\mathcal{S}_w = \{\widetilde{w}_j\}_1^N$ is observed. The estimator $T$ given in Section 4.7 requires only knowledge of the sample of bitstrings $\mathcal{S}_{\mathbf{x}} = \{\widetilde{\mathbf{x}}^{(j)}\}$ (and not of the associated probabilities). Another estimator considered below in (4.6), $V$, is not a statistic in the sense that computing it requires the knowledge of $w^{(2)} := \sum_{i=1}^M w_i^2$. We mention again that a quantum circuit produces a sample of bitstrings $\mathbf{x}_{\widetilde{w}_1}, \dots, \mathbf{x}_{\widetilde{w}_N}$; however, calculating the associated $\widetilde{w}_1, \dots, \widetilde{w}_N$ is done by classical computers only, and for limited values of $M = 2^n$. For $n < 40$, a classical supercomputer can compute all $w_i$ and hence $V$ can also be computed. For $n > 50$ or so, computing the $w_i$'s is considered practically impossible, and hence the statistics presented in [3] and here (with the exception of $T$) cannot be computed. Therefore, Google's supremacy claim with $n = 53$ is based on skillful extrapolations.

Google's estimator $U$ turns out to be biased for each given circuit. Averaged over sufficiently many different circuits, or in expectation over the random probabilities $w_{\mathbf{x}} = \mathcal{P}_C(\mathbf{x})$, it becomes unbiased, and hence consistent as a function of the number of circuits, as we show below. The bias for each circuit increases the variance between circuits; this variance diminishes along with the bias as $\phi$ becomes small for large $M$ and negligible for $M = 2^{53}$.

### 4.1 Moment Estimators

Google's estimator (see (1.3) or (2.3)), is basically a moment estimator, a notion we briefly review. The *method of moments* (see, e.g., [7], Chapter 18) can be described as follows: let $x_i$ be $N$ iid observations taking values in some space $\mathcal{X}$ and assume that $x_i \sim F_\theta$, where $\theta \in \Theta \subseteq \mathbb{R}^p$. Let $\varphi : \mathcal{X} \to \mathbb{R}^p$ and assume that $E_\theta \varphi(x_i) = g(\theta)$, where $g : \Theta \to \mathbb{R}^p$ is one-to-one. A moment estimator of $\theta$ is of the form

$$T_N = g^{-1}\left(\frac{1}{N}\sum_{i=1}^{N} \varphi(x_i)\right). \tag{4.1}$$

Under standard assumptions, the law of large numbers implies that $\frac{1}{N}\sum_{i=1}^{N} \varphi(x_i) \to E_\theta \varphi(x_i)$ as $N \to \infty$, and further standard smoothness assumptions imply that $T_N$ is *consistent*; that is, $T_N$ converges to $\theta$ in probability; moreover, $\sqrt{N}(T_N - \theta) \to N(\mathbf{0}, \Sigma)$ in distribution, where the asymptotic covariance matrix $\Sigma$ depends on the variance of the variables $x_i$, the functions $\varphi$ and $g$, and their derivatives. In this case, we say that $T_N$ is *root-N consistent* and *asymptotically normal*.

In Section 4.4, we shall discuss the *maximum likelihood estimator* (MLE). The MLE is known to be *asymptotically efficient*, that is, asymptotically normal with the smallest asymptotic variance among asymptotically normal estimators, under certain regularity conditions (see, e.g., Theorem 3.10, Chapter 6 in [14]). Theorems 4.3 and 5.3 of Chapter 6 in [14] show that one Newton–Raphson iteration (toward the MLE) starting from any root-$N$ consistent estimator yields an asymptotically efficient estimator, which means that like the MLE it has the smallest asymptotic variance.

### 4.2 Google's Estimator $U$ of the Fidelity and Its Unbiased Variant $V$

Let $E^w$ denote the conditional expectation of some function of $\{\widetilde{w}_j\}_1^N$ given $\{w_i\}_1^M$, where $\{\widetilde{w}_j\}_1^N$ are sampled according to (1.1). Expectation over the randomness of $\{w_i\}_1^M$ is denoted by $E$. Thus, $Eg(\widetilde{w}_j) = EE^w g(\widetilde{w}_j)$ denotes expectation over both the sampling of $\widetilde{w}_j$ and the randomness of the random probabilities $w_i$. Similarly, $\mathrm{Var}^w$ denotes variance conditioned on $\{w_i\}_1^M$.

In this section, we discuss Google's estimator of $\phi$ for the sampling distribution of (1.1). Consider a vector $(w_1, \dots, w_M) \sim \text{Dirichlet}(\mathbf{1})$, which can be obtained as mentioned before by $w_i = z_i / \sum_{j=1}^M z_j$ with i.i.d. $z_i \sim \text{Exp}(1)$. Let $\widetilde{w}_1, \dots, \widetilde{w}_N$ be a sample (with replacement) of $N$ values of $w_i$ that are sampled independently according to the mixture probabilities

$$\pi(w_i) = \phi w_i + (1 - \phi)/M. \tag{4.2}$$

Google's estimator $U$ of $\phi$ is (in our notation)

$$U = \frac{1}{N}\sum_{j=1}^{N} M\widetilde{w}_j - 1. \tag{4.3}$$

We now compute its expectation and variance conditioned on $(w_1, \dots, w_M)$. Put $w^{(k)} = \sum_{i=1}^M w_i^k$ and so $Ew^{(k)} = k!/[(M+1)\cdots(M+k-1)]$ by (1.2), and in particular $Ew^{(2)} = 2/(M+1)$.

With expectations being conditional on $w_i$ we have, using $\sum_{i=1}^M w_i = 1$,

$$\begin{aligned} E^w M\widetilde{w}_i &= \sum_{i=1}^{M} Mw_i\bigl[\phi w_i + (1-\phi)/M\bigr] \\ &= \phi\bigl(Mw^{(2)} - 1\bigr) + 1. \end{aligned} \tag{4.4}$$

REMARK. If (4.2) is replaced by $\pi(w_i) = \phi w_i + (1 - \phi)v_i$ for some exchangeable random $\{v_i\}$ (with $\sum_{i=1}^M v_i = 1$ and therefore $Ev_i = 1/M$) independent of $\{w_i\}$, then $E^w M \sum_{i=1}^M w_i v_i = M \sum_{i=1}^M w_i Ev_i = 1$ and we obtain the same result as (4.4), showing that $U$ is robust against changing the sampling probabilities under errors. Robustness properties of $U$ and other estimators are discussed in Section 4.3. The $v_i$'s do affect the variance of $U$; however, we shall not pursue these variance calculations.

Returning to (4.4), we obtain

$$E^w U = \phi\bigl(Mw^{(2)} - 1\bigr). \tag{4.5}$$

Thus, we see that for a given set $\{w_i\}_1^M$, that is, a given circuit, Google's estimator $U$ is biased. Clearly, relative to $\phi$, the bias converges to zero as $M \to \infty$; see below. However, the bias does not depend on the sample size $N$, and therefore as $N \to \infty$ the estimator $U$ does not converge to $\phi$; that is, it is inconsistent.

Google's estimator $U$ can be improved and rendered consistent in $N$ by adjusting it to be unbiased. Define

$$V = U/\bigl(Mw^{(2)} - 1\bigr). \tag{4.6}$$

An averaged version of $V$ (that is still biased and hence not consistent) appears in Equation (21) in the Supplement to [3]. We shall see that the unbiased estimator $V$ has a smaller variance than that of $U$, and thus it is an improvement upon $U$. Unlike $U$, computing the estimator $V$ requires access to all $\{w_i\}_1^M$ so that we can compute $w^{(2)}$, and hence it is not a statistic relative to the sample $\widetilde{w}_1, \dots, \widetilde{w}_N$.

Recall that $(w_1, \dots, w_M)$ has the Dirichlet distribution; however, the calculations that follow require only conditions on moments up to order 4. Taking expectation over $\{w_i\}_1^M$ and using $Ew^{(2)} = 2/(M+1)$, we get

$$\begin{aligned} EU &= EE^w U \\ &= E\bigl[\phi\bigl(Mw^{(2)} - 1\bigr)\bigr] = \phi\left(\frac{M-1}{M+1}\right) \approx \phi, \end{aligned} \tag{4.7}$$

and the law of large numbers implies that $U \to \phi(\frac{M-1}{M+1})$ almost surely as $N \to \infty$. Thus, $U$ is nearly unbiased for large $M$ when considered over the randomness of both the sampling process of (1.1) and the random probabilities $\{w_i\}_1^M$, and can be adjusted by dividing it by $(M-1)/(M+1)$ to make it unbiased and consistent as $N \to \infty$. Thus, if $U$ is computed from samples taken from different independent sets of $\{w_i\}_1^M$ (that is, different independent circuits) and the results are averaged, then the resulting average is nearly unbiased for large $M$; however, as we shall see, the bias of $U$ for each realization $\{w_i\}_1^M$ makes its variance larger than that of $V$.

To obtain the variance, we compute $E^w M^2 \widetilde{w}_i^2 = M^2 \sum_{i=1}^M w_i^2[\phi w_i + (1-\phi)/M] = \phi(M^2 w^{(3)} - M w^{(2)}) + M w^{(2)}$. Therefore,

$$\begin{aligned} \operatorname{Var}^w U &= \frac{1}{N} \operatorname{Var}^w(M\widetilde{w}_i) \\ &= \frac{1}{N}\big[\phi(M^2 w^{(3)} - M w^{(2)}) + M w^{(2)} \\ &\quad - \big(\phi(M w^{(2)} - 1) + 1\big)^2\big] \\ &= \frac{1}{N}\big[\phi(M^2 w^{(3)} - 3M w^{(2)} + 2) \\ &\quad - \phi^2(M w^{(2)} - 1)^2 + M w^{(2)} - 1\big]. \end{aligned} \tag{4.8}$$

Taking the denominator $(M w^{(2)} - 1)$ of (4.6) into account, we obtain

$$\begin{aligned} &\operatorname{Var}^w(V) \\ &= \frac{1}{N(M w^{(2)} - 1)^2}\big[\phi(M^2 w^{(3)} - 3M w^{(2)} + 2) \\ &\quad - \phi^2(M w^{(2)} - 1)^2 + M w^{(2)} - 1\big]. \end{aligned} \tag{4.9}$$

Given all $\{w_i\}_1^M$ of a given circuit, or just $w^{(2)}$ and $w^{(3)}$, the quantities of (4.8) and (4.9) can be computed.

Under the moment conditions of (1.2), we have $w^{(2)} \approx 2/M$ and $w^{(3)} \approx 6/M^2$ (where the approximation is in the sense that the ratio of the two sides converges to 1 as $M \to \infty$), and therefore

$$\operatorname{Var}^w(U) \approx \operatorname{Var}^w(V) \approx \frac{1}{N}(2\phi - \phi^2 + 1). \tag{4.10}$$

The difference between the two estimators $U$ and $V$ is that $V$ is unbiased and $U$ is not, and the overall (unconditional) variance of $U$ depends also on $\operatorname{Var}(E^w U)$, which we discuss next. Equation (4.10) provides $\operatorname{Var}^w(U)$, the conditional variance of $U$ of (4.3). Recalling the Pythagorean formula, aka the law of total variance, we have (for any $U$)

$$\operatorname{Var}(U) = E\big[\operatorname{Var}^w(U)\big] + \operatorname{Var}\big[E^w(U)\big], \tag{4.11}$$

and given the fact that $V$ is conditionally unbiased, that is, $E^w(V) = \phi$ and hence $\operatorname{Var}[E^w(V)] = 0$, we conclude by (4.10) that

$$\operatorname{Var}(V) = E \operatorname{Var}^w(V) \approx \frac{1}{N}(2\phi - \phi^2 + 1). \tag{4.12}$$

The situation with $U$ is different and in view of (4.11), we have to compute $\operatorname{Var}(E^w U)$. By (1.2) $\operatorname{Var}(w_i^2) = E w_i^4 - (E w_i^2)^2 \approx 4!/M^4 - (2/M^2)^2 = 20/M^4$, and by (4.5) we have $\operatorname{Var}(E^w U) = \operatorname{Var}[\phi(M w^{(2)} - 1)] \approx 20\phi^2/M$. From (4.11) and (4.10), we conclude that the overall variance of $U$ is

$$\operatorname{Var}(U) \approx \frac{1}{N}(2\phi - \phi^2 + 1) + 20\phi^2/M. \tag{4.13}$$

This variance does not decrease to zero when the sample size $N \to \infty$, and hence $U$ as a function of $N$ is not a consistent sequence of estimators. If $M$ and $N$ are of similar order, the term $20\phi^2/M$ may matter and $U$ will be inferior to $V$ (and to the MLE as shown later). This may happen if one extrapolates from relatively small circuits, that is, small $M$. When $n = 53$, the number of qubits in Google's ultimate quantum computer, and when $N$ is of order $10^6$, a typical sample size in Google's experiment, that term is unlikely to matter.

The variances of $V$ and $U$ given in (4.12) and (4.13) are approximations to the respective variances when $\{w_i\}_1^M$ are considered random with the first four moments corresponding to the Dirichlet distribution. Suppose that we have $L$ estimators $V_\ell$ arising from different circuits having a common fidelity $\phi$. Then for $\overline{V} = \frac{1}{L}\sum_{\ell=1}^L V_\ell$ we have $E\overline{V} = \phi$, and by (4.12) $\operatorname{Var}(\overline{V}) \approx \frac{1}{LN}(2\phi - \phi^2 + 1)$. A similar result holds for $U$, where by (4.7), we obtain for $\overline{U} = \frac{1}{L}\sum_{\ell=1}^L U_\ell$ that $E\overline{U} \approx \phi$, and by (4.13) that $\operatorname{Var}(\overline{U}) \approx \frac{1}{LN}(2\phi - \phi^2 + 1) + \frac{20}{LM}\phi^2$. These results will be used in Section 7 for constructing approximate confidence intervals.

### 4.3 A Robustness Property of *U*

Consider sampling according to (4.2), but suppose that certain bitstrings $\mathbf{x}_i$ and hence the associated $w_i$ are rejected from the sample with probability $\rho_i \in [0, 1]$, and hence are accepted with probability $\tau_i := 1 - \rho_i$, where the $w_i$'s are i.i.d. as usual and are assigned to the $\tau_k$'s at random. Sampling continues until $N$ bitstrings are obtained. We show that Google's estimator $U$ as well as $V$ and the MLE remain valid for estimating $\phi$ and there is no need for adjustments, nor to know the acceptance probabilities $\tau_i$. Conditioning on the $w_i$'s and attaching the $\tau_k$'s to them at random we have instead of (4.2) the sampling probabilities

$$\begin{aligned} \pi_\tau(w_i) &= \sum_{k=1}^M \tau_k\big[\phi w_i + (1-\phi)/M\big] \\ &\quad \Big/ \left\{\sum_{i=1}^M \sum_{k=1}^M \tau_k\big[\phi w_i + (1-\phi)/M\big]\right\} \\ &= \phi w_i + (1-\phi)/M. \end{aligned}$$

Thus the sampling distribution of a bitstring is the same as that of (4.2) and hence the estimators of Section 4 continue to be valid. However, once sampling is repeated with the same (random) association of the $\tau_k$'s to the $w_i$'s, the sampled $\widetilde{w}_i$ are not independent. If only two of the $\tau_k$'s are positive, say, then observing $\widetilde{w}_i$ suggests that it is associated with the bigger $\tau_k$ and is more likely to appear again. However, the dependence vanishes as $M \to \infty$ if, for example, half of the $\tau_i$'s are equal to 1 and half to zero. In this case, for large $M$, we simply have to replace $M$ by $M/2$ in the formulas of Section 4.2. A similar result holds if for some $c, d \in (0, 1]$ we have that $cM$ of the $\tau_i$'s satisfy $\tau_i > d$. For a discussion of the robustness of $U$ against changing the sampling probabilities under error from a uniform to a general distribution (independent of $\{w_i\}$) see the remark following (4.4). Like the acceptance-rejection model above, the nonuniform probabilities entail dependence, which affects the variance of $U$.

### 4.4 Maximum Likelihood Estimation of $\phi$

The likelihood function of the sample $\widetilde{w}_1, \dots, \widetilde{w}_N$ is given by the product $\prod_{j=1}^N (\phi\widetilde{w}_j + (1-\phi)/M)$, and the MLE of $\phi$ is obtained by maximizing it with respect to $\phi$. This can be done by equating the derivative of the log-likelihood to zero; that is, the MLE is the solution in $\phi$ to the equation

$$f(\phi) := \frac{\partial}{\partial\phi} \sum_{j=1}^N \log(\phi\widetilde{w}_j + (1-\phi)/M) \tag{4.14}$$
$$= \sum_{j=1}^N \frac{\widetilde{w}_j - 1/M}{\phi\widetilde{w}_j + (1-\phi)/M} = 0.$$

An algorithm for the solution will be given below. The MLE for several different circuits together is considered in Equation (20) in the Supplement to [3].

We now discuss the variance of the MLE. Under the present conditions, it is well known that the asymptotic (in $N$) variance of the MLE is given by $1/\mathscr{I}(\phi)$, where $\mathscr{I}(\phi)$ is the Fisher information, which is the expectation, conditioned on $\{w_i\}_1^M$, of the second derivative of the log-likelihood function with a negative sign. See, for example, [14] Chapter 6, Theorem 3.10. We have

$$\mathscr{I}(\phi) = -E^w\Big[\frac{\partial}{\partial\phi}\sum_{j=1}^N \frac{\widetilde{w}_j - 1/M}{\phi\widetilde{w}_j + (1-\phi)/M}\Big]$$
$$= NE^w \frac{(\widetilde{w}_j - 1/M)^2}{[\phi\widetilde{w}_j + (1-\phi)/M]^2}$$
$$= N\sum_{i=1}^M \Big\{\frac{(w_i - 1/M)^2}{[\phi w_i + (1-\phi)/M]^2} \tag{4.15}$$
$$\times [\phi w_i + (1-\phi)/M]\Big\}$$
$$= N\sum_{i=1}^M \frac{(w_i - 1/M)^2}{\phi w_i + (1-\phi)/M}.$$

Given all $\{w_i\}_1^M$ of a specific circuit and the fidelity $\phi$, the Fisher information can be computed. When $\phi$ is unknown, plug-in estimation is used.

The quantity given in (4.15) can be approximated by $N\int_0^\infty \frac{(z-1)^2}{\phi z+1-\phi} e^{-z}\,dz$ if $M$ is large, and we approximate $w_i$ by $z_i/M$ and assume that $z_i \sim \text{Exp}(1)$. Recalling that the asymptotic variance of the MLE is $1/\mathscr{I}(\phi)$, we have

$$\text{Var}^w(\text{MLE}) \approx \frac{1}{N\int_0^\infty \frac{(z-1)^2}{\phi z+1-\phi} e^{-z}\,dz}. \tag{4.16}$$

In view of (4.11), we need to consider $\text{Var}[E^w(\text{MLE})]$. In our case the variables, the log-likelihood, and its derivatives are all bounded and therefore $E^w[(\text{MLE}) - \phi] = O(1/N)$ (see, e.g., [5], Section 9.2) and therefore its variance is of order $O(1/N^2)$, which we neglect, and our approximation for Var(MLE), the unconditional variance, is the same as in (4.16).

If we have $L$ independent MLE estimators, denoted by $\text{MLE}_\ell$, from $L$ independent samples $\{\widetilde{w}_j\}$, arising either from the same or from several circuits assuming that they have a common $\phi$, we can define $\overline{\text{MLE}} = \frac{1}{L}\sum_{\ell=1}^L \text{MLE}_\ell$. We then have

$$\text{Var}(\overline{\text{MLE}}) \approx \frac{1}{LN\int_0^\infty \frac{(z-1)^2}{\phi z+1-\phi} e^{-z}\,dz}. \tag{4.17}$$

It is easy to see that computing the MLE from joint likelihood of all $L$ independent samples requires summing (4.14) over the $L$ samples, and the Fisher information will be the sum of $L$ terms that are similar to (4.15), and the resulting approximation of the variance of the joint MLE coincides with (4.17).

We next provide explicit formulas for the Newton–Raphson algorithm for computing the MLE. This requires computations of

$$J(\phi) = \frac{\partial^2}{\partial\phi^2}\sum_{j=1}^N \log(\phi\widetilde{w}_j + (1-\phi)/M) \tag{4.18}$$
$$= -\sum_{j=1}^N \frac{(\widetilde{w}_j - 1/M)^2}{[\phi\widetilde{w}_j + (1-\phi)/M]^2}.$$

With $f(\phi)$ of (4.14) the Newton–Raphson iterations are given by

$$\phi_{k+1} = \phi_k - f(\phi_k)/J(\phi_k). \tag{4.19}$$

To see the relation between the MLE and the estimator $V$, we write a first-order Taylor expansion of the function $f(\phi)$ of (4.14) at $\phi = 0$ to obtain

$$f(\phi) = \sum_{j=1}^N \frac{\widetilde{w}_j - 1/M}{\phi\widetilde{w}_j + (1-\phi)/M}$$
$$\approx M\sum_{j=1}^N (\widetilde{w}_j - 1/M) - \phi M^2 \sum_{j=1}^N (\widetilde{w}_i - 1/M)^2.$$

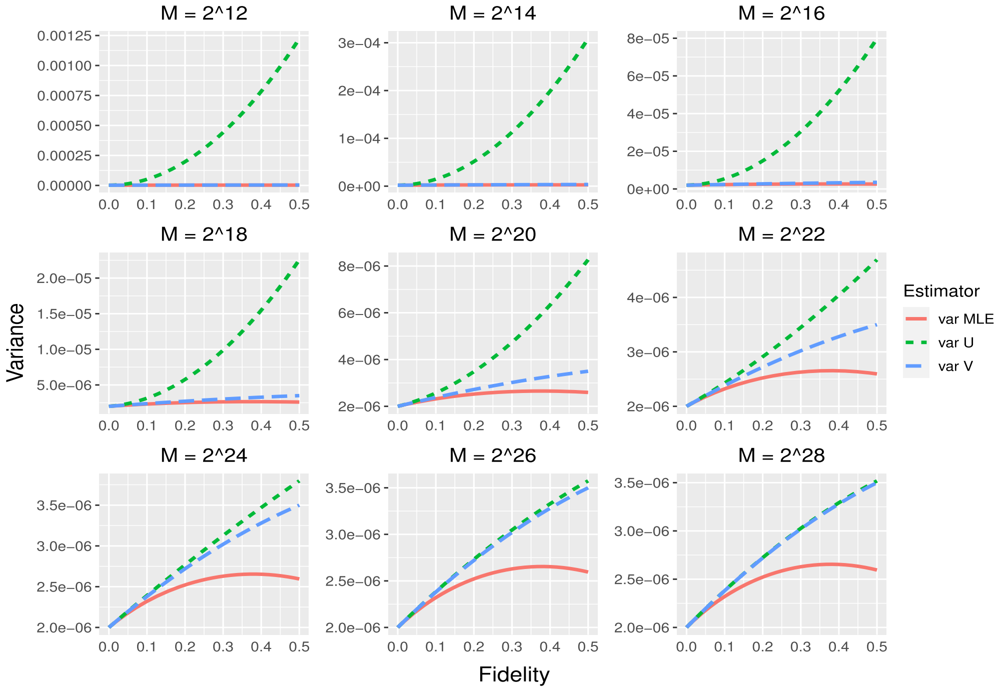

FIG. 1. *Variances of MLE, $U$, and $V$ as a function of $\phi$.*

By (4.2) we have

$$\begin{aligned}&M^2\frac{1}{N}\sum_{j=1}^{N}(\widetilde{w}_j-1/M)^2\\&\quad\approx M^2E^w(\widetilde{w}_j-1/M)^2\\&\quad=M^2\sum_{i=1}^{M}(w_i-1/M)^2[\phi w_i+(1-\phi)/M],\end{aligned}$$

and for $\phi=0$ the latter expression equals $Mw^{(2)}-1$ since $\sum_{i=1}^{M}w_i=1$. Thus, setting $f(\phi)=0$ in the above Taylor expansion and solving for $\phi$, we obtain that the solution is $(\frac{1}{N}\sum_{i=1}^{N}M\widetilde{w}_i-1)/(Mw^{(2)}-1)$, which is the estimator $V$, showing that $V$ is an approximation to the MLE for small values of $\phi$.

### 4.5 Estimator Comparisons

We compare the unconditional variances of the estimators $U$, $V$, and MLE for $N=500{,}000$, $M=2^{12}-2^{28}$, and $0\le\phi\le 1/2$ according to the formulas

$$\begin{aligned}\mathrm{Var}(U)&\approx\frac{1}{N}(2\phi-\phi^2+1)+\frac{20\phi^2}{M},\\\mathrm{Var}(V)&\approx\frac{1}{N}(2\phi-\phi^2+1),\\\mathrm{Var}(\mathrm{MLE})&\approx\frac{1}{N\int_0^\infty\frac{(z-1)^2}{\phi z+1-\phi}e^{-z}\,dz}.\end{aligned}$$

A second-order Taylor expansion of the latter approximation of Var(MLE) around $\phi=0$ yields $\frac{1}{N}(2\phi-5\phi^2+1)$, which is smaller than the approximate expression of Var($V$); however, we see again that for small $\phi$ these variances are close together. The variances are given as a function of $\phi$ in Figure 1. Note that the ranges of the y-axes change between plots in Figure 1 and some of the other figures. Consider the following numerical example: for $M=2^{18}$ and $\phi=0.15$ we have that Var($V$) based on about $N=299{,}000$ equals Var($U$) based on $N=500{,}000$, and so $V$ entails a reduction by 40% in the required sample size to achieve the same variance or confidence interval width.

In Figure 2, we generated files $\{w_i\}_1^M$ with $M=2^{12}, 2^{14}$, and $2^{16}$, from the Dirichlet distribution, and sampled from them using (1.1) with $\phi=0.3862, 0.3320$, and $0.2828$, respectively (indicated by orange dashed lines), and $N=500{,}000$. The above values of $\phi$ correspond to values computed in [3] using a slightly modified version of (2.1). We averaged the estimator over 10 files for each $M$, as done in [3]. This was repeated 100 times. We know that the MLE and $V$ are unbiased even without averaging, whereas $U$ is biased. The boxplots below show that the average value of $U$ over 10 files has a small bias, but its variance is larger than that of the other two estimators.

We next present comparisons of MLE, $U$, and $V$ conditional on $\{w_i\}_1^M$, that is, for given circuits. This is demonstrated on Google's data sets. In Figure 3, we took 10 files $\{w_i\}_1^M$ of size $M=2^n=2^{14}$, which are obtained from Google's data. These files of probabilities $\{w_i\}_1^M$ will be used several times below. As before, we sampled $N=500{,}000$ values from each set of the 10 probability files $\{w_i\}_1^M$ according to (1.1) with $\phi=0.3320$ (indicated

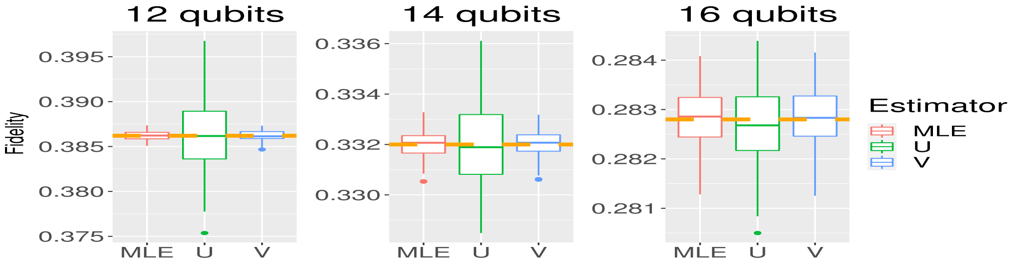

FIG. 2. *Boxplots of* 100 *averages of MLE, $U$, and $V$.*

by the dashed orange line). This was repeated 100 times for each of the 10 files. The dotted green line represents Google's estimate $U$ of the fidelity, which we computed for each file. The boxplots show the MLE, $U$, and $V$ for 100 values each.

We see that if the true fidelity is 0.3320 as suggested by Google, then their estimate $U$ is off target in each case (but only by 5% or less). There are several possible explanations: first, we know that $U$ is biased. For some of the files Google's estimator is within the range of our simulated $U$'s, but for others it is not, suggesting that further explanations are needed. Other potential explanations are that the theoretical number $\phi = 0.3320$ is not exactly the actual $\phi$ that generated the sample, or that there are some other deviations from (1.1) that affect the estimator $U$ (despite its robustness). When we compute Google's $U$ we see that indeed for each of the 10 files it is biased, and as expected the bias can be upward or downward, depending on the particular $\{w_i\}_1^M$. On the other hand, $V$ and the MLE are always centered around the value of $\phi$ that was used in the simulation, which is no surprise as $V$ is unbiased for each given set $\{w_i\}_1^M$, and the MLE is consistent and hence nearly on target when $N$ is large.

Figure 4 is similar to Figure 3, except that we sampled with the value of the average MLE over the 10 files (indicated by the dashed turquoise line), rather than a theoretical $\phi$. In Figure 5, we calculated the MLE of each of the 10 files separately, and used it as $\phi$ (indicated by the dashed turquoise line) for the simulation based on (1.1). This is the most accurate way to simulate Google-like experiments based on the full list of probabilities $\{w_i\}_1^M$. Figure 5 suggests that the fidelity varies only slightly between circuits, and that the MLE represents the actual value of $\phi$ for each circuit.

### 4.6 Further Analysis of Google's Estimator

Although the MLE is in general better than $U$, it is of interest to consider explicit estimators like $U$. We next show that Google's choice of $U$ is best in the sense of minimizing the variance as $\phi \to 0$ in a class of linear estimators defined below, while for larger values of $\phi$ another estimator, also discussed in [3], is better. Some of these results, which motivated their choice of $U$, are reported in the Supplement to [3]. We have the following proposition.

PROPOSITION 4.1. *Assume that* $(w_1, \dots, w_M) \sim \mathrm{Dirichlet}(\mathbf{1})$. *Among all linear unbiased estimators of the form*

$$U_g := \frac{1}{N} \frac{\sum_{j=1}^{N} g(\tilde{w}_j) - B}{A},$$

*that is, with $A$ and $B$ such that $EU_g = \phi$, the choice $g(w) = w$ for which $U_g = U$ of* (4.3) *minimizes* $\mathrm{Var}(U_g)$ *when $M$ is large and $\phi = 0$, and in this sense $U$ is optimal*

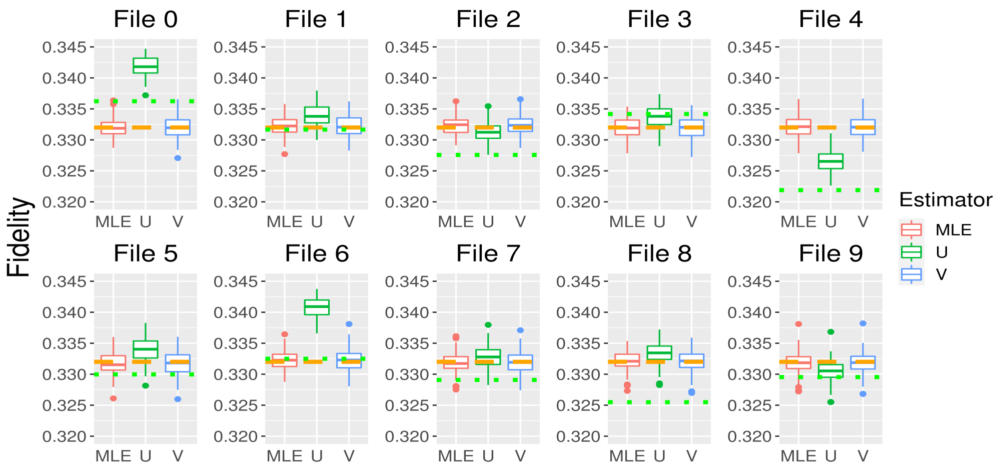

FIG. 3. *Boxplots of* 100 *values of MLE, $U$ and $V$, with $\phi = 0.3320$ and $n = 14$.*

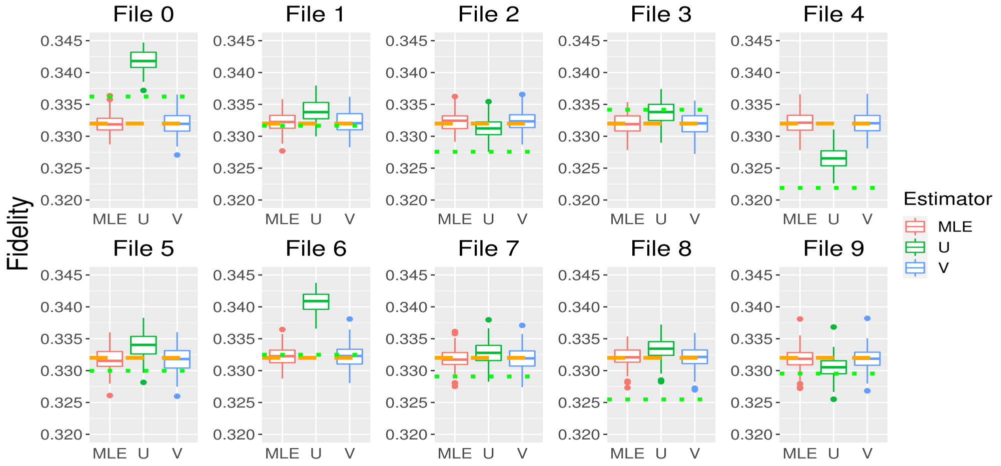

FIG. 4. *Boxplots of* 100 *values of MLE*, $U$, *and* $V$, *with* $\phi =$ *average MLE.*

*as* $\phi \to 0$. *For large values of* $\phi$ *(approximately* $\phi > 0.32$) *the estimator* $U_g$ *with* $g(z) = \log(z)$ *has a smaller variance than that of* $U$.

PROOF. Let $Y := \frac{1}{N}\sum_j g(\widetilde{w}_j)$. We have

$$\begin{aligned} E^w Y &\equiv E(Y \mid \{w_i\}) \\ &= E[g(\widetilde{w}_j) \mid \{w_i\}] \\ &= \sum_{i=1}^{M} g(w_i)(\phi w_i + (1-\phi)/M). \end{aligned} \tag{4.20}$$

Unconditioning, that is, taking expectation in (4.20) over $w_i$, we have

$$\begin{aligned} EY &= Eg(\widetilde{w}_j) \\ &= ME[\phi g(w)(w - 1/M)] + Eg(w) \\ &= \phi A + B, \end{aligned} \tag{4.21}$$

where $A = ME[g(w)(w - 1/M)]$, $B = Eg(w)$, and $w$ stands for a generic $w_i$.

We analyze the best approximate choice of $g$ for small $\phi$ by setting $\phi = 0$. Since $U_g$ is unbiased, Equation (4.11) implies that $\mathrm{Var}(U_g) = E[\mathrm{Var}(U_g)|\{w_i\}]$; we therefore consider $\mathrm{Var}(Y) = E[\mathrm{Var}(Y|\{w_i\})]$. When $\phi = 0$ we have uniform sampling and since conditionally on $\{w_i\}$ the $\widetilde{w}_i$'s are i.i.d., we have that

$$\begin{aligned} \mathrm{Var}\Bigg[\sum_{j=1}^{N} g(\widetilde{w}_j)|\{w_i\}\Bigg] &= N\,\mathrm{Var}[g(\widetilde{w}_j)|\{w_i\}] \\ &= \frac{N}{M}\sum_{i=1}^{M}[g(w_i) - \mu_g]^2 \end{aligned}$$

where $\mu_g = \sum_{i=1}^{M} g(w_i)/M$. It is easy to see that $\mu_g \to Eg(w_i)$ with probability 1 as $M \to \infty$ and therefore,

$$\begin{aligned} E\,\mathrm{Var}\Bigg[\sum_{j=1}^{N} g(\widetilde{w}_j)|\{w_i\}\Bigg] &= E\frac{N}{M}\sum_{i=1}^{M}[g(w_i) - \mu_g]^2 \\ &= EN[g(w_i) - \mu_g]^2 \\ &\to N\,\mathrm{Var}[g(w_i)]. \end{aligned}$$

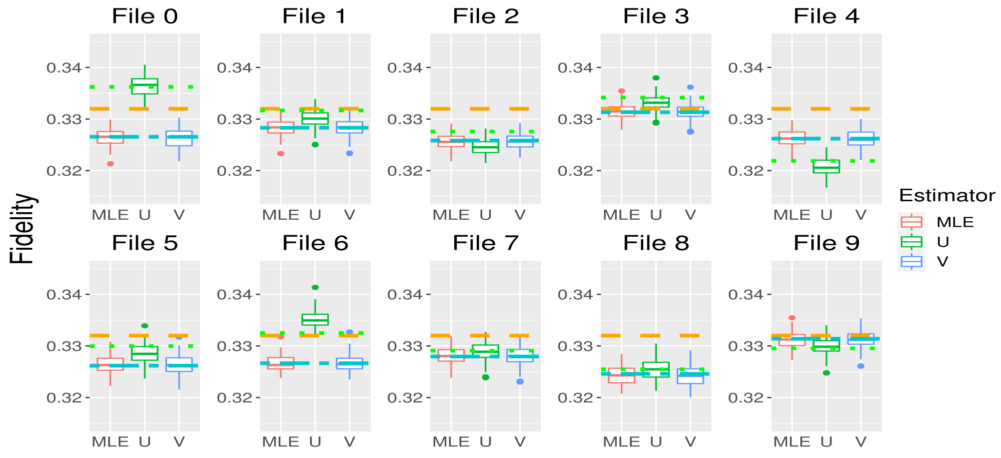

FIG. 5. *Boxplots of* 100 *values of MLE*, $U$, *and* $V$, *with* $\phi =$ *MLE of each file.*

With $w$ replacing $w_i$ we have $\mathrm{Var}(w) = Ew^2 - (Ew)^2 = 2/M(M+1) - 1/M^2 = (M-1)/M^2(M+1)$. Recalling $A$ from (4.21), we have for small $\phi$ and large $M$,

$$\begin{aligned} N\,\mathrm{Var}(U_g) &\approx \frac{N\,\mathrm{Var}[g(w)]}{M^2\{E[g(w)(w-1/M)]\}^2} \\ &= \frac{N}{M^2\,\mathrm{Var}(w)} \frac{\mathrm{Var}[g(w)]\,\mathrm{Var}(w)}{\{E[g(w)(w-1/M)]\}^2} \\ &= \frac{(M+1)}{(M-1)Corr^2(g(w),w)}. \end{aligned}$$

Thus, $\mathrm{Var}(U_g)$ is minimized when the correlation takes its maximal value 1, which is equivalent to $g$ being linear.

We now analyze the choice $g(w) = \log(w)$, which is mentioned in the Supplement to [3] and seems natural from the point of view of information theory and maximum likelihood theory. Using the approximation $w_i = z_i/M$, and assuming $z_i \sim \mathrm{Exp}(1)$ and some calculations given in the Appendix, we have

$$\begin{aligned} \mathrm{Var}(U_{g(w)=\log(w)}) &\approx \frac{1}{N}(\pi^2/6 - \phi^2) \\ &\approx \frac{1}{N}(1.6449 - \phi^2). \end{aligned} \tag{4.22}$$

Equation (4.13) with large $M$ implies

$$\mathrm{Var}(U) \approx \frac{1}{N}(2\phi - \phi^2 + 1). \tag{4.23}$$

The expression in (4.23) is increasing in $\phi$ and that in (4.22) is decreasing. They intersect when $\phi = 0.6449/2 = 0.32$. It follows that for small values of $\phi$ Google's linear estimator of (4.3), which coincides with $U_{g(z)=z}$, is better than $U_{f(z)=\log(z)}$; however, for approximately $\phi > 0.32$ we see that $U_{f(z)=\log(z)}$ is better in the sense of having a smaller variance. □

The results given in (4.23) and (4.22) and the comparison between them appear in the Supplement to [3] (without proof and details).

### 4.7 Further Estimation of Fidelity

As mentioned before, Google's estimator $U$ as well as the MLE require knowledge of the probabilities $\{\tilde{w}_j\}_1^N$ associated with the observed sample $\{\mathbf{x}_{\tilde{\mathbf{w}}_j}\}_1^N$, whereas computation of $V$ requires knowledge of the whole set of probabilities $\{w_i\}_1^M$, both considered hard to compute for large $M = 2^n$.

We now propose an estimator of the fidelity $\phi$ that is a function only of the observed sample of bitstrings $\mathcal{S}_{\mathbf{x}} = \{\tilde{\mathbf{x}}^{(j)}\}_1^N$ and, unlike the above-mentioned estimators, does not require any knowledge of the sampling probabilities. However, this estimator requires a large sample size $N$ relative to the circuit size $n$; therefore, it is proposed only for its conceptual value and not as a practical way to avoid the difficulty of computing the probabilities $\{w_i\}_1^M$.

The estimator $T$ proposed next is closely related to the notion of speckle purity benchmarking (SPB) from the supplement to [3], Section VI.C.4, and especially equation (49). This estimator was used in [3] for circuits of 2 qubits and was not computed for the experimental samples.

Given a sample $\mathcal{S}_{\mathbf{x}}$, define for any bitstring $\mathbf{x}^{(i)} \in \{0,1\}^n$

$$n_i := \text{number of bitstrings in } \mathcal{S}_{\mathbf{x}} \text{ that are equal to } \mathbf{x}^{(i)}.$$

With $\pi_i := \pi(\mathbf{x}^{(i)})$ and sampling according to (1.1) we have $n_i \sim \mathrm{Bin}(N, \pi_i)$. In order to construct an estimator of $\phi$, we first compute

$$\begin{aligned} E^w\left[\sum_{i=1}^M n_i^2\right] &= \sum_{i=1}^M E^w[n_i^2] \\ &= \left(\sum_{i=1}^M \mathrm{Var}^w(n_i) + [E^w n_i]^2\right) \\ &= \sum_{i=1}^M (N\pi_i(1-\pi_i) + N^2\pi_i^2) \\ &= N + (N^2 - N)\sum_{i=1}^M \pi_i^2. \end{aligned} \tag{4.24}$$

By (1.1), with $w^{(2)} := \sum_{i=1}^M w_i^2$ as before, we have

$$\begin{aligned} \sum_{i=1}^M \pi_i^2 &= \sum_{i=1}^M (\pi(\mathbf{x}^{(i)}))^2 \\ &= \sum_{i=1}^M (\phi w_i + (1-\phi)/M)^2 \\ &= \phi^2(w^{(2)} - 1/M) + 1/M, \end{aligned}$$

and then by (4.24) $E^w[\sum_{i=1}^M n_i^2] = N + (N^2 - N) \times (\phi^2(w^{(2)} - 1/M) + 1/M)$. Recalling $Ew^{(2)} = 2/(M+1)$ we readily obtain by taking another expectation

$$\begin{aligned} E\left[\sum_{i=1}^M n_i^2\right] &= N + (N^2 - N) \\ &\quad \times [\phi^2(M-1)/[M(M+1)] + 1/M]. \end{aligned}$$

This implies that the statistic

$$\begin{aligned} T^{[2]} &:= \frac{M(M+1)}{(N^2-N)(M-1)} \\ &\quad \times \left(\sum_{i=1}^M n_i^2 - N - (N^2 - N)/M\right) \end{aligned}$$

is an unbiased and consistent estimator of $\phi^2$. It may take negative values, and we define $T := \sqrt{\max\{T^{[2]}, 0\}}$, which is again a consistent estimator of $\phi$. Counting negative values as zero causes an upward bias. On the other

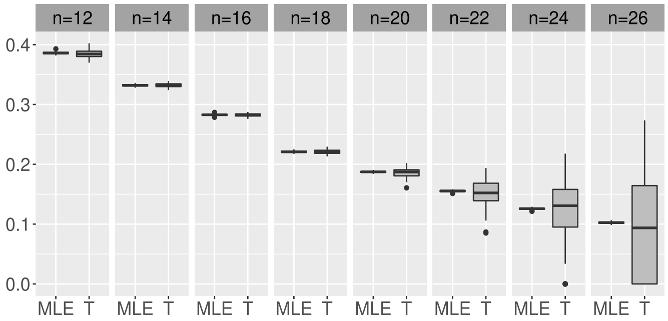

FIG. 6. *Boxplots of T and the MLE.*

hand, taking the root to estimate $\phi$ causes a downward bias that increases with the variance.

Each boxplot in Figure 6 represents 100 simulated values of the statistic $T$ and the MLE according to Google's model (1.1), which is an excellent estimator of the fidelity used for each simulation. We simulated for circuits with even sizes $n = 12 - 26$ qubits, with the corresponding fidelities according to (2.1), with sample size $N = 500{,}000$.

We see that for $n = 26$ the estimator $T$ has a very large spread, and hence it becomes useless for $N$ as above, which is of the order of Google's sample sizes.

Using Google's files, a comparison between average $T$ (of 10 files), average MLE, and the fidelity according to (2.1) (Equation (77) in the Supplement to [3]), given in Table 1 below, shows that $T$ is biased upward. It is easy to see that the statistic $T$ does not possess the robustness of $U$ described in Section 4.3, and selection against part of the bitstrings will tilt $T$ upwards. Whatever the explanation of the overestimation of $\phi$ by $T$ in Google's file, it suggests that they deviate from the sampling model (1.1), and in particular have a larger empirical variance than expected by the model. We return to this issue in Section 8.

Note that if the estimators $U$, $V$, and the MLE yield a significant value of $\phi$ for a sample produced by a given circuit with given probabilities, it is evidence that the circuit performed a sampling task related to the given probabilities. This is not so for the estimator $T$, which cannot provide such evidence; its distribution does not depend on the particular circuit's probabilities $\{w_i\}$, and would be the same regardless of whether the circuit performed the sampling task using the claimed set of $\{w_i\}$, or another set generated at random according to the relevant Porter–Thomas distribution.

TABLE 1
*A comparison of fidelity estimates*

| n | (77) | AVG MLE | AVG T |
|---|---|---|---|
| 12 | 0.3862 | 0.3687 | 0.4689 |
| 14 | 0.3320 | 0.3275 | 0.4392 |
| 16 | 0.2828 | 0.2725 | 0.3917 |
| 18 | 0.2207 | 0.2444 | 0.3557 |
| 20 | 0.1875 | 0.2184 | 0.3210 |
| 22 | 0.1554 | 0.1651 | 0.2989 |
| 24 | 0.1256 | 0.1407 | 0.2838 |
| 26 | 0.1024 | 0.1140 | 0.2600 |

## 5. THE GENERAL CASE OF $p$

### 5.1 A Google-Type Estimator

We now briefly consider sampling according to (1.5) for a general $p$. As mentioned before, this will be relevant when we consider more general sampling models that account for different kinds of sources of noise. The Google-type estimator of $\phi_k$ can be defined as

$$U_k = \frac{1}{N}\sum_{i=1}^{N}(M\widetilde{w}_{ki} - 1). \tag{5.1}$$

Using the fact that $\phi_1 + \cdots + \phi_p = 1$ and $\sum_{i=1}^{M} w_{ki} = 1$, we have

$$\begin{aligned} E^w U_1 &= \sum_{i=1}^{M}(Mw_{1i} - 1)(\phi_1 w_{1i} + \cdots + \phi_p w_{pi}) \\ &= M\phi_1 w_1^{(2)} + M\sum_{k=2}^{p}\sum_{i=1}^{M}\phi_k w_{1i} w_{ki} - 1, \end{aligned}$$

where $w_1^{(2)} = \sum_{i=1}^{M} w_{1i}^2$. For small $M$ the estimator is significantly biased and like $U$ of (4.3), the bias can be corrected if all $\{w_{ki}\}$ are known (in fact the sum of squares and inner products suffice). Taking another expectation, we have

$$\begin{aligned} EU_1 &= EE^w U_1 \\ &= E\Bigg[M\phi_1 w_1^{(2)} + M\sum_{k=2}^{p}\sum_{i=1}^{M}\phi_k w_{1i} w_{ki}\Bigg] - 1 \\ &= 2\phi_1 M/(M+1) - \phi_1 = \phi_1\left(\frac{M-1}{M+1}\right), \end{aligned}$$

where we assume that $\{w_{ki}\}$ and $\{w_{\ell i}\}$ are independent (or just uncorrelated) for all $k \neq \ell$ and satisfy the moment conditions in (1.2). In fact, it suffices to assume that $Ew_1^{(2)} = 2/(M+1)$ and $Ew_{ki} = 1/M$. Thus, $U_k$ is asymptotically unbiased in $M$ when considered over random circuits.

### 5.2 Maximum Likelihood Estimation

The log likelihood of the sample is

$$\begin{aligned} \ell(\boldsymbol{\phi}) &= \sum_{i=1}^{N}\log(\phi_1\widetilde{w}_{1i} + \cdots + \phi_p\widetilde{w}_{pi}) \\ &= \sum_{i=1}^{N}\log\big[\phi_1(\widetilde{w}_{1i} - \widetilde{w}_{pi}) + \cdots \\ &\quad + \phi_{p-1}(\widetilde{w}_{(p-1)i} - \widetilde{w}_{pi}) + \widetilde{w}_{pi}\big]. \end{aligned}$$

The maximum likelihood estimator of $\boldsymbol{\phi} = (\phi_1, \dots, \phi_p)$ such that $\phi_1 + \cdots + \phi_p = 1$ is obtained by computing the gradient $\mathbf{f}(\boldsymbol{\phi})$ (also known as the score) and solving the equation

$$\mathbf{f}(\boldsymbol{\phi}) = \left(\frac{\partial}{\partial \phi_1}\ell(\boldsymbol{\phi}), \dots, \frac{\partial}{\partial \phi_{p-1}}\ell(\boldsymbol{\phi})\right) = \mathbf{0},$$

where

$$\begin{aligned} &\frac{\partial}{\partial \phi_k}\ell(\boldsymbol{\phi}) \\ &= \sum_{i=1}^{N} \Big\{ (\widetilde{w}_{ki} - \widetilde{w}_{pi}) \\ &\quad /\big[\phi_1(\widetilde{w}_{1i} - \widetilde{w}_{pi}) + \cdots \\ &\quad + \phi_{p-1}(\widetilde{w}_{(p-1)i} - \widetilde{w}_{pi}) + \widetilde{w}_{pi}\big]\Big\}. \end{aligned} \tag{5.2}$$

Starting with root-$N$ consistent estimators such as the moment estimators described above, we compute the MLE's by the Newton–Raphson algorithm. As mentioned before, asymptotically, one iteration suffices. Specifically, form the $(p-1) \times (p-1)$ Hessian matrix $J(\boldsymbol{\phi})$ by taking another derivative in (5.2) (see (4.18)) and then iterate by $\boldsymbol{\phi}_1 = \boldsymbol{\phi}_0 - \mathbf{f}(\boldsymbol{\phi}_0)J^{-1}(\boldsymbol{\phi}_0)$ (see (4.19)).

## 6. REFINED NOISE MODELS AND READOUT ERRORS

Recall that in Google's notation $\mathcal{P}(\mathbf{x}^{(i)}) = w_i$ is the (random) probability of observing the bitstring $\mathbf{x}^{(i)}$ ($i = 1, \dots, M$) in the noiseless situation, namely, when $\phi = 1$. The values $w_i$ depend on the quantum circuit $C$ and they are modeled to behave as independent Exp(1) random variables normalized by their sum (that is, as a Dirichlet distribution). We write $\mathcal{P}_C(\mathbf{x}^{(i)})$ instead of $\mathcal{P}(\mathbf{x}^{(i)})$, to indicate the dependence on the quantum circuit $C$. In Google's basic noise model (1.1) the sampling probabilities are $\phi\mathcal{P}_C(\mathbf{x}^{(i)}) + (1-\phi)/M$.

We now consider refined descriptions of the noise, replacing (1.1) by sampling probabilities such as $\phi_1\mathcal{P}_C(\mathbf{x}^{(i)}) + \phi_2\mathcal{N}_C(\mathbf{x}^{(i)}) + \phi_3/M$, where $\phi_1 + \phi_2 + \phi_3 = 1$, and $\mathcal{N}_C(x)$ represents noise that depends on the circuit $C$.

In the next section, we describe a noise model of this kind, where $\mathcal{N}_C(x)$ describes the effect of readout errors when there are no gate errors. $\mathcal{N}_C$ can be seen as a secondary weak signal within the Google data that is largely independent of the primary signal $\mathcal{P}_C$. This refines the analysis of errors, and allows us to offer independent estimation of the fidelity by using the tools of Sections 4.2 and 5. Our analysis of the cases $n = 12$ and 14 indeed identifies the presence of this secondary signal and largely agrees with the fidelity estimation based on the primary signal $\mathcal{P}_C$. In Section 8, we show that Google's samples are far from fitting the sampling model (1.1). The models discussed here improve the fit, but not by much.

### 6.1 A Readout Noise Model

We consider a refined version of Google's model that treats carefully the effect of readout errors. (Readout errors are discussed in Section VI.D in the Supplement to [3].) For this study, we define the *total gate fidelity* $\phi_g$ as the probability that no errors occur in the gate operations. Two crucial ingredients of the Google experiment are the prediction of the fidelity (2.1) based on the fidelities of the individual components, and the statistical estimator (2.3) of the fidelity based on the experimental bitstrings. The statistical assumptions that allow good predictions of the fidelity also allow prediction of the total gate fidelity $\phi_g$, and an extension of the Google estimator allows an estimation of $\phi_g$ from the experimental bitstrings.

We need to consider first how to estimate $\phi_g$ from Equation (2.1), and then how to estimate $\phi_g$ from the experimental bitstrings. The first task is easy. Equation (2.1) (Google's Formula (77)) gives us an approximate description of the total gate fidelity $\phi_g$ via its relation with the fidelity $\phi$:

$$\phi = \phi_g \cdot \prod_{q \in \mathcal{Q}} (1 - e_q).$$

In this section we use a further simplification based on (2.2), namely, $\phi = \phi_g(1 - 0.038)^n$, where $n$ is the number of qubits. Let $\phi_{ro} = \phi_g - \phi$ be the probability that there are no gate errors but there are some readout errors.

We now discuss the estimation of $\phi_g$ based on the experimental bitstrings, a task that fits our general framework. As before, the quantum circuit $C$ defines a probability $\mathcal{P}_C(\mathbf{x}^{(i)}) = w_i$, also denoted by $w_{\mathbf{x}^{(i)}}$, $i = 1, \dots, M = 2^n$, where $\mathbf{x}^{(i)} \in \{0,1\}^n$. The Hamming distance between such strings is relevant to the nature of noise considered in this section. Assuming that in the computation process there are no gate errors, we can observe a bitstring $\mathbf{x}$ in two ways:

1. Reading the output without error, and therefore observing $\mathbf{x}$ with probability $\mathcal{P}_C(\mathbf{x}) = w_{\mathbf{x}}$, or
2. Observing $\mathbf{x}$ because the true output is $\mathbf{x} \oplus \mathbf{y}$ for some $\mathbf{y} \neq \mathbf{0}$, an event whose probability is $\mathcal{P}_C(\mathbf{x} \oplus \mathbf{y})$ (where $\oplus$ denotes the XOR operation, that is, mod-2 addition) and then readout errors occur exactly in the coordinate $i$ in which $y_i = 1$, an event whose probability is $q^{|\mathbf{y}|}(1-q)^{n-|\mathbf{y}|}$, where $q$ is the probability of an individual readout error.

We assume independent errors with a common probability $q$, and so the components of $\mathbf{y}$ are iid Bernoulli($q$) and the probability of $\mathbf{y}$ is defined as $\mathcal{B}_q(\mathbf{y}) := \mathcal{B}_q(|\mathbf{y}|) = q^{|\mathbf{y}|}(1-q)^{n-|\mathbf{y}|}$. We can take $q = 0.038$; see Equation (2.2) and [3]. Let $D := P(\mathbf{y} \neq \mathbf{0}) = \sum_{|\mathbf{y}| \neq 0} \mathcal{B}_q(\mathbf{y}) = 1 - (1-q)^n$, and let $\mathcal{N}_C^{ro}(\mathbf{x})$ denote the probability of observing $\mathbf{x}$ due to the second reason (point 2. above), conditioned on the existence of readout errors only (and no gate errors). Then

$$\begin{aligned}\mathcal{N}_C^{ro}(\mathbf{x}) &= \frac{1}{D}\sum_{\mathbf{y}\in\{0,1\}^n,\mathbf{y}\neq\mathbf{0}} \mathcal{P}_C(\mathbf{x}\oplus\mathbf{y})\mathcal{B}_q(\mathbf{y}) \\ &= \frac{1}{D}\sum_{\mathbf{y}\in\{0,1\}^n,\mathbf{y}\neq\mathbf{0}} w_{\mathbf{x}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y}).\end{aligned} \tag{6.1}$$

We consider the noise model for the quantum circuit $C$ that produces a sample of size $N$ of $n$-strings, $\widetilde{\mathbf{x}}_1,\dots\widetilde{\mathbf{x}}_N$, by drawing $\mathbf{x}^{(i)}$'s independently $N$ times where the probability of drawing $\mathbf{x}\in\{0,1\}^n$ is given by

$$\begin{aligned}\pi(\widetilde{\mathbf{x}}^{(j)} = \mathbf{x}) &= \phi\mathcal{P}_C(\mathbf{x}) + \phi_{ro}\mathcal{N}_C^{ro}(\mathbf{x}) \\ &\quad + (1-\phi_g)/M, \quad j=1,\dots,N.\end{aligned} \tag{6.2}$$

This is a refinement of Google's basic model (1.1), offering a more detailed description of the sampling probability when there are only readout errors and no gate errors. When there are gate errors, we still assume that the samples have the uniform distribution. As explained already in [3] and in the previous sections, such a noise model will not make a difference in estimating $\phi$, based on the main signal $\mathcal{P}_C(x)$), but it will give us an opportunity for an independent estimation of $\phi_{ro}$ (and hence also $\phi$ itself) based on the secondary signal $\mathcal{N}_C^{ro}(\mathbf{x})$. Note also that (6.2) is a special case of our general model (1.5) from the Introduction.

Using the notation $v_i := \sum_{|\mathbf{y}|\neq\mathbf{0}} w_{\mathbf{x}^{(i)}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y})/D$ we have $\sum_{i=1}^M v_i = 1$, and recalling that $\mathcal{P}_C(\mathbf{x}^{(i)}) = w_{\mathbf{x}^{(i)}} = w_i$, it is easy to see that Equation (6.2) is equivalent to sampling $N$ times (with replacement) with probabilities

$$\begin{aligned}\pi(\mathbf{x}^{(i)}) &= \phi w_i + \phi_{ro}v_i \\ &\quad + (1-\phi_g)/M, \quad i=1,\dots,M.\end{aligned} \tag{6.3}$$

Note that $v_i$ contains all the $w_j$'s except for $w_i$. The variables $w_i$ and $w_j$, being coordinates of the Dirichlet distribution, are very weakly dependent when $M$ is large. Thus, the components of (6.3) are almost independent. In particular, the relevant fact here is that $E(w_i w_j) = 1/[M(M+1)]$ for $i\neq j$ (see (1.2)), which for large $M$ is very close to $1/M^2$ that would obtain had they been independent.

### 6.2 Estimating the Total Gate Fidelity

Let $\widetilde{\mathbf{x}}^{(j)}$ denote an observation, $j=1,\dots,N$, corresponding to $(\widetilde{w}_j,\widetilde{v}_j)$ where $\widetilde{v}_j = \sum_{|\mathbf{y}|\neq 0} w_{\widetilde{\mathbf{x}}^{(j)}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y})/D$. To estimate $\phi_g$ or $\phi_{ro}$, define a Google-type estimator

$$W = \frac{M}{N}\sum_{j=1}^N \widetilde{v}_j - 1,$$

and again let $U = \frac{M}{N}\sum_{j=1}^N \widetilde{w}_j - 1$. Note that computing $W$ requires knowledge of all $\mathcal{P}_C(\mathbf{x}^{(i)}) = w_i$ and not just those in the sample. Taking expectation of $W$ with respect to the sampling probabilities of (6.3) conditioned on $\{w_i\}_1^M$ and $\{v_i\}_1^M$ and using $\phi_g = \phi + \phi_{ro}$, we have

$$\begin{aligned}E^w W &= M\phi\sum_{i=1}^M v_i w_i + M\phi_{ro}\sum_{i=1}^M v_i^2 \\ &\quad - (\phi+\phi_{ro}), \\ E^w U &= M\phi\sum_{i=1}^M w_i^2 + M\phi_{ro}\sum_{i=1}^M w_i v_i \\ &\quad - (\phi+\phi_{ro}).\end{aligned} \tag{6.4}$$

For given $\{w_i\}_1^M$ and $\{v_i\}_1^M$ it is easy to see that $W$ and $U$ are averages of i.i.d. variables, and hence they are consistent estimators of their expectations. The method of moments (see Section 4.1) applied here consists of replacing $E^w W$ by the statistic $W$ and $E^w U$ by $U$ in the above two equations and solving the resulting linear system for $\phi$ and $\phi_{ro}$.

We next consider $EW = EE^w W$. Define $G := \frac{M}{M+1}\times\{[q^2+(1-q)^2]^n - 2(1-q)^n + 1\}$, and recall that $D = 1-(1-q)^n$. In the Appendix, we show that

$$EW = \phi_{ro}[G/D^2 - 1] - \frac{1}{M+1}\phi, \tag{6.5}$$

and neglecting the last term, the estimator

$$\widetilde{\phi}_{ro} := \frac{1}{G/D^2 - 1}W \tag{6.6}$$

is nearly unbiased for $\phi_{ro}$ when expectation is taken with respect to both the sampling in (6.3) and with respect to all $\{w_i\}_1^M$ and $\{v_i\}_1^M$, that is, in the same sense that Google's $U$ is nearly unbiased (see (4.7)). The average results for $U$ and $\widetilde{\phi}_{ro}$ for the ten files ($n=12$) are (0.3674, 0.1956).

The computation of $\mathrm{Var}(W)$ is rather cumbersome and will not be presented. For data analysis purposes, we shall compute it by simulations.

Using $Ew_i v_i = E(w_i w_j) = 1/(M(M+1))$ for $i\neq j$, it is easy to see that $U$ is nearly unbiased for $\phi$ since $EU = \phi\frac{M-1}{M+1} - \phi_{ro}/(M+1)$.

### 6.3 Maximum Likelihood Estimation of Fidelity

We now consider the MLE of $\boldsymbol{\phi} = (\phi,\phi_{ro})$. We rewrite the sampling rule (6.3) as

$$\begin{aligned}\pi(\mathbf{x}^{(i)}) &= \phi(w_i - 1/M) + \phi_{ro}(v_i - 1/M) \\ &\quad + 1/M, \quad i=1,\dots,M.\end{aligned} \tag{6.7}$$

The log-likelihood function is

$$\ell(\boldsymbol{\phi}) = \sum_{j=1}^N \log[\phi(\widetilde{w}_j - 1/M) + \phi_{ro}(\widetilde{v}_j - 1/M) + 1/M].$$

The maximum likelihood estimator of $\boldsymbol{\phi} = (\phi, \phi_{ro})$ is obtained by computing the gradient $\mathbf{f}(\boldsymbol{\phi})$ and solving the equation $\mathbf{f}(\boldsymbol{\phi}) = (\frac{\partial}{\partial \phi}\ell(\boldsymbol{\phi}), \frac{\partial}{\partial \phi_{ro}}\ell(\boldsymbol{\phi})) = \mathbf{0}$, where

$$\frac{\partial}{\partial \phi}\ell(\boldsymbol{\phi}) = \sum_{j=1}^{N} \frac{(\widetilde{w}_j - 1/M)}{\phi(\widetilde{w}_j - 1/M) + \phi_{ro}(\widetilde{v}_j - 1/M) + 1/M}$$

and

$$\frac{\partial}{\partial \phi_{ro}}\ell(\boldsymbol{\phi}) = \sum_{i=1}^{N} \frac{(\widetilde{v}_j - 1/M)}{\phi(\widetilde{w}_j - 1/M) + \phi_{ro}(\widetilde{v}_j - 1/M) + 1/M}.$$

Starting with root-$N$ consistent estimators such as the moment estimators described above, we approach the MLE's by the Newton–Raphson algorithm. As mentioned before, asymptotically, one iteration suffices. For the Newton–Raphson algorithm we need the 2 × 2 Hessian matrix $J(\boldsymbol{\phi})$, which is easy to compute by taking second derivatives of the derivatives given above. The Newton–Raphson iteration can now be written as $\boldsymbol{\phi}_1 = \boldsymbol{\phi}_0 - \mathbf{f}(\boldsymbol{\phi}_0)J^{-1}(\boldsymbol{\phi}_0)$.

We estimated $\phi$ and $\phi_{ro}$ under the sampling model (6.2)–(6.3) for ten Google files for $n = 12$ and sample size $N = 500{,}000$. For these files formula (2.1) gives $\phi = 0.3862$. The average of the MLE estimates of the pair $(\phi, \phi_{ro})$ over the ten files is $(0.3687, 0.1958)$. The average MLE computed as in Section 4.4 with sampling according to (1.1) comes out to the same value 0.3687, and individual estimates in the ten files differed slightly in the third or fourth digit after the decimal point when estimated according to the two models.

### 6.4 Related Noise Models

We remark briefly on other noise models. In (6.1) the random noises $\mathcal{N}_C^{ro}(\mathbf{x})$ for different $\mathbf{x}$'s are dependent, with stronger dependence between $\mathbf{x}$'s that are close together in the Hamming distance. The noise term $\mathcal{N}_C^{ro}(\mathbf{x})$ is a sum of $w_i$'s with weights. Using the representation $w_i = z_i / \sum_{j=1}^{M} z_j$ with iid $z_j$'s, it can be shown, for example, by Lyapunov's central limit theorem, that for fixed $\mathbf{x}$ the sum composing the error $\mathcal{N}_C^{ro}(\mathbf{x})$ is approximately normal for large $M$. A Gaussian assumption simplifies the dependence structure across $\mathbf{x}$'s and allows further calculations. One can also consider replacing the Binomial distribution $\mathcal{B}_q(\mathbf{y})$ in (6.1) by another, for example, a distribution that is supported only on $|\mathbf{y}| \le k$, thus taking into account only neighbors of $\mathbf{x}$ having Hamming distance $\le k$, with, say, $k = 1$ or 2 or so. This approximation can simplify the necessary computations.

Finally, as it turns out, for the actual readout errors of the Google device the probability $q_1$ that 1 is read as 0 is 0.055 and the probability $q_2$ that 0 is read as 1 is 0.023. This leads to the following sampling model:

$$\begin{aligned} \pi(\mathbf{x}) = \phi_g \sum_{\mathbf{y}\in\{0,1\}^n} &\big[\mathcal{P}_C(\mathbf{x}\oplus\mathbf{y}) \\ &\times q_1^a q_2^b (1-q_1)^c (1-q_2)^d\big] \\ &+ (1-\phi_g)\mathcal{B}_q(\mathbf{x}), \end{aligned} \tag{6.8}$$

where $q_1 = 0.055$, $q_2 = 0.023$, $a = |\{i : y_i = 1 \text{ and } x_i = 0\}|$, $b = |\{i : y_i = 1 \text{ and } x_i = 1\}|$, $c = |\{i : y_i = 0 \text{ and } x_i = 1\}|$, $d = |\{i : y_i = 0 \text{ and } x_i = 0\}|$, and $\mathcal{B}_q(\mathbf{x}) = \mathcal{B}_q(|\mathbf{x}|) = q^{|\mathbf{x}|}(1-q)^{n-|\mathbf{x}|}$, where now $q = (1 - q_1 + q_2)/2 = 0.484$.

The first term of (6.8) represents the probability that $\mathbf{x}$ is sampled, conditioned on no gate errors. Here $\mathcal{P}_C(\mathbf{x}\oplus\mathbf{y})$ is the probability that without noise $\mathbf{x}\oplus\mathbf{y}$ will be produced by the circuit. If, for example, $x_i = 0$ and $y_i = 1$ (or $y_i = 0$) then the $i$th coordinate will be read as 0 if a readout error occurs with probability $q_1$, corresponding to $a$ above (if no readout error occurs, with probability $1 - q_2$, corresponding to $d$).

We continue to assume that the effect of gate errors is to replace the desired distribution with a uniform probability but the asymmetric readout errors make a difference also there. If uniformly distributed $n$-vectors are read with errors such that 1 is read as 0 with probability $q_1 = 0.055$ and 0 is read as 1 with probability $q_2 = 0.023$ then their distribution is no longer uniform. Instead, for each coordinate the probability of 1 becomes $(1 - 0.055)/2 + 0.023/2 = 0.484$, and the probability of 0 is 0.516.

When $n = 12$ we have by (2.1) $\phi = 0.3862$ and then $\phi_g = \phi/(1 - .038)^{12} = 0.6148$. Using ten Google samples of size $N = 500{,}000$ with $n = 12$ the average MLE of $\phi_g$ is 0.5880. Using (6.8), we computed the MLE with $(\phi_g, q_1, q_2)$ as parameters, and obtain the estimates $(0.5571, 0.0465, 0.0196)$ for Google's file 1 (with similar results for the other files for $n = 12$). Thus, our estimate of $\phi_g$ and the value derived from (2.1) are all within at most 10% of each other, while our estimates of $q_i$ differ from Google's numbers by about 15%. The fact that the fidelity estimation based on (6.2) and (6.8) largely agree can be seen as a demonstration of the robustness discussed in Section 4.3. The model (6.8) can further be improved by taking into account the individual readout errors for different qubits.

## 7. CONFIDENCE INTERVALS

In this section, we compute confidence intervals for $\phi$ for the three estimators discussed in Section 4. We start with $V = [\frac{1}{N}\sum_{j=1}^{N} M\widetilde{w}_j - 1]/(Mw^{(2)} - 1)$ of (4.6), which is unbiased. For the computation of $V$, knowledge of the sample values $\{\widetilde{w}_j\}_1^N$ is not enough, and it requires knowledge of $w^{(2)}$, which is a function of $\{w_i\}_1^M$. This differs from the computation of $U$ and the MLE, which are functions of $\{\widetilde{w}_j\}_1^N$, and hence statistics. As mentioned before, this distinction is relevant if for some reason only the sample values $\{\widetilde{w}_j\}_1^N$ are available, and this holds for some of Google's data files. The variance of $V$ conditional on the realization $\{w_i\}_1^M$ is given in (4.9). We denote it now by

$$\begin{aligned}\sigma_N^2(V;\phi,w^{(2)},w^{(3)})\\ &= \frac{1}{N(Mw^{(2)}-1)^2}\big[\phi\big(M^2w^{(3)}-3Mw^{(2)}+2\big)\\ &\quad - \phi^2\big(Mw^{(2)}-1\big)^2 + Mw^{(2)}-1\big].\end{aligned}$$

We assume that the realization $\{w_i\}_1^M$ is known, and therefore so are $w^{(2)}$ and $w^{(3)}$. If in a simulation $\phi$ is known, and we study the distribution of the estimator $V$, then we can use the known $\phi$ in computing $\sigma_N^2(V) = \sigma_N^2(V;\phi,w^{(2)},w^{(3)})$. If $\phi$ is unknown, it is standard practice to plug in an estimator, such as $V$ itself. Thus we assume that $\sigma_N(V;\phi,w^{(2)},w^{(3)})$ is given. Since $V$ is based on a sum of iid variables it is asymptotically normal, and a standard 95% confidence interval is

$$\phi \in V \pm 1.96\sigma_N\big(V;\phi,w^{(2)},w^{(3)}\big). \tag{7.1}$$

Both the normal and plug-in approximations suggest rounding 1.96 to 2; we shall keep 1.96 only because of its connotation. This is a *conditional* (on the realization $\{w_i\}_1^M$) confidence interval.

Suppose now that we have $L$ independent samples, all taken under the same $\phi$ according to (1.1). These samples may arise from a single realization $\{w_i\}_1^M$ or from several different ones, all known. If we have estimates $V_1,\dots,V_L$ then a natural approach is to first estimate $\phi$ by $\overline{V} = \frac{1}{L}\sum_{i=1}^L V_i$ and plug it in to obtain variance estimates $\sigma^2_{N_1}(V_1),\dots,\sigma^2_{N_L}(V_L)$. We then replace $\overline{V}$ by the variance-minimizing convex combination of the estimators $\overline{\overline{V}} = \sum_{i=1}^L \frac{V_i}{\sigma^2_{N_i}(V_i)} / \sum_{i=1}^L \frac{1}{\sigma^2_{N_i}(V_i)}$, whose variance is $\frac{1}{\sum_{i=1}^L 1/\sigma^2_{N_i}(V_i)}$. This process can be iterated, that is, plug in $\overline{\overline{V}}$ to reestimate the variances, etc.; however, we shall not pursue this. We obtain the conditional confidence interval

$$\phi \in \overline{\overline{V}} \pm 1.96\sqrt{\frac{1}{\sum_{i=1}^L 1/\sigma^2_{N_i}(V_i)}}, \tag{7.2}$$

where in $\sigma^2_{N_i}(V_i) = \sigma^2_{N_i}(V_i;\phi,w_i^{(2)},w_i^{(3)})$ we set $\phi = \overline{\overline{V}}$.

Consider now the maximum likelihood estimator MLE discussed in Section 4.4. To compute the MLE, as well as $U$, only the sample values $\{\widetilde{w}_i\}_1^N$ are needed. We can repeat all the above steps and obtain the same kind of confidence intervals as in (7.1) and (7.2) by now replacing $V$ by the MLE and $\sigma_N^2(V;\phi,w^{(2)},w^{(3)})$ by

$$\sigma_N^2\big(\mathrm{MLE};\phi,w^{(2)},w^{(3)}\big) = 1/N\sum_{i=1}^M \frac{(w_i-1/M)^2}{\phi w_i + (1-\phi)/M};$$

see (4.15). Since Google's estimator $U$ is conditionally biased, we cannot apply to it the same procedure to obtain a conditional confidence interval. Unconditional confidence intervals are discussed next.

We now consider the above estimators when both $\{\widetilde{w}_i\}_1^N$ and $\{w_i\}_1^N$ are random, and we do not condition on them. We refer to intervals as *unconditional*. This seems to be the approach taken by Google [3]. In this case, the estimators considered above are practically unbiased (for large $M$) and their (approximate) variances are given in (4.12) and (4.16) (where the approximate conditional and unconditional variances are the same as those discussed in Sections 4.2 and 4.4) and in (4.13), where the variance reflects the conditional bias of $U$.

As above, we consider $L$ samples from $L$ different circuits with the same parameter $\phi$, and $L$ estimators of $\phi$. In this case Google's approach, in principle, is to average these estimators, leading to the confidence interval

$$\phi \in \overline{U} \pm 1.96\sqrt{\frac{1}{LN}\big(2\phi-\phi^2+1\big) + \frac{20}{LM}\phi^2}, \tag{7.3}$$

where $\overline{U} = \frac{1}{L}\sum_{\ell=1}^L U_\ell$ is the average of the different estimators. For the estimator $V$, we have in the same way

$$\phi \in \overline{V} \pm 1.96\sqrt{\frac{1}{LN}\big(2\phi-\phi^2+1\big)}, \tag{7.4}$$

where $\overline{V} = \frac{1}{L}\sum_{\ell=1}^L V_\ell$, and for the MLE, we have by the results of Section 4.4

$$\phi \in \overline{\mathrm{MLE}} \pm 1.96\sqrt{\frac{1}{LN\int_0^\infty \frac{(z-1)^2}{\phi z+1-\phi}e^{-z}\,dz}}. \tag{7.5}$$

Figure 7 shows confidence intervals computed using (7.3) and (7.5) with $L = 10$ and $N = 500{,}000$ for 10 Google files for each $n$. Note that the scale of the $y$-axis changes with $n$. We see that for $n = 12$ and 14 the estimate $\overline{U}$ (the turquoise dot) is not in the confidence interval around the MLE, and the latter confidence interval is much smaller than the one around $\overline{U}$. As $n$ increases the estimates and their confidence intervals get closer together.

## 8. TESTING MODELS AND DISTRIBUTIONS

To demonstrate quantum supremacy, a convincing separation of the fidelity estimator from zero is crucial. But there are various reasons to try to understand the actual state of the quantum computer from its experimental output and its relation to probabilities it represents. There are also various ways, some mentioned in [3], to compare the values $\mathcal{P}_C(\mathbf{x})$ for bitstrings obtained in an experiment with the size-biased theoretical distribution, and this goes beyond the estimator of the fidelity. For small values of $n$, Google's sample sizes $N$ are rather large compared to $M = 2^n$ and we can test the goodness of fit of the experimental output to the sampling probabilities.

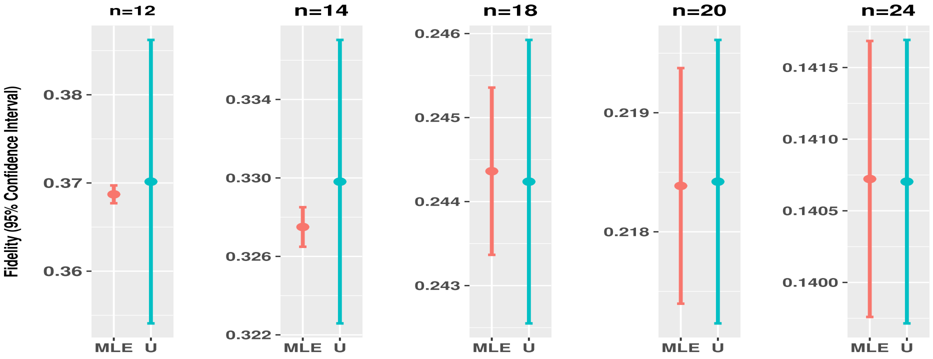

FIG. 7. *Unconditional confidence intervals for the fidelity based on* 10 *files for circuits of different sizes* $n$.

Given a sample $\mathcal{S}_{\mathbf{x}} = \{\tilde{\mathbf{x}}^{(j)}\}_1^N$, with the sampling rule (1.1), let $n_i$ denote the number of sampled bitstrings that equal $\mathbf{x}^{(i)}$, that is, $n_i = |\{j : \tilde{\mathbf{x}}^{(j)} = \mathbf{x}^{(i)}\}|$, $i = 1, \dots, M$. We have $\sum_{i=1}^M n_i = N$. As explained in Section 1 there is an association between the probabilities and bitstrings $\mathbf{x}^{(i)}$ expressed by $\mathcal{P}(\mathbf{x}^{(i)}) = w_i = w_{\mathbf{x}^{(i)}}$. Assuming all $\{w_i\}_1^M$ known, set $\pi_i = \phi w_i + (1-\phi)/M$, $i = 1, \dots, M$ according to the sampling model (1.1). Pearson's chi-square test statistic is

$$\mathcal{X}^2 = \sum_{i=1}^{M} \frac{(n_i - N\pi_i)^2}{N\pi_i},$$

and its asymptotic distribution for fixed $M$ and large $N$ is chi-square with $M-1$ degrees of freedom (or $M-2$ if $\phi$ is estimated). If the sample size $N$ does not suffice, we can merge cells and probabilities, starting with cells with small $\pi_i$. The above test is relevant when we wish to verify the sampling model for a given realization $\{w_i\}_1^M$. We looked at the 10 files given in [3] with $n = 12$ and sample size $N = 500{,}000$.

Using the $w_i$'s given in [3] and the MLE of the sample for $\phi$ we determine $\pi_i$ and calculate $\mathcal{X}^2$ for Google's samples. We obtain very large $\mathcal{X}^2$ values, of the order of 40,000. This is extremely significant and the $p$-value is practically zero (since, for example, the level $\alpha = 10^{-15}$ critical values for $M - 2 = 4{,}094$ degrees of freedom is about 4855). A rough calculation shows that the average deviation of the cells' empirical probabilities from the theoretical $\pi_i$ is about 0.25 standard deviations, and that the $\mathcal{X}^2$ statistic is sufficiently large to reject the model with about a tenth of the sample size used here, and so the large $\mathcal{X}^2$ is not only due to the large sample size. We computed the value of $\phi$ that minimizes the $\mathcal{X}^2$ statistic in the 10 Google files, and find that it closely agrees with the MLE; hence the large $\mathcal{X}^2$ is not due to our choice of $\phi$ in determining $\pi_i$. The $\mathcal{X}^2$ statistics for the different readout models discussed in Section 6 were smaller by only about 10%, and still extremely significant. The above test, the results on the estimator $T$ in Section 4.7, and further results below, all indicate that the sampling model (1.1) does not provide an adequate description of the data. Thus, additional models should be explored.

In Figure 8, the two left-hand plots show $N\pi_i$ against the corresponding empirical frequencies for two files with $n = 12$. These are compared (the two right-hand plots) to similar plots using the same $\{w_i\}$'s for a sample generated by (1.1) with $\phi = 0.3687$, which is the MLE estimated from the 10 files combined. One can see with the naked eye that Google's sample is much more variable than a sample obtained according to the theoretical model. The truncation on the left is due to the fact that $N\pi_i \geq N(1-\phi)/M = 77$ in the present case.

We plotted various histograms of Google's samples (for which the $\mathcal{X}^2$ values were huge) and compared them to histograms of simulated samples based on the same $\{w_i\}_1^M$ and the MLE (for which $\mathcal{X}^2$ was around 4000 as expected). Histograms involve smoothing and with the naked eye we could not tell these histograms apart, in spite of the large difference in the $\mathcal{X}^2$ values. Such histograms, with 200 cells each, are given in Figure 9. The reader is invited to choose which histogram corresponds to Google's sample, and which to our simulated sample using (1.1) with a non-significant $\mathcal{X}^2$ value.[4] To understand the theoretical distribution of such samples, recall that in Google's model $w_i = z_i / \sum_{j=1}^M z_j = \frac{z_i/M}{\sum_{j=1}^M z_j/M} \approx z_i/M$, where the approximation follows from the fact that $z_j \sim \text{Exp}(1)$ with mean $= 1$, and so the denominator $\to 1$ as $M \to \infty$. If we approximate the sampling model (1.1) by $\pi(\mathbf{x}^{(i)}) = \pi(z_i) = \phi z_i/M + (1-\phi)/M$ then, by Proposition 3.1, the resulting sample $\{\tilde{z}_{\mathbf{x}_i}\}$ will be approximately from the distribution with density $\phi z e^{-z} + (1-\phi)e^{-z}$, which is a mixture of Gamma(2,1) (the size-biased Exp(1)) and Exp(1) itself. This approximation is used in [3].

Similarly, samples obtained by (1.1) from a Dirichlet vector $w_1, \dots, w_M$ are expected to have a distribution that

[4] On the right-hand side is Google's sample; on the left-hand side our simulated sample.

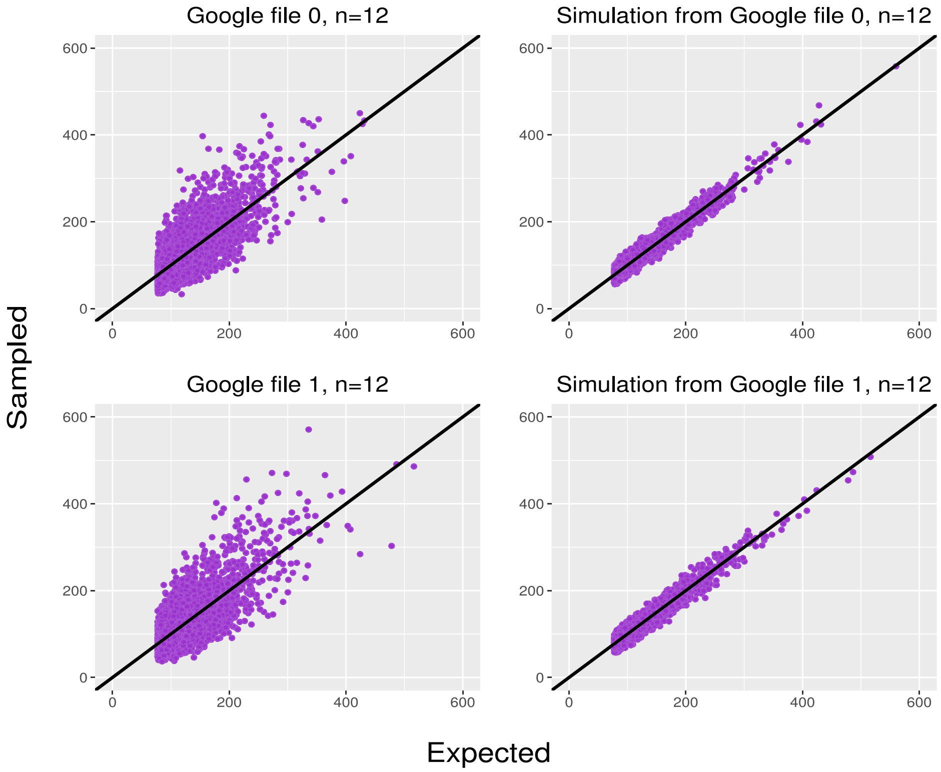

FIG. 8. *The left-hand side scatterplots display theoretical vs. empirical frequencies of two Google files with $n = 12$. The right-hand side scatterplots display theoretical vs. our simulated empirical frequencies according to* (1.1).

is close to the distribution whose density is a mixture of Beta densities given by

$$\phi M(M-1)t(1-t)^{M-2} + (1-\phi)(M-1)(1-t)^{M-2}. \tag{8.1}$$

To see this note that the marginal distribution of the Dirichlet vector is Beta(1, $M$-1) with density $(M-1)(1-t)^{M-2}$ for $t \in (0, 1)$. Therefore, the size-biased distribution has the Beta density $M(M-1)t(1-t)^{M-2}$, and sampling by (1.1) implies that the distribution of the sample $\widetilde{w}_1, \dots, \widetilde{w}_N$ should be a discrete distribution (taking values from the set $\{w_i\}_1^M$) that is close to the above mixture by Proposition 3.1. This is a mixture of Beta distributions, with the first part being the size-biased version of the second (see Section 3). From $\widetilde{z}_{\mathbf{x}_i} \sim \phi z e^{-z} + (1-\phi)e^{-z}$ we see that $\widetilde{z}_{\mathbf{x}_i}/M \sim \phi M^2 t e^{-Mt} + (1-\phi)Me^{-Mt}$, which is seen to be approximately equal to (8.1) by using $e^{-t} \approx 1-t$.

The above analysis can be applied to more general noise models and we can consider size-biased distributions not only with respect to the Dirichlet or exponential distributions. If a proposed noise model has several ingredients, for example,

$$\mathcal{Q}(\mathbf{x}) = \phi_1 \mathcal{P}_C(\mathbf{x}) + \phi_2 \mathcal{N}_2(\mathbf{x}) + \phi_3 \mathcal{N}_3(\mathbf{x}) + \cdots,$$

then given a sample $\widetilde{\mathbf{x}}^{(1)}, \dots, \widetilde{\mathbf{x}}^{(N)}$, the statistical tools of the previous sections can be used to estimate the different $\phi_i$'s, and we can test the distribution of the sample as above for a fixed given realization or over different realizations.

It is important to note that the fidelity estimators (apart from $T$ of Section 4.7) and the size-biased distributions,

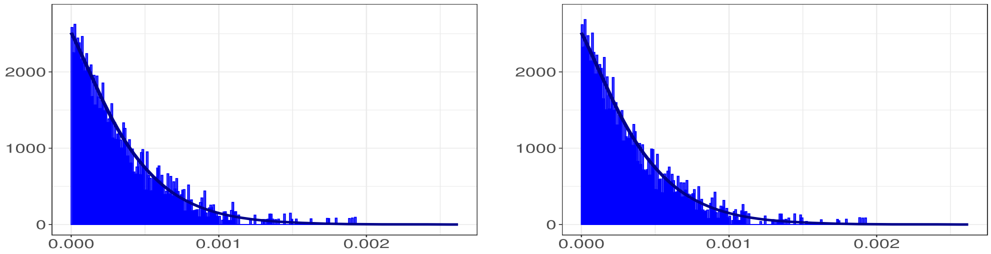

FIG. 9. *Histogram of a simulated sample $\{\widetilde{w}_i\}_1^N$ with $N = 500{,}000$ from $\{w_i\}_1^N$ of Google's file No.* 0 ($n = 12$) *according to* (1.1) *with $\phi = 0.3862$, and histogram of Google's empirical frequencies. The black curves represent the asymptotic density of* (8.1).

when examined by standard histograms, are robust to various radical changes of the underlying probability distribution (see Sections 3.2 and 4.3). For example, they will not be affected if we exclude from the sample all bitstrings with an odd number of 1s. They will also be robust to various different noise models. Such radical changes will greatly affect the empirical distribution and the outcomes of the chi-square test.

Returning to Figure 9, we note that the fact that the estimator $T$ (Section 4.7) is considerably larger for the Google files compared to the simulated samples generated by the theoretical models means that if we zoom into the individual cells, we see that the samples generated by the quantum computers are more clustered than the samples generated by the theoretical models.

## 9. CONCLUSION

We describe in statistical language the ideas and problems involved in quantum supremacy demonstrations, and study central statistical aspects of the analysis of quantum computers' output involved in such demonstrations. The scientific and technological aspects of Google's demonstration are beyond the scope of this paper.

We study various related fidelity estimators. Google's estimator $U$ of the fidelity, conditioned on a realization of the probabilities $\{w_i\}_1^M$, is biased in a way that depends on the realization. We study two unbiased estimators, MLE and $V$, that are more suitable for relatively small circuits. As a fidelity estimator of an unknown realization, that is, when $\{w_i\}_1^M$ are considered random (with some assumptions), the estimator $U$ is unbiased but has a larger variance compared to $V$ and the MLE. When the number of qubits increases and the fidelity decreases, the gap between these estimators diminishes and thus our results largely confirm the Google team's choice of the estimator $U$ for large-scale circuits.

Based on Google's readout analysis we considered various readout error models that refine Google's basic model, and the associated estimators of the fidelity. A preliminary study of the Google data on 12 and 14 qubits suggests that neither Google's basic noise model nor our refined readout error model fits the observed data. An extension of the analysis beyond 12 and 14 qubits can shed light on the statistical assumptions behind the Google fidelity predictions and on other aspects of their experiment. We note that there are further statistical aspects of the experiments described in [3], mainly those related to the quality of individual components of the circuit and to the process of "calibration," which are not considered here.

Finally, we expect that our study can contribute to the understanding of the nature of noise in noisy intermediate-scale quantum computers. Sampling tasks are natural candidates for testing and proving the potential of quantum computers. Therefore, our statistical analysis is expected to be useful for various near-term experimental efforts for NISQ computers and future attempts for "quantum supremacy" demonstrations.

## APPENDIX: SOME PROOFS

PROOF OF (4.22), PROPOSITION 4.1. In order to compute $\mathrm{Var}(U_{g(w)=\log(w)})$ we use the approximation (valid for large $M$) $w_i = z_i/M$, and we assume $z_i \sim \mathrm{Exp}(1)$. We need the following facts:

$$\begin{aligned}
&\int_0^\infty \log(z)(z-1)e^{-z}\,dz = 1,\\
&\int_0^\infty \log(z)e^{-z}\,dz = -\gamma\\
&\approx -0.5772,\\
&\int_0^\infty (\log(z)+\gamma)^2(z-1)e^{-z}\,dz = 0 \quad \text{and}\\
&\int_0^\infty (\log(z)+\gamma)^2 e^{-z}\,dz = \pi^2/6\\
&\approx 1.6449.
\end{aligned} \tag{A.1}$$

where $\gamma$ is Euler's constant. The first equation in (A.1) shows that in this case $A = 1$; see (4.21). The second equation, with $w = z/M$ implies that $B = -\gamma - \log(M)$, and $EY = \phi - \gamma - \log(M)$, where $Y$ is defined in the beginning of the proof of Proposition 4.1 with $g(w) - \log(w)$. Thus, using the approximation $w_i = z_i/M$ in the third equality below,

$$\begin{aligned}
&\mathrm{Var}[\log(\tilde{w}_j)]\\
&\quad = EE^w[(\log(\tilde{w}_j) - EY)^2 \mid \{w_i\}]\\
&\quad = E\left\{\sum_{i=1}^M (\log(w_i) - EY)^2[\phi w_i + (1-\phi)/M]\right\}\\
&\quad \approx E\{(\log(z/M) - EY)^2[\phi z + (1-\phi)]\}\\
&\quad = \int_0^\infty (\log(z) - \phi + \gamma)^2[\phi(z-1)+1]e^{-z}\,dz,
\end{aligned}$$

and the above integrals imply

$$\begin{aligned}
&\int_0^\infty (\log(z) - \phi + \gamma)^2[\phi(z-1)+1]e^{-z}\,dz\\
&\quad = \int_0^\infty [(\log(z)+\gamma)^2 + \phi^2 - 2\phi(\log(z)+\gamma)]\\
&\qquad \times [\phi(z-1)+1]e^{-z}\,dz\\
&\quad = \int_0^\infty [(\log(z)+\gamma)^2]e^{-z}\,dz + \phi^2 - 2\phi^2\\
&\qquad + 2\phi\gamma - 2\phi\gamma = \pi^2/6 - \phi^2.
\end{aligned}$$

With $A$ and $B$ calculated above, we get $U_{g(z)=\log(z)} = \frac{1}{N}\sum_{i=1}^N \log(\tilde{w}_i) + \gamma + \log(M)$ and $\mathrm{Var}(U_{f(z)=\log(z)}) \approx \frac{1}{N}(\pi^2/6 - \phi^2) \approx \frac{1}{N}(1.6449 - \phi^2)$. □

PROOF OF (6.5). We claim that the expectation $EW$ taken with respect to the sampling rule (6.2) and then over $\{w_i\}_1^M$ can be expressed after some calculations as

$$\begin{aligned} EW &= \frac{M}{M+1}\phi \\ &\quad + M^2\phi_{ro}E\Big[\Big(\sum_{|\mathbf{y}|\neq 0} w_{\mathbf{x}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y})/D\Big)^2\Big] \\ &\quad + (1-\phi_g) - 1 \\ &= M^2\phi_{ro}E\Big[\Big(\sum_{|\mathbf{y}|\neq 0} w_{\mathbf{x}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y})/D\Big)^2\Big] \\ &\quad - \phi_{ro} - \frac{1}{M+1}\phi, \end{aligned} \tag{A.2}$$

where $D := \sum_{|\mathbf{y}|\neq 0}\mathcal{B}_q(\mathbf{y}) = 1-(1-q)^n$. To see this take expectation over $E^wW$ in (6.4) whose $M$ terms all have the same expectation; therefore, we can multiply by $M$ instead of summing. The term $\phi$ in (A.2) arises from

$$\begin{aligned} M\phi\sum_{i=1}^{M} E(v_iw_i) &= M^2\phi E\Big[\sum_{|\mathbf{y}|\neq 0} w_{\mathbf{x}^{(i)}}w_{\mathbf{x}^{(i)}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y})/D\Big] \\ &= \frac{M}{M+1}\phi, \end{aligned}$$

since $E(w_{\mathbf{x}^{(i)}}w_{\mathbf{x}^{(i)}\oplus\mathbf{y}}) = E(w_iw_j) = 1/(M(M+1))$ by (1.2) (with $i \neq j$) for $|\mathbf{y}| \neq 0$. In order to compute $M^2Ev_i^2$ we compute $G := M^2E[(\sum_{|\mathbf{y}|\neq 0} w_{\mathbf{x}\oplus\mathbf{y}}\mathcal{B}_q(\mathbf{y}))^2]$ for any fixed $\mathbf{x}$. For off-diagonal terms in the square of the sum, that is, for $\mathbf{y}' \neq \mathbf{y}$, we obtain again terms of the form $Ew_{\mathbf{x}\oplus\mathbf{y}}w_{\mathbf{x}\oplus\mathbf{y}'} = 1/(M(M+1))$. The total sum of these off-diagonal terms in $G$ is $\sum_{\mathbf{y}\neq\mathbf{y}',1\leq|\mathbf{y}|,|\mathbf{y}'|\leq n}\mathcal{B}_q(\mathbf{y})\mathcal{B}_q(\mathbf{y}')/(M(M+1))$. For the diagonal terms the expectations satisfy $Ew_{\mathbf{x}\oplus\mathbf{y}}w_{\mathbf{x}\oplus\mathbf{y}} = 2/M(M+1)$. For these terms we have

$$\begin{aligned} H &:= \sum_{|\mathbf{y}|\neq 0}\mathcal{B}_q(\mathbf{y})^2 \\ &= \sum_{k=1}^{n}\binom{n}{k}q^{2k}(1-q)^{2n-2k} \\ &= [q^2+(1-q)^2]^n - (1-q)^{2n}, \end{aligned}$$

and therefore

$$\begin{aligned} K &:= \sum_{\mathbf{y}\neq\mathbf{y}',1\leq|\mathbf{y}|,|\mathbf{y}'|\leq n}\mathcal{B}_q(\mathbf{y})\mathcal{B}_q(\mathbf{y}') \\ &= [1-(1-q)^n]^2 - \{[q^2+(1-q)^2]^n - (1-q)^{2n}\}. \end{aligned}$$

Therefore,

$$\begin{aligned} G &= \frac{M}{M+1}[2H+K] \\ &= \frac{M}{M+1}\{[1-(1-q)^n]^2 \\ &\quad + [q^2+(1-q)^2]^n - (1-q)^{2n}\} \\ &= \frac{M}{M+1}\{[q^2+(1-q)^2]^n \\ &\quad - 2(1-q)^n + 1\}. \end{aligned} \tag{A.3}$$

It follows that

$$EW = \phi_{ro}[G/D^2 - 1] - \frac{1}{M+1}\phi,$$

which is (6.5). □

## ACKNOWLEDGMENT

The numerical work in this paper was done by the R Package and Wolfram Mathematica. We are grateful to Larry Goldstein for insightful comments and questions, to Alexander Vlasov for helpful comments on the Google data and other matters, and to Carsten Voelkmann for two very careful readings of the manuscript and many invaluable corrections. We are also thankful to members of the Google team and especially to John Martinis and Sergio Boixo for helpful discussions. Two referees, an associate editor, and the Editor provided very useful comments on the content and organization of the paper.

## FUNDING

The second and third author were supported by ERC advanced grant 834735.

## REFERENCES

[1] AARONSON, S. and GUNN, S. (2020). On the classical hardness of spoofing linear cross-entropy benchmarking. *Theory Comput.* **16** 11. MR4194611 https://doi.org/10.4086/toc.2020.v016a011

[2] ARRATIA, R., GOLDSTEIN, L. and KOCHMAN, F. (2019). Size bias for one and all. *Probab. Surv.* **16** 1–61. MR3896143 https://doi.org/10.1214/13-ps221

[3] ARUTE et al. (2019). Quantum supremacy using a programmable superconducting processor. *Nature* **574** 505–510. https://doi.org/10.5061/dryad.k6t1rj8

[4] CHAN, K. C. G. (2013). Survival analysis without survival data: Connecting length-biased and case-control data. *Biometrika* **100** 764–770. MR3094451 https://doi.org/10.1093/biomet/ast008

[5] COX, D. R. and HINKLEY, D. V. (1974). *Theoretical Statistics*. CRC Press, London. MR0370837

[6] DEVITT, S. J., MUNRO, W. J. and NEMOTO, K. (2013). Quantum error correction for beginners. *Rep. Progr. Phys.* **76** 076001. https://doi.org/10.1088/0034-4885/76/7/076001

[7] GREENE, W. H. (2003). *Econometric Analysis*, 5th ed. Prentice Hall, Hoboken, NJ.

[8] HENSHER, D. A., ROSE, J. M. and GREENE, W. H. (2015). *Applied Choice Analysis*, 2nd ed. Cambridge Univ. Press, Cambridge.

[9] HUANG, C. et al. (2020). Classical simulation of quantum supremacy circuits. Available at arXiv:2005.06787.

[10] KALAI, G. (2020). The argument against quantum computers, the quantum laws of nature, and Google's supremacy claims. In *The Intercontinental Academia Laws: Rigidity and Dynamics* (M. J. Hannon and E. Z. Rabinovici, eds.). *Proceedings of the ICA Workshops* 2018 & 2019. World Scientific, Singapore. To appear.

[11] KALAI, G. and KINDLER, G. (2014). Gaussian noise sensitivity and boson sampling. arXiv preprint. Available at arXiv:1409.3093.

[12] KINGMAN, J. F. C., (1975). Random discrete distribution. *J. Roy. Statist. Soc. Ser. B* **37** 1–15. MR0368264

[13] KOTZ, S., BALAKRISHNAN, N. and JOHNSON, N. L. (2000). *Continuous Multivariate Distributions. Volume* 1: *Models and Applications*. Wiley, New York.

[14] LEHMANN, E. L. and CASELLA, G. (1998). *Theory of Point Estimation*, 2nd ed. *Springer Texts in Statistics*. Springer, New York. MR1639875

[15] NATHAN, G. (2011). Superpopulation models in survey sampling. In *International Encyclopedia of Statistical Science* 1575–1577. Springer, Berlin.

[16] NIELSEN, M. A. and CHUANG, I. L. (2000). *Quantum Computation and Quantum Information*. Cambridge Univ. Press, Cambridge. MR1796805

[17] PAN, F. and ZHANG, P. (2021). Simulating the Sycamore quantum supremacy circuits. Available at arXiv:2103.03074.

[18] PEDNAULT, E., GUNNELS, J. A., NANNICINI, G., HORESH, L. and WISNIEFF, R. (2019). Leveraging secondary storage to simulate deep 54-qubit Sycamore circuits. Available at arXiv:1910.09534.

[19] PORTER, C. E. and THOMAS, R. G. (1956). Fluctuations of nuclear reaction widths. *Phys. Rev.* **104** 483–491.

[20] RAUSAND, M. and HØYLAND, A. (2004). *System Reliability Theory*: *Models*, *Statistical Methods*, *and Applications*, 2nd ed. *Wiley Series in Probability and Statistics*. Wiley, Hoboken, NJ. MR2016162

[21] SÄRNDAL, C.-E., SWENSSON, B. and WRETMAN, J. (1992). *Model Assisted Survey Sampling*. *Springer Series in Statistics*. Springer, New York. MR1140409 https://doi.org/10.1007/978-1-4612-4378-6

[22] WANG, Y. and SONG, X. (2020). Quantum science and quantum technology. *Statist. Sci.* **35** 51–74. MR4071358 https://doi.org/10.1214/19-STS745

[23] ZHONG, H.-S., WANG, H., DENG, Y.-H., CHEN, M.-C., PENG, L.-C., LUO, Y.-H., QIN, J., WU, D., DING, X. et al. (2020). Quantum computational advantage using photons. *Science* **370** 1460–1463. https://doi.org/10.1126/science.abe8770